TeamForge 8.2

Contents

TeamForge 8.2

CollabNet TeamForge® is the industry's only open and extensible collaborative software delivery platform for distributed teams. It enables organizations to improve collaboration, agility and governance by connecting teams, processes, IP and tools via a centralized, secure, web-based system. With TeamForge, your entire organization can collaborate and become more agile at scale. And it scales to tens of thousands of users.

How to use TeamForge 8.2

TeamForge makes it easy and fun to manage all phases of the software lifecycle, from requirements through release. You get software configuration management (SCM), continuous build and test, issue tracking, project management, lab management, and collaboration tools, all in a single tightly-integrated, cloud-based platform that empowers distributed teams to deliver great software.

For highlights of the many new features that TeamForge 8.2 offers, take a look at the *release notes*.

For downloading this Help as a PDF file, see *Downloadable help*.

Get started

Take these steps to get started quickly and efficiently on your TeamForge 8.2 site.

☞ **Note:** If you are coming to TeamForge from a CollabNet Enterprise Edition site, see *Moving to a CollabNet TeamForge project from a CollabNet Enterprise Edition project* on page 978 for guidance on how to keep doing what you were doing on your old site.

Quick start: Working on a TeamForge project
Jump right in! Join a project, check out the code, meet your collaborators.
Join a TeamForge site
To collaborate with others on a CollabNet TeamForge site, start by getting a user account.

☞ **Note:** You need a license to use a TeamForge site. Your site administrator may have already assigned you a license. If not, you'll be asked to choose the type of license you need when you create your account.

- An ALM license enables you to use the full set of tools on the site, including Source Code, Trackers, Documents, Tasks and Reports.
- An SCM license enables you to use the site for basic source code management. You can browse repositories and participate in Wiki conversations.

1. If your site authenticates through LDAP, follow these steps.

   If your site does not authenticate through LDAP, skip to the next set of steps.

   a) In the **Log Into CollabNet TeamForge** section of the site home page, enter your corporate LDAP user name and password.

   ☞ **Tip:** In most cases, your user name and password are the user name and password with which you log in to your corporate network.

   b) Click **Log In.**
   c) On the **Create Account** page, re-enter your LDAP password.
   d) Enter your full name and email address and click **Create**.

2. If your site does not authenticate through LDAP, follow these steps.

   a) Click **Create an Account** in the **New Users** section of the TeamForge home page.
   b) On the **Create New Account** page, enter a user name for your account.

   ☞ **Note:** Your user name must meet these criteria:

   - User name is case sensitive.

- Minimum number of characters as specified in the `site-options.conf` file.
- No spaces.
- Should have at least one letter.
- The first character is a letter.

c) Enter and confirm a password.

d) Fill in the rest of the fields and click **Create**.

3. Check your email at the address that you provided.

4. Follow the link in the email to the **Log In** page.

5. Enter your user name and password, and click **Log In**.

Your TeamForge account is now activated.

Join a project

To create, store, and share work on a CollabNet TeamForge site, first join a project.

1. Log into TeamForge .

2. From your **My Page** menu, click **Projects**.
Now you're looking at the projects of which you are already a member, if any.

3. Click the **All Projects** tab to see all TeamForge projects, based on each project's access settings.

4. Select the project you want to join and click **Request Membership**.

5. On the **Join Project** page, explain why you want to join the project in the **REQUEST COMMENT** text box, and click **Submit**.
Your request is submitted to the project administrator for approval. You will receive an email notification when your request is either approved or denied.

☞ **Tip:** You can also request project membership by clicking **Join this Project** on the home page of the project you want to join.

Identify yourself

To help get started on collaborating with other project members, you can provide some information about yourself.

This information appears whenever someone clicks your name anywhere on the site. For example, when an artifact is assigned to you, your name appears as a link that other users can click.

1. Click **My Settings** from the **My Page** menu.

2. In the **User Details** section, click **Edit** and provide some information to help potential coworkers get to know you.

a) Click **Browse** to upload a 100-by-100-pixel picture of yourself, or of something that suggests who you are.

b) Under **Detail**, provide a summary of your interests, skills, or other characteristics.

3. Select the language you prefer to use. You can choose to receive emails from your TeamForge site in English, Chinese, Japanese, or Korean.

☞ **Note:** If your browser is set to use one of the available languages, you should already be seeing TeamForge in that language in your web browser.

4. If necessary, review and change your password, official name, and email address.

☞ **Note:** This is the email address where you receive alerts about changes to items you are involved with, such as a discussion forum you are monitoring or a code commit associated with an artifact you created.

5. Click **Save**.

Change your password

To protect the security of your TeamForge account, change the password regularly.

1. Select **My Settings** from the **My Page** menu.

2. In the **User Details** section, click **Change Password**.

3. In the **My Workspace / Change Password** page, enter your current password.

4. Enter your new password and confirm it.

The password requirements depend on the security policies enforced on your site. Here's an example:

5. Click **Update**.
You will see a message that your password was changed successfully.

Get the code

Browse TeamForge to find the code you want to work on, then check out the code.

You can view the contents of each file in a repository, plus additional information about each file such as revision history, comments, date and time of submission, and branch and tag information. You can also view differences (diffs) between any two files.

☞ **Restriction:** You can see only those paths in the repository that the repository administrator has given you access to.

☞ **Note:** If you're getting code from a Subversion replica repository, the TeamForge account used while setting up the replica determines what's available to be checked out. This account could have been provided total access to the master repository, or restricted access using path-based permissions.

1. Click **SOURCE CODE** from the **Project Home** menu.
2. On the list of project repositories, click the name of the repository in which you want to view code.
For each file, the revision number, time since check-in, author, and last log entry appear in the **Repository Browser**.

- To view a file, or to view the diffs between two files, click the file name.
- To view a specific version of the file, click **Download**.
- To view the differences between two files, do either of these:

  - Click **[select for diffs]** next to each of the two files that you want to compare.
  - Enter the file revision numbers in the Diffs between boxes at the bottom of the page.

3. If you need to diff files, choose a display from the **Type of Diff** menu, then click **Get Diffs**.
The differences between the two files are displayed.

4. Use your source control client to check out the code to your local machine.

Internal code browser

For Subversion and Git repositories, you have the option to use the TeamForge code browser which is turned on by default while integrating the source code server. Fore more information, see *Integrate a source code server*.

5. On the list of project repositories, click the name of a Subversion or a Git repository in which you want to view code.
On the top right of code browser, you can select the branch/tag (for Git) or specify the revision (for SVN) you want to browse.

- **View**: This tab allows you to do the following:

  - Browse through the folder hierarchy of the repository and view the content of specific files. For any folder or file you are viewing within a branch (Git) or revision (SVN), you can obtain the commit information pertaining to its last update.
  - While viewing a single specific commit or a file, you can see the paths that were modified in that commit, the associations including JIRA such as builds, code reviews and so on (from Orchestrate events) for the specific commit and the differences between files in that commit.

    ☞ **Important:** To view the associations, you must have installed Orchestrate and must have RBAC (role-based access control) permissions to use the "Pipeline Read" or "Reporting API" of

TeamForge Orchestrate. If either of these requirements is not met, this section will not show up at all.

- While viewing a folder, if there is a file named `readme`, `readme.txt` or `readme.md` that file will automatically be rendered beneath the list of files in the folder. If the file contains markdown formatting, it will be rendered as rich text.
- **Changes**: This tab lets you view all of the commits that touched a specific path you are browsing within a branch or revision. Click a commit to view its details.
- **Download**: This tab allows you to download a copy of the required file.
- **Graph**: This tab provides a graphical representation of the changes made including branching and merging of repositories.
- **Branches** (for Git): This allows you to see all of the branches in the repository in their relation to the default (master) one. Using **Compare Branch** you can see the commits in the branch that do not exist in the default branch.

Find a forum

Read through a project's discussion forums to find out about the project's plans, accomplishments, and challenges.

1. Click **Discussions** from the **Project Home** menu.
   The **Forum Summary** page displays a list of all project forums, and summary information about each forum's activity.
2. Click a forum name.
   The **Topic Summary** page displays a list of all forum topics and summary information about each forum topic's activity.
3. Click a topic name.
   The **Forum Topic** page displays all messages that have been posted in response to the original forum topic.

   You can configure your view in the following ways:

   - To set the number of messages that display on each page, choose a value from the View menu, then click **Update**.
   - To display threaded messages, choose one of the following options, then click **Update**.

     - **Nested - by Reply** - Displays threaded messages in a hierarchical, nested format of up to 20 layers. This is the default display.
     - **Flat - Chronological** - Displays all messages in chronological order with no nesting.

Catch up on project news

Watch project news updates to stay up to date on events that might affect your participation in the project.

News items appear on the project home page, on the **News** tab of each user's **My page**, and on the CollabNet TeamForge home page.

1. Go to the project home page.

   - From within the project, click **Project Home**.
   - From anywhere else in CollabNet TeamForge, choose the project from the list of projects available under **My Workspace** > **My Projects.**

2. Look for news items at the bottom of the project home page.

   Each news item lists its author and the date it was posted.

Find project resources

When you want to search for TeamForge objects such as tracker artifacts or documents, you can quickly search using a unique identifier or a keyword or you can perform an advanced search.

☞ **Tip:** In addition to the **Jump to ID** search, you can select an object category from the drop-down list and search within the selected object category. You can also select **Advanced Search** from the drop-down list to perform an advanced search.

1. Log on to TeamForge.

   ☞ **Note:** If you are not logged on, you can search only projects and items that have been designated public.

2. If you know the unique identifier of an object and want to quickly go to the object, type the unique identifier in the **Jump to ID** text box and click the search icon.

   ☞ **Note:** The default quick search option is **Jump to ID**.

3. If you want to do a keyword search of a specific object type (such as documents or discussions), type the keyword in the text box, select an object type from the drop-down list and click the search icon.

   The following table lists the search-able object types you can select from the drop-down list.

Searchable object types	Description
Discussions	Select this option to search in discussion forums.
Documents	Select this option to search for documents.
File Releases	Select this option to search in file releases.
News	Select this option to search project news.
Project Pages (Visible)	Select this option to search project pages.
Projects	Select this option to search projects.
Source Code	Select this option to search the source code.
Tasks	Select this option to search for tasks.
Trackers	Select this option to search for tracker artifacts.
Users	Select this option to search for users.
Wiki	Select this option to search in Wiki pages.

4. If you want to perform an advanced search, select **Advanced Search** from the drop-down list.

5. On the **Search Criteria** page, enter the keywords to search for.

6. Select the **Search Attachments** check box and the **Search Comments** check box if you want to search attachments and comments respectively.

   Attachments refer to tracker artifact attachments. Comments refer to tracker artifact comments and task comments.

7. Select one of the two options, **Search Active Versions Only** or **Search All Versions**, to specify whether you want to search active document versions only or all document versions respectively.

   Searching only active document versions allows you to eliminate search results for outdated documents.

8. Click **Search**.

Your search results are organized by TeamForge application. The search score indicates the relevance of each result to your search criteria.

☞   **Note:** You can see only those items that your project membership and permissions allow you to see.

Filtering tables

Filtering lets you view a subset of a list of items that match your filter criteria.

The TeamForge application uses tables for presenting list of items such as tracker artifacts, documents, tasks, and so on. You can filter a subset of a list of items that meets your filter criteria on all the tables with filter fields.

1. Specify the filter criteria in one or more filter fields (at the top of each column) and click **FILTER**.

   • You can find a filter field at the top of each column in most of the tables in the TeamForge application.
   • The filter field could be a text box or a drop-down list with multi-select check boxes.

   • You can type your filter criteria in the text boxes. The search text is case-insensitive.
   • You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the *FILTER_DROPDOWN_MAX_SELECTION* token in the `site-options.conf` file to increase or decrease the count.
   • **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing "R" instantly shows all statuses having the alphabet "R". The search text is case-insensitive.

   • Some search filters may not appear if your site administrator has not enabled them.

2. After filtering, if you want to save a filter for future use:

   a) Click **FILTER** and select **Save** from the drop-down list.
      The **Save Filter As** window appears.
   b) Type a name for the filter in the **FILTER NAME** text box.
   c) Click **Save**.
      The filter is saved. You can view and select the saved filters at a later point in time by clicking the **Filter** drop-down list.

   ☞   **Note:** You can save filters only in specific contexts. This feature may not be available in all the tables where you can filter table list items.

3. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.

4. To delete saved filters:

   a) Click **FILTER** and select **Delete** from the drop-down list.
      The **Select Filters To Be Deleted** window appears

   b) Select one or more filters to delete.

      👉 **Tip:** Press and hold the **Ctrl** key to select more than one filter.

   c) Click **Delete**.
      A message such as `2 tracker filter(s) have been deleted successfully.` is displayed if the process was successful.

Quick start: Managing a TeamForge project

Jump right in! Set up your project site, add your project members, plan out your work.

Create a TeamForge project

Create a new project when you have identified work to be done that has its own distinct character, dependencies or schedule.

What constitutes a project depends on your organization. Some organizations favor a small number of big, centralized projects. Others prefer a larger number of smaller, specialized projects. Your site administrator can help you decide if your work should be part of a larger project or a project of its own.

1. Access the **Projects** page through either of the following ways:

   • Go to **My Workspace** > **View All Projects**
   • Go to **My Page** > **Projects.**

2. Click **Create New Project**.

3. On the **Create Project** page, give the project a name and a brief description.

   • The name will appear in project lists and on the project's home page.
   • A terse description is recommended. There will be unlimited room to discuss the project's aims and methods in detail on the project pages themselves.

4. Provide a URL name for the project, if you want the URL for the project to be different from the internal project name.

   If you do not enter a URL name, the project URL will be the same as the project name.

5. If your site administrator has provided project templates, select the appropriate one for your new project.

   Project templates give you ready-made artifact types, work flow support, user roles and other start-up content appropriate to the kind of project you are creating.

6. Click **Create**.

The project is submitted to the CollabNet TeamForge site administrator for approval. You will receive an email notification when the site administrator approves or rejects your project. When your project is approved, you can get to it from your **My Projects** page or from the **Projects** menu in your navigation bar.

Add users to a project

Before a person can work on a project, you have to make him a member of the project.

You can make any registered user on your CollabNet TeamForge site a project member. You can assign roles to the user at the same time.

👉 **Note:** Project members, roles and the associated permissions can be inherited via project hierarchy and reused in subprojects. If your project is a subproject of any other project, you may have inherited some roles, project members or user groups.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin** menu, click **User Membership**.

3. Click **Add**.

4. On the **Add User** page, find the users you want by one of these methods:

- Under **Search for Users**, filter the list of site users eligible to join this project. You can filter by full or partial name or user name.

  ☞ **Note:** Search text is not case-sensitive.

- Browse the list of registered users on the site. Sort them by name, user name, email address or membership status.

  The inherited project members continue to hold the inherited roles with corresponding permissions, as specified in the source projects.

  ☞ **Note:** If a site has a great many users, you must filter them first to narrow down the list. This helps avoid slowing down the system.

5. Select the users you want to add.

6. Under **Assign Roles**, select the roles you want the users to have.

   ☞ **Tip:** If you prefer, you can skip this step and assign roles later on the **Project Admin** > **Permissions** page. Note that using the **Assigned Project Members** page, you can assign roles only to the direct project members.

7. Save your changes.

   - Click **Save** to return to the **Project Membership** page.
   - Click **Save and Add More** to keep adding users.

Create a project page

To provide information and functionality to people viewing your project, build one or more project pages.

1. Go to the page to which your new page will belong.

   ☞ **Tip:** Any project page can have sub-pages belonging to it. A page that belongs directly to the project home page is called a "top-level page."

2. Click **Configure: On**.

3. Choose where your new page will fit in your project's structure.

   - To create a page just under the project home page, click **Add top-level page**.

     A top-level page's title is always visible in the navigation tree at left.
   - To create a page under the page you are on right now, click **Add sub-page**.

4. Give your new project page a title.

   Keep the title brief and descriptive.

5. Choose who can see this page.

   ☞ **Note:** Your choice will apply to all subpages that you create under this page.

   - To show this page to anyone with the necessary permissions, select **Visible**. For example, if you have defined a group of users who have access to your project, your new project page is visible only to those users. If your project is open to the public, anyone in the world can see it. Use this option when the information on this page is ready for a wide audience.
   - To show this page only to users with the project administrator role, select **Hidden**. Use this option if you are drafting content that you aren't ready to share yet, or want to share only with other project managers.

6. Click **Save**.

7. Click **Configure: Off**.

Now you are ready to build functionality into your project page with components such as text, news or tracker queries.

Create a tracker

Create a tracker whenever you need to report and track bugs, feature requests, support requests, or any other type of issue where ownership, status, and activity must be managed.

Individual tracker entries are referred to as tracker artifacts. A tracker is a set of tracker items with a common purpose, such as bug reports, feature requests, or tasks.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. Click **Create**.
4. On the **Create Tracker** page, provide a name and description for the tracker.

   ☞ **Tip:** Descriptions help users learn how best to provide the information you want from them. To maximize your chances of getting useful data, make your description as informative as you can.

5. Select an icon that suggests the type of work the tracker is handling.

   This icon will appear with any artifact in this tracker, wherever it is viewed on the site. For example, if someone brings an artifact from this tracker into a planning folder, users of the planning folder can glance at the artifact's icon to see where it comes from.

6. Select the relevant unit from the **DISPLAY EFFORT IN** field. The units displayed here are configured based on the size of the artifacts in the tracker. Eg. Select the unit as **HOURS** for a tracker of small defects, **DAYS** for a tracker of tasks, and **WEEKS** for a tracker of epics.

   ☞ **Note:** Configure the units at the project level and not at the planning folder level.

7. Select **INCLUDE FOREIGN CHILDREN** to include points and efforts from children artifacts across the projects in TeamForge.

   ☞ **Note:** In a parent artifact, enabling **CALCULATE POINTS** field sums and rolls up the points from all its children artifacts within the project. In this total, if you want to include children artifacts from other projects across TeamForge, have the **INCLUDE FOREIGN CHILDREN** option enabled.

8. Click **Create**.
   The new tracker appears at the bottom of your list of trackers.
9. If necessary, drag the tracker to a place in your tracker list that makes sense.

   The order you set here controls the order of every tracker list in your project.
10. You'll probably need some custom fields to capture information that's specific to your project.

    See *Create your own tracker fields* on page 38.
11. To speed up the team's work, you may want to set up some rules for automatically reassigning artifacts when their contents change.

    See *Create a tracker work flow* on page 40.

Create a planning folder
A planning folder is a way to organize work into feasible chunks and monitor its progress. As a project admin or as a user with the appropriate permissions, create and populate all the planning folders you need to capture the work you are planning.

When you've thought through your plan, express it in one or more planning folders.

☞ **Tip:** It often makes sense to set up planning folders *after* you have outlined and analyzed the features you plan to deliver. See *Define the scope of your project* on page 120.

For your agile projects, you have the option to create planning folders specific to iterations and releases, or you can create generic planning folders which you may customize later. When you've thought about the general categories the work falls into, you are ready to create planning folders that reflect those ideas.

A planning folder can represent:

- A set of tasks, such as Iteration 3, or "Initial infrastructure development."
- A period of time, such as "April," or "Q2-2010."
- A phase of development, such as "Testing" or "Deployment."
- A component of the product, such as "Chapter 12" or "Rear stabilizer."

1. Click **TRACKERS** from the **Project Home** menu.

2. Under **PLANNING FOLDERS**, click the folder in which you want to create a new planning folder. Click **NEW**.
   An appropriate planning folder creation page is displayed.

3. Select the type of planning folder you want to create: **Release**, **Iteration** or **Folder.**

   While "Release" and "Iteration" are specific planning folder types, "Folder" is your standard, generic planning folder. Though having multiple planning folder types help you logically organize your planning folders, there's no hierarchical restrictions on these different planning folder types. For example, as a user, you have the liberty to create a "Release" planning folder as a child of an "Iteration" planning folder, which may not be the ideal use case in an Agile project.

4. Enter a brief and descriptive name for your planning folder.
   For example:

   • In an agile project, depending on your requirement, you can create two or more iteration folders called "Iteration 1", "Iteration 2" and so on within the Release planning folder.
   • In a phased, waterfall-style project, you might name your first planning folder "Design," the next "Build," and so on.

   ☞ **Tip:** Don't worry if you don't have anything of interest to put into your planning folders yet. The parallel process of filling out the feature tree will provide plenty of material for this.

   ☞ **Note:** This field is mandatory.

5. Use the **DESCRIPTION** to briefly signal the kind of work that will be contained in this planning folder.

   Include enough information to help people get up to speed quickly when they join your project. But save most of the detail for the individual tracker artifacts, where project members will spend most of their time.

6. For **STATUS**, (available since TeamForge 5.4), select a value that communicates where the planning folder is in its life cycle.

   For a new planning folder, you'll probably want to select Not started.

   ☞ **Tip:** You can create more values to choose from in your Project Settings.

7. If you have set up a file release in the File Releases tool to deliver the work you are tracking here, you can identify it in the **FILE RELEASE** field (available since TeamForge 5.4).

Any artifact you add to this planning folder will also appear in the Planned Tracker Artifacts tab when you look at that file release.

> ☞ **Note:** If you are creating an Iteration folder, this field displays the default file release value set at its parent folder level ("Release" folder) which you can modify.

8. For **CAPACITY**, (available since TeamForge 5.4), provide a number that expresses how much work your team thinks it can do in the period represented by this planning folder. This field is available only for "Iteration" and "Folder" types.

> ☞ **Tip:** When you create a new planning folder, it can be a good idea to set its capacity to zero until you get a feel for the artifacts and resources you are working with.

9. For **POINTS CAPACITY**, (available since TeamForge 7.1), provide a number that represents the amount of work (expressed as number of story points) that you think can be handled in that specific planning folder. This field is available only for "Release" and "Folder" types.

> ☞ **Note:** To assist project managers gauge the planning folder's points capacity, when a release or standard planning folder is selected in one of the *planning board* swim lanes, this information is displayed as a ratio of planned number of story points versus the actual number of story points. This information does not apply to Iteration folders and therefore does not appear on the planning board swimlanes when you select an Iteration planning folder. For more information, see *Use the Planning Board* on page 131.

10. Select or enter a start and end date for the work that this planning folder will contain.

You can change these dates at any time. However, it's important to specify dates even if they're not firm, because useful visual aids and reports depend on them.

> ☞ **Note:**
> - The start and end dates of the child planning folder should be within that of its parent folder.
> - These dates are mandatory for an "Iteration" planning folder because the burndown chart for Iteration planning folders is enabled only when you specify the start and end dates of an Iteration planning folder. Also, these dates are not mandatory for a Release planning folder because the burndown chart of a Release planning folder takes into account the start and end dates of its child planning folders (Iterations) and not its own.

11. Select the unit that is suitable for the planning folder from the **DISPLAY EFFORT IN** field. Eg. if the planning folder represents a sprint, then select the unit as **Hours** and if it represents a collection of sprints, then select the unit as **Days**.

> ☞ **Note:** Units are configured at the project level and not at the planning folder level.

12. Select either **Effort** or **Points** from the **BURNDOWN VIEW BY** drop-down list to set the View Mode in the burndown charts (in the planning folder **List Artifacts** page). This field is available only for "Folder" type.

Depending on the planning folder type, the burndown chart is displayed.

- Burndown chart for the standard planning folder type which includes the View Mode:

- Burndown chart for "Iteration" planning folder type:

- Iteration-wise burndown chart for a "Release" planning folder type:

13. Click **Create**. The planning folder is created.

Set up hardware for your team to use

When you set up Lab Management, your team members can use TeamForge to access their own virtual machines for developing and testing.

1. Go to your Lab Management project.

2. Select **Project Admin** from the **Project Home** menu.

3. On the **Project Admin** menu, click **Tools**.

4. Click **Add Tool**.

5. Select **Other** from **Select Tool Type** drop-down.

6. Enter the display name as **Lab Mgmt**.

7. Select the **Show in Toolbar** checkbox.

8. Enter the server location or URL of your Lab Management server.

9. If you are a TeamForge administrator, select whether you want to use single sign-on for the linked application.

   • If you use single sign-on, TeamForge manages authentication for Lab Management, and users don't have to log into Lab Management after they have logged into TeamForge.

   • If you do not use single sign-on, users must log into Lab Management using its native authentication system.

10. Click **Browse** and select this icon for the linked application:

    This icon will appear with the name **Lab Mgmt** in the project navigation bar.

11. Click **Save**.

A **Lab Mgmt** button is added to your **Project Home** menu.. Clicking it launches the application in the main TeamForge project window.

Contribute to a CollabNet TeamForge project

CollabNet TeamForge is a set of integrated tools designed to make your collaborative experience more efficient, productive and fun.

Plan and track anything

Use the issue tracking system to track and manage everything from requirements, user stories, and feature requests to defects, build issues, and customer support cases.

Project administrators can model any desired lifecycle process by customizing tracker fields to correspond to the different states required, defining a work flow that governs the transitions between those states, and specifying which users can make those transitions.

Estimate effort for agile development

Once the product features have been sketched out, if falls to the project team to figure out what it will take to deliver them.

Effort estimates are almost always more accurate when they address small, discrete chunks of work. But you need to forecast effort for a large, complex undertaking. TeamForge helps bridge this gap by dynamically calculating effort estimates at multiple levels.

There are two ways to determine the effort involved in an artifact:

• Break the work down into discrete chunks, then use your experience and judgment to come up with a real-world estimate for this artifact.

• Have the effort estimates for all the artifact's children (if any) automatically added and shown in this artifact.

You will be using a combination of these techniques as you work through the estimating process.

Analyzing the work is an iterative, interactive process. As you go through it, you'll find your effort estimates changing shape in response to feedback from scoping the product and planning out the work.

1. With your team, review the artifacts the product owner has created.

   How much effort will each one require? Do any of them require special skills or training? Record your observations about each artifact in its **Comment** field.

   Note: This is a rough cut. Don't try to come up with a final effort estimate yet.

2. For each user story, identify the specific tasks that will be required. Create an artifact to contain each of these tasks.

   Declare each of these task artifacts a "child" of the initial artifact.

   Now you can attempt to estimate the work involved in each task artifact. This will be the raw material for your higher-level effort estimates and progress tracking.

3. Back in the "parent" artifact, select **Sum effort from children**.
   Now this artifact's effort field shows the total of the effort units assigned to each of its "children."

   ☞ **Note:** If an artifact has children that belong to another project, the effort value for those child artifacts is not counted.

4. As you proceed, you will find the overall picture taking on definition.

   Relationships among artifacts will emerge, and many of them will seem to fall into one logical group or another. Let these groupings determine the names and purposes of your planning folders.

Find tracker artifacts

The tracker contains a comprehensive search system that enables you to find a specific artifact or set of artifacts quickly.

View the right tracker columns

When you're looking at the artifacts list in a tracker, a planning folder or a team, you can select the columns you want to see, either for this session or permanently.

You set your column preferences for each tracker, planning folder or team independently. If your project administrator has set default columns for the entire project, your individual column choices override those settings.

1. Click **TRACKERS** from the **Project Home** menu.

2. Select a tracker, planning folder or team and click **COLUMNS** > **Configure.**

   • If you've already saved a column configuration, click it and skip the rest of these steps.
   • To go back to the default column configuration, click **System (default)** and skip the rest of these steps.
   • To set up a new configuration, click **Configure.**

3. Choose your columns.

   a) Move the columns you want from **Available Columns** to **Selected Columns**.

      Artifact ID : Title, **Priority** and **Status** are required columns.

      ☞ **Note:** Selecting more columns can increase the time required to load the listing page.

   b) Remove any columns you don't need from **Selected Columns**.
   c) Use the move up and move down arrows to change the display order of the columns.

4. Apply your choices to your view of the tracker.

   • To use this arrangement this time only, click **Apply**. The next time you log in, you'll start with the default view again.
   • To save your column layout for repeated use, click **Apply and Save**, then give your arrangement a name. The next time you log in, you'll see the column arrangement you just selected. (If you've sorted the records in your view, that sort order is saved too.)

      ☞ **Tip:** If you are editing a column configuration that already exists, you can rename it by saving it under a new name.

5. To make the same set of columns appear every time you come to this tracker, planning folder or team, click **COLUMNS** > **Save** and from **Save Column Configuration** page, select **Make this my default view**.

View the right tracker artifacts

When a large number of tracker artifacts makes it difficult to find the one you want, filter and sort the list to narrow down the possibilities.

1. Click **TRACKERS** from the **Project Home** menu.

2. On the **Tracker Summary** page, click the title of the tracker in which you want to look at artifacts.

3. Specify the filter criteria in one or more filter fields (at the top of each column) and click **FILTER**.

- You can find a filter field at the top of each column in most of the tables in the TeamForge application.
- The filter field could be a text box or a drop-down list with multi-select check boxes.

- You can type your filter criteria in the text boxes. The search text is case-insensitive.
- You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the *FILTER_DROPDOWN_MAX_SELECTION* token in the `site-options.conf` file to increase or decrease the count.
- **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing "R" instantly shows all statuses having the alphabet "R". The search text is case-insensitive.

- Some search filters may not appear if your site administrator has not enabled them.

4. After filtering, if you want to save a filter for future use:
    a) Click **FILTER** and select **Save** from the drop-down list.
       The **Save Filter As** window appears.
    b) Type a name for the filter in the **FILTER NAME** text box.
    c) Click **Save**.
       The filter is saved. You can view and select the saved filters at a later point in time by clicking the **Filter** drop-down list.

    ☞ **Note:** You can save filters only in specific contexts. This feature may not be available in all the tables where you can filter table list items.

5. To delete saved filters:
    a) Click **FILTER** and select **Delete** from the drop-down list.
       The **Select Filters To Be Deleted** window appears
    b) Select one or more filters to delete.

    ☞ **Tip:** Press and hold the **Ctrl** key to select more than one filter.

    c) Click **Delete**.
       A message such as `2 tracker filter(s) have been deleted successfully.` is displayed if the process was successful.

6. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.

7. Use the up-down arrow at the top of any column to sort your list by that column.

   • Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
   • Click the up-down arrow again to reverse the sort order.

View the right planning folder artifacts

When a large number of artifacts in a planning folder makes it difficult to find the one you want, you can filter tracker artifacts by their overall status or you can filter and sort the tracker artifact list column-wise in the Planning Folders list view.

• In the Planning Folder list view, you can filter only by **Priority, Artifact ID, Title, Assigned To** and **Team** columns.
• Though the artifacts are listed in a tree view (parent artifact with its child artifacts), the filter is applicable only for the parent artifacts and not their children.
• The filter is available only in the Sort mode and not in the Rank mode.
• The filter that you set is retained even after you navigate to other pages and return to this page.

1. Click **TRACKERS** from the **Project Home** menu.
2. Click **Planning Folders**.
3. On the Summary page, click the planning folder in which you want to look at artifacts.
4. Specify the filter criteria in one or more filter fields (at the top of the filterable columns) and click **FILTER**.

   • The filter field could be a text box or a drop-down list with multi-select check boxes.
   • You can type your filter criteria in the text boxes. The search text is case-insensitive.
   • You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the *FILTER_DROPDOWN_MAX_SELECTION* token in the `site-options.conf` file to increase or decrease the count.
   • **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. The search text is case-insensitive.

5. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.
6. Use the up-down arrow at the top of any column to sort your list by that column.

   • Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
   • Click the up-down arrow again to reverse the sort order.

Filter by artifact status

In addition to the column-wise filter, you can also filter and view artifacts based on their status alone using the status drop-down list.

7. Select **All**, **Open only** or **Closed** from the drop-down list and all the relevant artifacts (including child artifacts) are displayed. For example, select **Open only** and all the artifacts with associated status values such as 'Under Construction', 'Open' and 'In Progress' are displayed.

☞   **Note:** The statuses **All**, **Open only** and **Closed** are associated with user-defined status values while configuring a tracker field on the **Edit Tracker Field** page (**Project Admin > Tracker Settings**). For more information, see *Configure tracker "Selec"t field values* on page 39.

View the right team artifacts

You can filter artifacts by their overall status or you can filter and sort the tracker artifact list column-wise in the Teams list view.

- In the Teams list view, you can filter only by **Priority, Artifact ID, Title, Assigned To** and **Planned For** columns.
- The filter that you set is retained even after you navigate to other pages and return to this page.

☞   **Note:** The Sort and Rank modes are not available on the Teams list view.

Filter by columns

1. Click **TRACKERS** from the **Project Home** menu.
2. Click **Teams**.
3. On the Summary page, click the team in which you want to look at artifacts.
4. Specify the filter criteria in one or more filter fields (at the top of the filterable columns) and click **FILTER**.

   - The filter field could be a text box or a drop-down list with multi-select check boxes.
   - You can type your filter criteria in the text boxes. The search text is case-insensitive.
   - You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the *FILTER_DROPDOWN_MAX_SELECTION* token in the `site-options.conf` file to increase or decrease the count.
   - **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. The search text is case-insensitive.

5. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.
6. Use the up-down arrow at the top of any column to sort your list by that column.

   - Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
   - Click the up-down arrow again to reverse the sort order.

Filter by artifact status

In addition to the column-wise filter, you can also filter and view artifacts based on their status alone using the status drop-down list.

7. Select **All**, **Open only** or **Closed** from the drop-down list and all the relevant artifacts (including child artifacts) are displayed. For example, select **Open only** and all the artifacts with associated status values such as 'Under Construction', 'Open' and 'In Progress' are displayed.

   ☞   **Note:** The statuses **All**, **Open only** and **Closed** are associated with user-defined status values while configuring a tracker field on the **Edit Tracker Field** page (**Project Admin > Tracker Settings**). For more information, see *Configure tracker "Selec"t field values* on page 39.

Search a tracker

If the filter returns too many results or not enough, try the search facility.

Search for tracker artifacts

Find tracker artifacts by supplying a keyword, an artifact ID, a date range or some other value.

You can also search comments, attachments, and user-defined fields.

1. Click **TRACKERS** from the **Project Home** menu.
2. Set the scope of your search.

   - To search all the trackers in your project, click **Search Trackers**.
   - To search a specific tracker, click the tracker you want to search, then click **Search Tracker** in that tracker's list view.

3. In the **Tracker Search Criteria** section, enter the keywords to search for.

   ☞ **Note:** Wildcards are allowed.

4. Select the elements of the artifact to search.
   For example, to search for a keyword in the title and description fields select **All Text Fields**. To search for a keyword in the comments field, select **Include Comments**.

   ☞ **Note:** User-defined text fields are not included.

5. If you know the artifact's ID, enter that.
6. Click the calendar icon to select dates, if appropriate. You can also specify relative dates such as "Within the last 7 days".
7. Use as many of the remaining tracker search criteria as seems useful.
8. Click **Search**.

All tracker artifacts matching your search criteria are displayed.

☞ **Note:** If your search included muliple trackers, the icon next to each artifact in your search results can help you identify which tracker that artifact belongs to.

Find your own artifacts

To narrow your scope, try searching only for artifacts assigned to or submitted by you.

1. Click **TRACKERS** from the **Project Home** menu.
2. On the list of project trackers, click the title of a tracker.
3. In the tracker artifact list view, click **Search Tracker**.
4. In the **Tracker Search Criteria** section, select the **User running search** option in the **Assigned To** or **Submitted By** fields.
5. Click **Search**.

All tracker artifacts matching your search criteria are displayed.

Save a search for tracker artifacts

To reuse a search that you devised for tracker artifacts, save the search criteria.

To be able to save your search, you must first select a tracker, and create and run your search.

1. In the **Tracker Search Results** section, click **Save Search from Results**.
2. In the **Save Search As** page, enter a descriptive name for the search, and click **Save.**

The search appears under **My Saved Tracker Searches**. The name of the search, the tracker for which it was run, the criteria used, and a system-assigned ID are displayed.

Share a saved tracker search

To enable other users to use a search that you have devised, save it as a shared search.

To share your search with other users, you must be a tracker administrator.

1. In the **Tracker Search Results** section, click **Save Search from Results**.
2. In the **Save Search As** page, enter a descriptive name for the search.
3. Select the **Shared Search** option.

The search appears under **Shared Tracker Searches**. The name of the search, the tracker for which it was run, the criteria used, the user who created the search, and a system-assigned ID are displayed.

Run a saved search

To find artifacts in the current tracker or a different one in the project, run a saved search in the appropriate context.

1. Click **TRACKERS** from the **Project Home** menu.
2. On the list of project trackers, click the title of a tracker.
3. In the tracker artifact list view, click **Search Tracker**.
4. Select the **Show My Searches Across Trackers** and **Show Shared Searches Across Trackers** options.
5. Expand the **My Saved Tracker Searches** and **Shared Tracker Searches** sections.
   The saved searches for all trackers in the project are displayed.
6. To run a search for the current tracker, click the **Search** or **Search Here** link below the search name in the required search.
   All artifacts in the current tracker, that match the search criteria, are displayed.
7. To run a search for a different tracker, click the **Search There** link below the tracker name in the required search.
   All artifacts that match the search criteria in the selected tracker, are displayed.

Refine a saved search

Refine your saved searches and keep them updated.

To be able to refine (edit) a previously saved search, you must first run the search and click **Refine Search**.

1. Click **TRACKERS** from the **Project Home** menu.
2. Click **Search Trackers**.
3. Click **My Saved Tracker Searches**.
4. Click the **Search** link under the saved search (in **Saved Search Name** column) you want to refine.
   The saved search is run and results appear.
5. Click **Refine Search**.
   The Tracker Search Criteria for the saved search shows up.
6. Modify the search criteria and click **Save Search**.
   The **Save Search As** dialog box appears.
7. Type a name and click **Save**.

   You may save the search with the same name, particularly if you have already shared the search with others. Otherwise, you can save the search with a new name too.

The search is saved with your new search criteria.

Remove a saved search

To remove a saved search for tracker artifacts, delete it from the list of personal or shared searches.

👉 **Note:** Only tracker administrators can remove a shared search.

1. Click **TRACKERS** from the **Project Home** menu.
2. On the list of project trackers, click the title of the tracker in which you want to delete a saved search.
3. In the tracker artifact list view, click **Search Tracker**.
4. Expand **My Saved Tracker Searches** or **Shared Tracker Searches** so that the available searches are displayed.
5. Select a search and click **Delete**.
6. Accept the confirmation message.

The search is removed from the table where you deleted it.

Create a tracker artifact

Create a tracker artifact whenever you need to report and track a bug, feature request, support request, or other type of issue.

Individual tracker entries are referred to as tracker artifacts, or just artifacts.

1. In any tracker, planning folder or team view, click **Submit New Artifact in** and select the tracker in which you want to create your artifact.

   By default, your new artifact is created in the tracker, planning folder or team you are currently looking at.

2. Answer the questions posed by the required fields.

   👉 **Note:** Different trackers will have different combinations of fields to fill out, depending on what the tracker administrator has set up.

   a) Provide a **Title** and **Description** that summarize the issue or work item in a few words.

   👉 **Tip:** Descriptions help users learn how best to provide the information you want from them. To maximize your chances of getting useful data, make your description as informative as you can.

   b) For **Priority**, select a value that expresses the importance or urgency of the work you are describing.

   c) Assign the artifact to a team by selecting a name from the **Team** list.

   d) If you want to assign it to a specific project member, choose a name from the **Assigned To** list. This lists displays the names of all the project members, irrespective of the team you may have selected in the previous step.

   👉 **Note:** If your project administrator has configured the tracker to automatically assign artifacts to project members, you can skip this step.

   e) Select the planning folder that the work belongs to from the **Planning Folder** list.

3. Record any other information that may be appropriate.

   For example, if your project is using a Scrum-based methodology, your project manager may have provided a **Points** field to track estimates of relative effort.

4. Add a file attachment, if appropriate.

5. Save your changes.
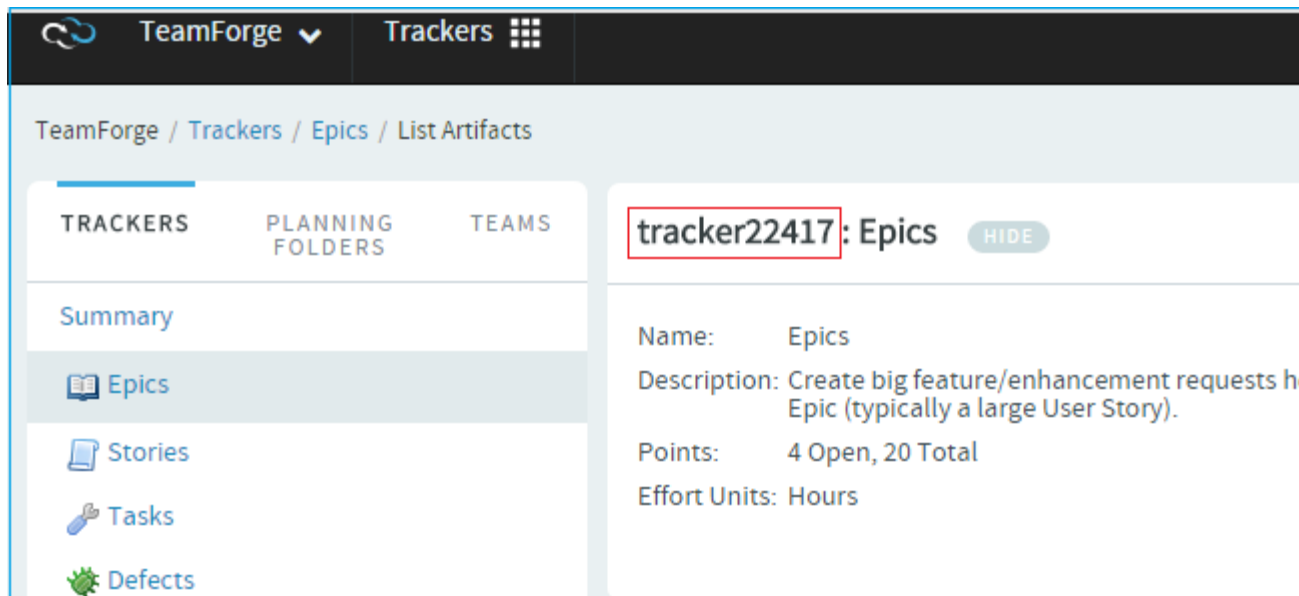
Create a tracker artifact by email

To create a tracker artifact without being logged into CollabNet TeamForge , send an email to the tracker.

You do not have to be logged into TeamForge to submit a tracker artifact using email, but you must have the tracker submit permission for the tracker to which you are submitting.

Send an email message to <tracker id>@< TeamForge server>.

👉 **Tip:** You can find the tracker ID on the **List Artifacts** page.

TeamForge maps your email to the tracker record like this:

Email field	Tracker field
To	Tracker email address
Subject	Artifact title
Body	Artifact description
Attachments	Attachments

Edit a tracker artifact

Updating the information in tracker artifacts is one important way that project members can work together effectively.

When work has been done on a tracker item, or more information is needed, the project member to whom the item is assigned should update the item's status accordingly. Comments from other project members help the artifact's owner decide how to handle the work.

For example, when the work defined in the tracker item is completed, change its status from **Open** to **Fixed**.

Your tracker administrator may have set up work flow rules that constain your ability to do certain kinds of updates. For example, an administrator may have specified that only users with the "QA Engineer" role can change an artifact's status from **Open** to **Closed**.

A project manager might also change an artifact's priority, return it to the submitter for additional clarification, or assign it to a project member for resolution or action.

Generally, the project member to whom the tracker artifact is assigned will update the status. Any project member with the appropriate permissions can add comments.

> ☞ **Tip:** Each comment in an artifact or task has a unique ID with its own URL. To link directly to a particular comment, copy that comment's URL and paste it into an email, a project page, or another comment. For example, to point to the third comment in artifact 12345, write `artf12345#3` in your comment. (If the artifact or task you are linking to is on a different site, give the complete URL, like this: `http://mysite.com/sf/go/task1234#3`.)

1. Click **TRACKERS** from the **Project Home** menu.
2. On the list of project trackers, click the tracker you want to update.
3. On the **List Artifacts** page, select the tracker artifact you want to update.
4. On the **View Artifact** page, click **Edit**.
5. On the **Edit Artifact** page, make your changes.
   a) Select a tracker type from the **Tracker** drop-down list to change the artifact's tracker type.
   b) Edit the artifact's title and description.
   c) Click **Save**.
6. On the **View Artifact** page, change any other fields if required, and click **Update**.

All updates to the artifact are recorded in the **Comments** section of the **Status/Comments** tab.

Edit a tracker artifact by email

To comment on a tracker artifact when you are not logged into TeamForge, send an email to the tracker, or reply to an automatic update email about the artifact.

You can also add a comment or an attachment to a tracker artifact that you are monitoring by responding to the monitoring email notification.

You can also use email to add an attachment. But to edit any other fields, you must make the changes in TeamForge.

You do not have to be logged into TeamForge to edit a tracker artifact using email, but you must have the tracker edit permission for the tracker containing the artifact you want to edit.

> ☞ **Tip:** Each comment in an artifact or task has a unique ID with its own URL. To link directly to a particular comment, copy that comment's URL and paste it into an email, a project page, or another comment. For example, to point to the third comment in artifact 12345, write `artf12345#3` in your comment. (If

the artifact or task you are linking to is on a different site, give the complete URL, like this: `http://mysite.com/sf/go/task1234#3`.)

Send an email message to <artifact id>@<CollabNet TeamForge server>.

👉 **Tip:** You can find the artifact ID on the **Artifact Details** page.
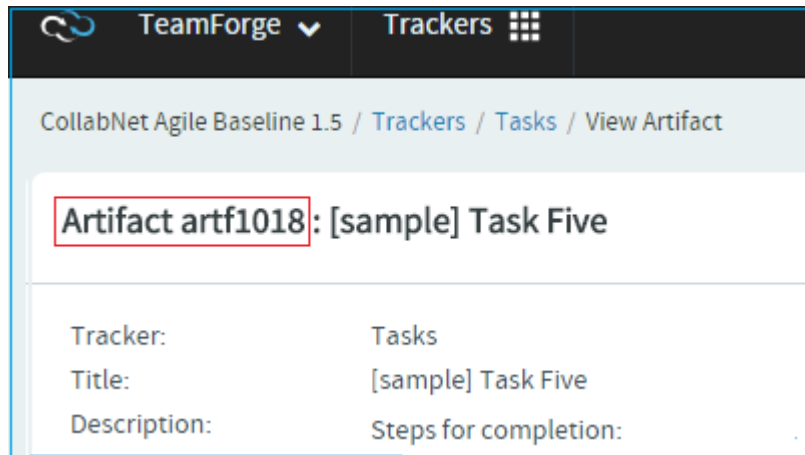
TeamForge maps your email to the tracker record like this:

Email field	Tracker field
To	Tracker email address
Subject	Artifact title
Body	Artifact description
Attachments	Attachments

Change multiple artifacts

When you have a large number of artifacts to update (for example, all the artifacts in a tracker, or a filtered list of artifacts in a planning folder), you can edit all the artifacts at once.

👉 **Note:** When you update two or more artifacts at a time, each user who is monitoring any of the changed artifacts gets a single email describing all the updates.

1. Click **TRACKERS** from the **Project Home** menu.

2. Go to the tracker or planning folder that contains the tracker artifacts you want to edit.

   👉 **Tip:** In a tracker, you can use the filter to help you find the desired artifacts.

3. Select the artifacts you want to edit, and click **Mass Update**.

4. For artifacts in a tracker, choose which ones you want to update.

   - **Selected** - Updates only the artifacts that you selected.
   - **Filtered Set** - Updates all artifacts returned by your filter, or all artifacts in the tracker if you did not apply a filter.

   👉 **Tip:** Choose **Filtered Set** when the artifacts span multiple pages and you want to select them all.

5. Make your changes and click **Update**.

   👉 **Important:** Some fields in your tracker may have values that depend on values in other fields, or use validation rules to ensure correct content. If your mass update operation breaks any such dependencies, you must fix the errors before running the mass update.

Edit multiple artifacts inline

When you have a list of tracker or planning folder artifacts to update, you can edit all the artifacts inline at once.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the **List Trackers, Planning Folders and Teams** page, click the tracker or the planning folder that contains the artifacts you want to edit inline.
3. In the **List Artifacts** page, click **Edit Inline**.
   To help you identify editable columns, all non-editable columns are disabled. You can see a hand symbol when you hover the mouse pointer over the editable columns.
4. Click a field in a column and edit the selected artifact.
   For example:

   - Clicking a field in the Assigned To column lets you edit the person assigned to the specific artifact.
   - Clicking a field in the Planned For column displays the **Planning Folder** dialog box to let you change the planning folder for the specific artifact.

5. When you are done, click **Save**.

Move a tracker artifact

As work proceeds on a tracker artifact, its focus may change. If this happens, it may be appropriate to move the artifact into a different tracker, or to a tracker in a different project.

To move a tracker artifact between projects, you must have either the tracker administration permission or tracker 'view, submit and edit' permissions in both the source and destination projects.

> ☞   **Important:** Tracker 'delete' permission is not required to cut and paste tracker artifacts.

You can move one or more tracker artifacts in the same operation.

1. Click **TRACKERS** from the **Project Home** menu.
2. On the list of project trackers, click the title of the tracker containing the tracker artifact that you want to move.
3. Select the tracker artifact (or artifacts) that you want to move, and Then click **Cut**.
   CollabNet TeamForge removes the tracker artifacts and places them on the clipboard.
4. Go to the tracker into which you want to paste the tracker artifacts.

   > ☞   **Note:** To move the tracker artifacts into another project, first find the destination project using your **Projects** menu.

5. Click **Paste**.

   - If all artifact assignees have the appropriate permissions in the destination tracker, the tracker artifacts are now moved.
   - If not, the **Paste Artifacts** window tells you how many tracker artifacts cannot be reassigned automatically.

6. Choose one of the following methods to reassign any unassigned tracker artifacts.

   - **Reassign Artifacts to:** - Assigns all unassigned tracker artifacts to the project member you select.
   - **Myself** - Reassigns all unassigned tracker artifacts to you.

7. Click **Next**.

The tracker artifacts are now moved to the destination tracker. The details of the move are recorded in the **Change Log** tab of the **Artifact Details** page.

- If the project members to whom the tracker artifacts are assigned have the appropriate permissions in the destination project, the tracker artifacts are automatically reassigned to the same project members. If not, you are offered several options for reassigning them.
- If both trackers use the same user-defined fields and field values, values in these fields are retained.
- If each tracker uses different user-defined fields or field values:

  - Values are set to the default value, if one was specified by the tracker administrator. Default values are mandatory for required fields.
  - Values are set to **None**, if no default value was specified by the tracker administrator.

Associate tracker artifacts with other items

You can connect tracker artifacts with other CollabNet TeamForge items such as documents or topics. Creating associations between items enables you to define relationships, track dependencies, and enforce work flow rules.

Associate a tracker artifact with a document, task, integrated application object, or forum

When a tracker artifact is related to other TeamForge items, such as tasks, documents, integrated application objects, or discussions, you can connect the tracker artifact to the other item by creating an association.

Creating associations between items helps you to define relationships, track dependencies, and enforce work flow rules.

1. Click **TRACKERS** from the **Project Home** menu.
2. On the list of project trackers, find the tracker artifact with which you want to create an association.
   Use the filter if needed.
3. Click the artifact title.
4. On the **View Artifact** page, click the **Associations** tab.
   The list of existing associations appears.
5. Click **Add**.
6. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:
   - **ENTER ITEM ID** - If you know the item's ID, you can enter it directly.

     ☞ **Note:**

     - To associate an object in an integrated application from within TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked.
     - Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.
   - **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
   - **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.
7. Click **Next**.
8. You may add a comment in the **ASSOCIATION COMMENT** text box.
9. Save your work.

   - Click **Finish and Add Another** to add additional associations.
   - Click **Finish** to return to the **Details** page.
10. Click the **Associations** tab to view a graphical representation of all the associated items.

    Through the Association Viewer, you can choose to view associations in the form of a list or flip over to the Trace view to explore the layers of associations (including parent/child dependencies) laid out in a timeline. You can scroll across the Trace view by dragging the mouse over the association layer or use the 'Previous' and 'Next' arrows to view all the objects as events in a timeline.

    While the Associations tab shows the count of the total number of associations, you can only view the most recent 500 associations when you click the Associations tab in case the artifact has more than 500 associations. You can, however, browse through the Association Viewer to view older associations.

    ☞ **Note:** To remind yourself of the details of the association later, look in the **Change Log** tab of the associated **View Artifact** page.

Associate a tracker artifact with a file release

To track the source and resolution of a bugs or a feature request, associate its tracker artifact with the file release in which it was discovered and fixed.

Tracker artifacts associated with file releases are also displayed separately, providing a simple way to track all issues that were discovered in or fixed in a specific file release.

☞ **Tip:** You can also add associations from the tracker artifact's **View Artifact** page.

1. On the **Project Administration** page, enable both the **REPORTED IN RELEASE** and **FIXED IN RELEASE** fields.

   • **REPORTED IN RELEASE** - When submitting a new artifact, the user can choose from a drop-down list of all releases in the project to identify the release in which the issue was discovered.

   • **FIXED IN RELEASE** - After the issue is fixed, the user can choose from a drop-down list of all releases in the project to identify the release in which the issue was fixed.

   ☞ **Tip:** If either of these fields says `Unknown`, the artifact you are working on may be associated with a file release that you don't have permission to view. You can leave that as it is or change it to a file release that you do have permission to view.

2. When submitting a new artifact, choose the release in which the issue was discovered from the **REPORTED IN RELEASE** drop-down list.

   The drop-down list shows all releases in the project, except those that are in **pending** status.

3. After the issue is fixed, update the artifact by choosing the release in which the issue was resolved from the **FIXED IN RELEASE** drop-down list.

   The drop-down list shows all releases in the project, except those that are in **pending** status.

The associated tracker artifacts are displayed on the **View Release** page, in the **REPORTED TRACKER ARTIFACTS** and **FIXED TRACKER ARTIFACTS** sections.

☞ **Note:** To remind yourself of the details of the association later, look in the **CHANGE LOG** tab of the associated **View Artifact** page.

Associate a tracker artifact with a code commit
When checking in files to your SCM repository, you can create links to one or more tracker artifacts or other CollabNet TeamForge items.

Associations track the links between code and the bugs, feature requests, or other tracker artifacts that the code addresses. You can also associate code commits with other TeamForge items, such as tasks or documents.

A project administrator can make associations mandatory for all code commits. When this is made mandatory, the following additional rules pertaining to code commit can also be set:

• Code commits can be performed only for open artifacts.
• To perform a code commit, the committer must be the owner of the specific artifact.

☞ **Note:** Once you enforce the above rules, validations are strictly enforced for commits against tracker artifacts only. In case you commit against any other TeamForge object, for example a wiki or a document, mere existence of the object ID ensures successful commit and association and no validations are performed against the status of the artifact or who it is assigned to.

You can create tracker artifact associations from whatever interface you normally use to check code into your SCM repository. You do not have to log into TeamForge.

Use the same syntax for commits to CVS and Subversion repositories.

   When making a code commit, add the associate command in the commit message like this:
   `[<item id>]`, such as the TeamForge tracker artifact ID or task ID.

   • TeamForge item IDs are always letters followed by four or more numbers, such as `task1029` or `artf10011`.
   • To associate a commit with multiple TeamForge items, separate the item IDs with commas.
   • All associations are displayed in the **ASSOCIATIONS** tab of the **Commit Details** page.
   • The **Comment** section lists the comments made with each commit.

   ☞ **Note:** To associate an object in an integrated application, use the `[<prefix_objectid>]` format. Each integrated application displays its prefix on moving the mouse over the application name.

☞ **Tip:** To remind yourself of the details of the association later, look in the **CHANGE LOG** tab of the associated **View Artifact** page.

Make a tracker artifact depend on another artifact

To organize your work with tracker artifacts, you can make an artifact a "child" or a "parent" of another artifact.

- A child artifact can have only one parent.
- A parent artifact can have any number of children.
- A parent artifact cannot be closed if a child is open.

☞ **Note:** "Open" and "Closed," in this context, refers to a type of status, not the status itself. A tracker administrator can specify a group of statuses, such as "Deferred," "Fixed," or "Rejected," as equivalent to "Closed," while "In progress" and "Under consideration" might be specified as "Open" statuses. For example, when you look at the artifact summaries at the top of any tracker list page, you are seeing a summary of the status type, not the status, of the artifacts.

1. On an artifact page, click the **DEPENDENCIES** tab.
2. Choose or create the related artifact.

   - If the parent artifact already exists, click **Choose Child** or **Choose Parent**. In the pop-up, type in the artifact ID or choose from the list of your recently edited artifacts.
   - If the related artifact does not exist yet, click **Create Child in** or **Create Parent in** and select the tracker that the new related artifact will belong to. Fill in the form the same way as you would for submitting an unrelated artifact.

     ☞ **Note:** If **Choose Parent** or **Create New Parent** is not visible, the artifact already has a parent artifact. An artifact can only have one parent artifact.

3. Click **Next**.
4. Write a comment that describes the relationship, if appropriate, and click **Finish**.

The parent-child relationship between the artifacts is established.

- To cancel a parent dependency, click **Remove**.
- To cancel a child dependency, select the child artifact and click **Remove**.

☞ **Note:** Task dependencies and tracker artifact dependencies are different things.

   - For tracker artifacts, an artifact with dependencies (a "parent" artifact) can't be considered closed unless all of its dependent artifacts ("children") are closed.
   - For tasks, a dependency means one task can't start until another task is completed.

Quickly create dependencies, associations or add attachments

Use the context menu available in planning folder List View to quickly choose a "parent" or "child", remove a "parent", add associations, add attachments or clone artifacts.

In the planning folder List View, you can see a down arrow icon next to the artifact ID (as shown in the following screenshot) when you move your mouse over artifact rows.

Down arrow icon
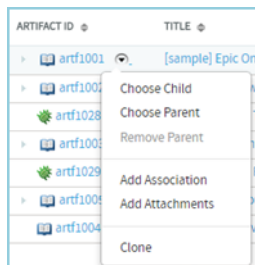
A context-sensitive menu pops up on clicking the down arrow icon.

Context menu if an artifact has a parent
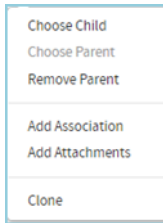
Context menu if an artifact has no parent
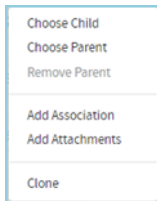
To quickly choose a parent or child, add associations and attachments, or remove a parent:

1. Select a planning folder on the **List Trackers, Planning Folders and Teams** page.
   The planning folder **List Artifacts** page appears.

2. Move your mouse over an artifact's ID.
   A down arrow icon appears.

3. Click the down arrow icon to see the context menu. Depending on the context, you can do one of the following tasks.

Choose a child

4. Click **Choose Child** from the context menu.
   The **Selecting Children...** window appears.

5. Select the **Enter Artifact ID** option and type an artifact ID. This artifact becomes the child.

   ☞ **Tip:** You can also select the **Add from Recently Edited** option and choose a child from the list of recently edited artifacts.

6. Click **Next**.

7. You may add a comment in the **DEPENDENCY COMMENT** text box.

8. Click **Finish**.

   ☞ **Note:** You may also click **Finish and Add Another** to continue adding more children for the same artifact.

Choose a parent

9. Click **Choose Parent** from the context menu.
   The **Selecting Parent...** window appears.

10. Select the **Enter Artifact ID** option and type an artifact ID. This artifact becomes the parent.

    ☞ **Tip:** You can also select the **Add from Recently Edited** option and select a parent from the list of recently edited artifacts.

11. Click **Next**.

12. You may add a comment in the **DEPENDENCY COMMENT** text box.

13. Click **Finish**.

Remove a parent

14. Click **Remove Parent** from the context menu.
    A confirmation message appears: `Do you want to remove the dependency?`.

15. Click **OK**.

Add associations

16. Click **Add Association** from the context menu.

The **Add Association Wizard** appears.

17. Select the **Enter Item ID** option and type an artifact ID for creating an association.

👉 **Tip:** You can select the **Add from Recently Edited** option and select an artifact from the list of recently edited artifacts.

18. Click **Next**.

19. You may add a comment in the **Association Comment** text box.

20. Click **Finish**.

👉 **Note:** You may also click **Finish and Add Another** to continue associating more artifacts.

Add attachments

21. Click **Add Attachments** from the context menu.
    The **Add Attachments** window appears.

22. Type a comment for the attachments in the **COMMENT TEXT** box.

23. Click **Choose File**.

24. Browse and select the file you want to attach.

25. Click the **Attach another file** link to add more attachments. Repeat this step for adding more attachments.

26. Click **Add** to attach the selected files to the artifact.

Clone an artifact

27. Click **Clone** from the context menu.
    The **Clone Artifact** window appears.

28. Provide a name and description for cloning the artifact.

29. Click **Clone** to clone the artifact.

Schedule work on an artifact

To plan how and when an artifact is to be addressed, assign it to a planning folder.

👉 **Note:** A planning folder can contain artifacts from many different trackers. When you assign an artifact to a planning folder, the artifact is still in the tracker where it was created.

1. Click **TRACKERS** from the **Project Home** menu.

2. Find the tracker that contains the artifact you want to assign, and select the artifact.

    You can select any number of artifacts at once.

3. Click **Plan for** and select a planning folder.

The planning folder you assigned is now shown in the **Planned For** column on the **Planning Folder Summary** page.

You can always reassign an artifact to another planning folder as conditions change. Any effort data you have provided is recalculated automatically.

👉 **Note:** When a tracker is disabled, artifacts from that tracker do not contribute to the effort totals calculated for any planning folder they are in.

👉 **Tip:** When you are working with a single artifact, you might prefer another way to assign it to a planning folder: open the artifact you want to assign and look for the **Planning Folder** field. Select a planning folder for the artifact and save the artifact.

Set up trackers

A tracker is a collection of related artifacts that describe work to be done or issues to be resolved. Every project should have one or more trackers.

When you start a tracker, you decide which fields will be used, who will use them, and how they will use them.

Create a tracker

Create a tracker whenever you need to report and track bugs, feature requests, support requests, or any other type of issue where ownership, status, and activity must be managed.

Individual tracker entries are referred to as tracker artifacts. A tracker is a set of tracker items with a common purpose, such as bug reports, feature requests, or tasks.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. Click **Create**.
4. On the **Create Tracker** page, provide a name and description for the tracker.

   ☞ **Tip:** Descriptions help users learn how best to provide the information you want from them. To maximize your chances of getting useful data, make your description as informative as you can.

5. Select an icon that suggests the type of work the tracker is handling.

   This icon will appear with any artifact in this tracker, wherever it is viewed on the site. For example, if someone brings an artifact from this tracker into a planning folder, users of the planning folder can glance at the artifact's icon to see where it comes from.

6. Select the relevant unit from the **DISPLAY EFFORT IN** field. The units displayed here are configured based on the size of the artifacts in the tracker. Eg. Select the unit as **HOURS** for a tracker of small defects, **DAYS** for a tracker of tasks, and **WEEKS** for a tracker of epics.

   ☞ **Note:** Configure the units at the project level and not at the planning folder level.

7. Select **INCLUDE FOREIGN CHILDREN** to include points and efforts from children artifacts across the projects in TeamForge.

   ☞ **Note:** In a parent artifact, enabling **CALCULATE POINTS** field sums and rolls up the points from all its children artifacts within the project. In this total, if you want to include children artifacts from other projects across TeamForge, have the **INCLUDE FOREIGN CHILDREN** option enabled.

8. Click **Create**.
   The new tracker appears at the bottom of your list of trackers.
9. If necessary, drag the tracker to a place in your tracker list that makes sense.

   The order you set here controls the order of every tracker list in your project.
10. You'll probably need some custom fields to capture information that's specific to your project.

    See *Create your own tracker fields* on page 38.
11. To speed up the team's work, you may want to set up some rules for automatically reassigning artifacts when their contents change.

    See *Create a tracker work flow* on page 40.

Clone a tracker
To save efforts in duplicating a tracker within the project and across the projects, choose cloning options available in TeamForge.

You can clone a tracker within a project along with the workflow. This can be done across the projects as well. If you are the Project Admin for the source and destination projects, you can create new roles that are not available in the destination project. However, the permissions associated with the role are not copied from the source project. A tracker admin cannot create new roles while cloning a tracker across projects.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. From the **Project Admin Menu,** click **Tracker Settings**.
3. Select **Clone External Tracker** from the drop-down list available in **Clone** button to clone a tracker from the source project.

   ☞ **Note:** To clone a tracker within the project, you can select the tracker from the **Tracker Settings** page and click **Clone Tracker**.

4. On the **Cloning Tracker** page, name the new tracker and enter a suitable description.
5. Enter **SOURCE TRACKER ID** of the tracker available in the source project and click **Next**.
   The **Cloning External Tracker** page appears.
6. On the **Cloning External Tracker** page, name the new tracker and enter a suitable description.

7. Click **Create**.

The cloned tracker appears at the bottom of your list of trackers.

Enable or disable fields

If a tracker field is disabled, it does not appear on the **Artifact** page. Most fields can be disabled.

Disabled fields are accessible only to tracker administrators on the **TRACKER FIELDS** tab.

You can enable or disable any field that is user-defined or configurable. However, you can't disable all configurable fields. For example, the **Status**, **Priority**, **Category,** and **Planning Folder** fields can't be disabled.

☞ **Note:** If your goal is to prevent users from entering data into a field when submitting an artifact, but still display the field on the **Edit Artifact** and **Tracker Search** pages, don't disable the field. Instead, clear the **DISPLAY ON SUBMIT** option on the **Edit Tracker Field** page.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of current trackers, click the tracker you want to configure.
4. On the **TRACKER FIELDS** tab, select the fields you want to enable or disable.

   • Click **Disable** to remove them from the **Artifact** page.
   • Click **Enable** to allow them to be configured and displayed.

   ☞ **Note:** Data in disabled fields is still search-able, but disabled fields do not appear as inputs on the **Search** pages.

Require users to complete a tracker artifact field

If a field is set as required, users cannot submit artifacts without completing it. Most tracker fields can be required or optional.

Each tracker can have its own required and optional fields. Required fields are marked with a red asterisk * on the **Submit Artifact** page.

☞ **Note:** When you make a field required, any field whose values depend on that field's values is also required. See *Help users select options in a tracker artifact* on page 118 for more about dependent field values.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of current trackers, click the one you want to configure.
4. On the **Tracker Fields** tab, click the name of the field you want to set as required or optional.

   By default, only the **Title**, **Description**, , and **Status** fields are set as required.

5. On the **Edit Field** page, select or clear the **Required** option to make a field required or optional.

   Required fields automatically show up on the **Submit Artifact** page.

   ☞ **Note:** System-defined fields and the **Status** field are always required.

6. For optional fields, select or clear the **DISPLAY ON SUBMIT** option.

   This specifies whether the field will appear on the **Submit Artifact** page.

7. Click **Save Field**.

Create your own tracker fields

To track data that is not captured by the default set of fields, create new fields that fit your project's purposes.

You can create the following user-defined fields in each tracker:

• Up to 30 text entry fields.
• Up to 30 date fields.
• Up to 30 single-select fields.
• An unlimited number of multiple-select fields.

⚠️ **Caution:** Creating a large number of user fields and multiple-select fields may affect site's performance.

Configure tracker "Select" field values

To help users provide meaningful information, supply them with useful field values to choose from in the input fields in the tracker entry form.

👉 **Tip:** Once a tracker has been created, you may:

- Create one or more user-defined single-select or multiple-select fields.
- Add predefined values to the fields. Remove values, if required.
- Enable or disable fields.
- Change the default values for fields.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the name of the tracker that you want to configure.
4. On the **TRACKER FIELDS** tab, click the name of the field whose values you want to edit.
5. On the **Edit Field** page, set up the field values you want users to see when they create a tracker artifact.

   - To define a new value, click **Add**.
   - To rename a value, edit the existing text. If you rename a value, the value is renamed in all existing artifacts.
   - To remove a value, check the box and click **Delete**. If you delete a value, the value is changed to **None** in all existing artifacts.
   - Select **DEFAULT VALUE** to set which option will be chosen if the user makes no selection. When you move a tracker artifact from one tracker to another, the default field value is the value that comes along.

   👉 **Note:**

      - When you edit the values of the **Status** field, you are also asked to describe what each value's status means, as shown in the **Values** section of the **Edit Field** page. This status meaning is used in **Advanced Search** to define which values are returned when searching for open or closed artifacts.
      - As always, when you create a new tracker, the default value for the '**Priority**' field is set as '4 - Low'. However, you can change the default value by editing this configurable single select field, '**Priority**'. You cannot delete or disable the **Priority** tracker field.
      - When you change the tracker fields, the values in the existing artifacts remain unchanged.

6. Click **Move Up** or **Move Down** to order the list the way you want it.
7. Click **Save Field**.

All values are now available in the selection menu for the field.

Assign a tracker artifact automatically

You can configure the tracker to automatically assign newly submitted artifacts to specific project members.

You can assign artifacts to individuals based on the values in the **Category** or **Release** field.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the tracker you want to configure.
4. On the **AUTO ASSIGNMENT** tab, select **Auto Assign By Category** or **Auto Assign By Release**.
5. For each category or release value, choose a project member from the **AUTOMATICALLY ASSIGN TO** drop-down menu.

   This menu contains all project members with the tracker edit permission. Click the Search icon to display a list of project members to whom you can auto-assign artifacts. Choose **None** if you do not want artifacts with a specific value automatically assigned.

6. Click **Save**.

Whenever a new artifact is submitted with a value in the relevant field, the assignee receives an email notification and the artifact appears on the asignee's **My page**.

Change a tracker

A tracker's real-world uses often outgrow the name or description you gave it when you created it. When that happens, it's a good idea to update the tracker to reflect its changing role.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the tracker you want to edit, and click **EDIT**.
4. On the **Edit Tracker** page, provide a new name or description for the tracker, and update the icon.
5. Update the units from **DISPLAY EFFORT IN** and click **Save**.

   ☞ **Note:** These units are configured in the **Units** page at the project level, and not at the planning folder level.

6. Select **INCLUDE FOREIGN CHILDREN** to include points and efforts from children artifacts across the projects in TeamForge.

   ☞ **Note:** In a parent artifact, enabling **CALCULATE POINTS** field sums and rolls up the points from all its children artifacts within the project. In this total, if you want to include children artifacts from other projects across TeamForge, have the **INCLUDE FOREIGN CHILDREN** option enabled.

7. If necessary, drag the tracker to a place in your tracker list that makes sense.

   The order you set here controls the order of every tracker list in your project.

Create a tracker work flow

To channel project members' work on tracker items, set up rules for how a tracker item can move forward.

Before creating a tracker work flow, see that these criteria are met:

- You have a tracker to work with.
- The tracker has a set of statuses defined, such as "In progress" and "Ready for QA."
- Roles exist, and you can assign project members to them.

A work flow is a sequence of changes from one status to another. You can define status transitions for any combination of tracker statuses in the tracker.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, select a tracker.
4. Click the **WORKFLOW** tab.
   The **Workflow** page lists all of your status values and the tracker workflow that you have configured.
5. On the **Workflow** page, click the status value for which you want to create a workflow.
6. On the **Edit Field Transition** page, select a status value from the **Create Transition to Status** drop-down menu.

   ☞ **Note:** By default, the **Any** workflow is shown. This means that you are defining workflow rules for the transition from the status that you selected to *any* other status.

7. Click **Add**.
   A new work flow is added. The **Any** work flow is changed to **Remaining Statuses**.
8. In the **ROLES TO MAKE THIS TRANSITION** section, specify which users can make this change.
   For example, only users with the QA Engineer role are allowed to change artifacts from **Open** to **Cannot Reproduce**.
9. In the **REQUIREDFIELD(S) FOR STATUSTRANSITION** section, select the fields for which the user *must* provide values.
   For example, the user must assign the tracker item to someone and enter a comment.

   ☞ **Note:**
   - Fields that are already required do not appear in this list.
   - Fields whose values depend on a required parent field are automatically required. (See *Help users select options in a tracker artifact* on page 118 for more on parent and child fields.)

10. Click **Save**.

The work flow is now saved. When a user submits or edits the status of a tracker artifact, he or she sees only the options that are allowed by the work flow.

Export tracker artifacts

To use the contents of artifacts from a tracker or a planning folder in other applications, export them to a `.csv`, `.xml`, or tab-delimited file.

For example, sometimes it can be useful to look at the status of a project by sharing a spreadsheet using Microsoft Excel or Google Spreadsheets.

☞   **Tip:** You can also export tracker artifacts from a tracker report or a set of search results.

1. Click **TRACKERS** from the **Project Home** menu.
2. Click a tracker or planning folder.

    In a tracker,

    •   you can export a subset of the artifacts in the list view by setting up a filter. See *View the right tracker artifacts* on page 22.
    •   You can select specific artifacts and export only those checked items.

    The current sort order in the list view is retained in the exported list of artifacts as well.

3. Click **Export**.
4. In the **Export Data** window, select an export format that you can import into the other application.

    For example, to use the data in a spreadsheet program, select `.csv`.

5. Move the fields you are interested in from the **AVAILABLE COLUMNS** list to the **SELECTED COLUMNS** list, then click **Export**.

    ☞   **Note:** When you export tracker artifacts directly from a tracker, you can choose from all the fields in the tracker. When you export from a planning folder, only the standard fields provided by TeamForge can be exported.

    For more detail on what you can do with different kinds of fields, see *What fields can I use in a tracker?* on page 1017

6. In the **Export Data** confirmation window, click the link to save the file where you want it.

Boards

You can start using the task and kanban boards, once your project administrator has configured them.

Use the Task Board

During a sprint, TeamForge project members can use the Task Board to view tasks, create tasks for backlog items, edit tasks and drag and drop tasks across swimlanes as they progress.

With the Task Board, you can:

•   View backlog items of a selected planning folder in the **Backlog Items** swimlane, which is the leftmost swimlane of the task board.
•   View tasks (belonging to backlog items) pinned to swimlanes based on the status.
•   Edit backlog items and tasks by clicking the artifact ID link.
•   Move tasks (cards) from one swimlane to the other as tasks progress.
•   Add new tasks for backlog items.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the **List Trackers, Planning Folders and Teams** page, click **TRACK**.

    A Task Board for the current project context is displayed.
3. Select a planning folder from the drop-down list.

    The Task Board is populated with backlog items and their tasks.

- If a selected planning folder has subfolders, the artifacts within the subfolders are not displayed.
- Cards have a color-coded bar to visually identify the priority. In addition, the open and closed cards' background is color-coded to uniquely identify the status.
- For each card, the artifact ID, title, priority, assigned to and effort (in terms of 'Points' for backlog items and 'Hours' for tasks) are displayed. Status information is shown on backlog cards only. Relevant tooltips appear when you hover your mouse over these data elements. You can also hover your mouse over the tracker icon in backlog and task cards to know the tracker name.

Add new tasks

4. Click the '+' icon.

The **Create Task** window appears.

5. Type a title and description and click **Submit**.

☞ **Tip:** To use the title you type as the description, select the **Use Title as Description** check box.

To edit a backlog item or task using the task ID link:

6. On the **List Trackers, Planning Folders and Teams** page, select the planning folder.

7. From the listed backlog items and tasks, click the artifact ID link you want to edit.

The **View Artifact** page appears.

8. Modify the backlog item or task and click **Update** to update the artifact and return to the Task Board.

To quickly edit a task

9. Click the edit artifact button in a task. The **Edit Task** window appears.

☞ **Note:** To be able to edit tasks, you need to have the required permissions.

Edit task button

Edit Task window

10. You can edit the following on the **Edit Task** window:

- **Artifact title**
- **Artifact description**
- **Primary and secondary attributes**

   You have two expandable frames, **Primary Attributes** and **Secondary Attributes**, that list fields you can edit. Primary attributes include Priority, Status, Assigned To, Estimated Effort, Remaining Effort, Actual Effort, Points and Team. The expandable frames and the fields listed in them depend on your application configuration that you set in Tracker Settings. For example:

   - Configurable fields which are enabled. For more information, see *Enable or disable fields* on page 38.
   - User-defined fields (flex fields) with the field type Text Entry and Single Select and set as **Required** (Tracker Settings > Tracker Administration > Create Field). For more information, see *Create your own tracker fields* on page 38.
   - Fields which are workflow configured. For more information, see *Create a tracker work flow* on page 40.

   By default, the following field types, which you select while creating a user-defined field (Project Admin > Tracker Settings), are non-editable from the task board's **Edit Task** window: Multiple select (**Multiple Select**), Date picker (**Select Date**) and User picker (**Select User(s)**).

- **Comments**

In this expandable frame, in addition to adding comments, you can view the last five comments for an artifact.

11. After editing an artifact, click **Update**.

Move tasks across swimlanes

12. To move tasks across swimlanes, drag a task card from the source swimlane and drop it on the destination swimlane.

☞ **Note:** You need to have task tracker 'edit' permission to move cards across swimlanes.

View both 'open' and 'closed' backlog items

13. ☞ **Note:** When you select a planning folder, the Task Board shows only 'open' backlog items by default.

Click the 'Show/hide closed artifacts' toggle button to view both 'open' and 'closed' backlog items.

Show/hide closed artifacts toggle button

Refresh the Task Board

14. Click the following 'Refresh' button to refresh the Task Board.

Auto assign task

If a task is not assigned to anyone, that is, if it is assigned to **None**, then it can be assigned to the user who has logged in using the **Auto Assign Task to Me** feature. This check box appears only if:

* The Task Board is configured.
* The user who logged in has the edit permission to the task tracker.
* The project is not locked.

15. Select the **Auto Assign Task to Me** check box and drag the artifact from one swimlane to another (from one status to another) and the artifact is automatically assigned to you.

Position artifact cards

16. If you have the requisite permission, you can drag and drop a card above or below the other cards within a swim lane to position it in the order you want the artifact addressed.

☞ **Note:** The positioning of cards is retained when you switch between the list and board views (planning, task and kanban).

For related information on ranking artifacts in the list view, see *Reorder the contents of a planning folder* on page 126.

Use Kanban Board

After the kanban board is configured, you can use it to view, plan and track work items for the selected planning folder or the selected team within that planning folder.

1. Click **TRACKERS** from the **Project Home** menu.

2. In the **List Trackers, Planning Folders and Teams** page, click **Kanban**.

3. Select the required kanban board from **Manage Boards** (available only for project administrators). If you are a project member, select your kanban board from the **Select Board** list.
The selected kanban board for the current project context is displayed.

4. Select the planning folder for which you want to view the status of the artifacts.

If your project contains teams, the **Select a Team** drop-down list appears. From this drop-down, select **All artifacts** or if you want to view artifacts specific to a team within the selected planning folder, select the relevant team.

Depending upon your selection, artifacts pertaining to the mapped trackers are displayed in the appropriate swimlane.

- You can view a maximum of 5 swimlanes at a time on your Kanban Board. If there are more, use the carousel scroll to slide across the swimlanes. You can expand or collapse each swimlane. When there are no results to display in a swimlane, the **Expand/Collapse** button does not appear.
- Based on the configuration values, if the constraints are violated, the relevant status headers are highlighted appropriately. A violation of minimum constraint is highlighted in orange indicating that resources are underutilized whereas that of maximum constraint is highlighted in red indicating resources being overloaded and bottlenecks to be fixed. For more information on how a kanban board is configured, see *Set up Kanban Board* on page 142.

- Each swimlane header displays the status label, the total number of artifacts within the selected planning folder, and the minimum and maximum constraints that were configured for the kanban state.

In the above scenario, for the 'In Progress' status:

'3' is the total number of artifacts in the 'In Progress' status within the selected planning folder.

'2' indicates the minimum constraint and '*' indicates the maximum constraint '-1'. (-1 as the minimum constraint translates to 0 whereas -1 as maximum indicates there is no limit and so represented by an *.)

> 👉 **Important:** These values and constraint validation are planning folder specific and not team specific. For example, in the above scenario, the total number of artifacts is the overall count of artifacts which are in the 'In Progress' status within the selected planning folder and not within the selected team.

- Each artifact card displays its points (story points) or estimated effort if these fields are enabled in your tracker settings. If both are enabled, only the estimated effort is displayed hovering the mouse on which the artifact's points is displayed. Similarly, if all the effort fields (estimated, remaining and actuals) are enabled, you can view them by hovering the mouse on the estimated effort.
- Cards have a color-coded bar to visually identify the priority. In addition, the open and closed cards' background is color-coded to uniquely identify the status.
- A parent artifact card displays the count of its child artifacts (**Show/Hide child artifacts** button). Click this button to view or hide the child artifacts. If there are more than 5 child artifacts, when you expand the parent artifact, you will see pagination arrows ('Previous'/Next') at the bottom of the parent artifact card.

- If the tracker types of both the parent and child artifacts have been mapped with the kanban statuses, the child artifacts appear within their parent artifact card and also as an individual card. In the above scenario, Epic (parent artifact) and Story (child artifact) trackers have been mapped to the kanban status 'In Progress'. Story 'artf1014' is the child artifact of the epic 'artf1002'. So you will see 'artf1014' within the parent artifact card as well as outside of it as an individual artifact card.
- When a child artifact is closed, a 'Closed' tag appears next to the child artifact ID within the parent artifact card.

Edit artifacts

Use Kanban Board to edit an artifact using the Edit icon on the artifact card or move artifacts from one status to another appropriately. Based on the status changes you make, the swimlane headers get updated appropriately.

☞ **Remember:**

- You can edit an artifact only if you have the tracker edit permission. Otherwise, you can only view it.

5. When you drag the artifacts from one swimlane to another, note the following validations:

- The changes you see in swimlane headers do not apply to any specific team but takes into account the total count of artifacts within the selected planning folder.
- You cannot move an artifact card to a status that is not mapped to that particular tracker type. For example, a 'Ready for QA' kanban status may not have been mapped to any of the epic tracker statuses. So when you attempt to move an epic to that status, you will get an `Invalid status configuration` error.
- You can move a parent artifact to the 'Closed' status only if all its child artifacts are closed. However, a child artifact is independent of its parent with regard to the status change, that is, when you move a parent artifact from one status to another, the status of the child may still remain unchanged. For example, an epic may have

many stories, the status of some may change from 'Not Started' to 'In Progress' whereas some may not. So when the status of an epic as one single unit may change, it need not apply to all of its child artifacts.

- When your move violates a constraint (minimum or maximum), a warning is displayed; but you can still move the artifacts.
- If a kanban status is mapped to more than one tracker status, when you move an artifact, you have an option to choose a status as shown in the following screen shot.

Create artifacts and child artifacts

Using the Create Artifact icon on the top right of your kanban board, you can quickly create an artifact. The Create Artifact icon appears only when you configure a kanban board and select a planning folder.

Similarly, using the Create Child Artifact icon available on the artifact card, you can quickly create a child artifact for any artifact card displayed on Kanban Board.

☞ **Note:** The Create Child Artifact icon does not appear on closed artifact cards.

You can create an artifact or a child artifact only if you have the required tracker permission.

6. Click the required icon: Create Artifact or Create Child Artifact.

7. Enter the required information in the relevant window and click **Submit**.

- Only trackers configured for the kanban board appear in the Trackers drop-down list.
- While you quickly add artifacts with data for just three fields such as the tracker type, title and description, the artifacts are, however, saved with default values for other required fields, which you may choose to update later. If the default state of the selected tracker is not mapped for the newly created artifact or child artifact, the artifact card does not show up on the kanban board.

Show / Hide closed cards

8. When you have a large volume of closed cards in a planning folder, you can restrict the number of closed cards you want to view by toggling between the 'Show all closed artifacts' icon:

Or the 'Hide artifacts older than 60 days' icon using which you can hide closed cards older than 60 days. This is the default option:

☞ **Note:**

- The Show/Hide toggle icon appears only on configured kanban boards.
- Your selection is saved for the subsequent sessions as well.

Position artifact cards

9. If you have the requisite permission, you can drag and drop a card above or below the other cards within a swim lane to position it in the order you want the artifact addressed.

> ☞ **Note:** The positioning of cards is retained when you switch between the list and board views (planning, task and kanban).
>
> For related information on ranking artifacts in the list view, see *Reorder the contents of a planning folder* on page 126.

Work on tasks

Manage your project plan by defining units of work and assigning them to project members.

Create a task

Create a task when you identify a unit of work that you want to track to completion as part of your project.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, go to the task folder in which you want to create the task.
3. Click **Create**.
4. On the **Create Task** page, provide a name and description for the task.
5. Assign the task a priority between 1 (highest) and 5 (lowest).
6. Assign the task to an individual.

   Click the people icon to list the project members to whom you can assign the task.
7. Assign the task a status.

   - Not Started
   - OK
   - Warning
   - Alert
   - Complete
8. Give the task a start and end date.

   Click the calendar icon to select dates.
9. In the **ESTIMATED HOURS** field, write the time you think is needed to complete the task.
10. Specify whether the time estimate is based on a five-day or seven-day calendar.

> ☞ **Note:** Choose a seven-day calendar to assign a start or end date that falls on on a weekend.

11. Click **Save**.

Edit a task

Tasks often change assignee, start and end dates, and dependencies before they are finished. You can update all those aspects of a task, along with its status.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, find the task folder containing the task you want to edit.
3. Click the task name.
4. On the **View Task Details** page, click **EDIT**.
5. On the **Edit Task** page, make your changes and click **Save**.

   If you change the end date of a task that has dependent successor tasks, you may want to change the start and end dates of the successor tasks. For example, if a successor task normally starts on a Friday, and is moved out two days by a change in a predecessor task, its new start date is Sunday (if it uses a seven-day calendar) or Tuesday (if it uses a five-day calendar).

   - To change the start and end dates of all successor tasks, click **Move Dependent Task Dates Forward/ Backward x Days**. All dependent task dates are moved forward or backward by the same number of days that the original task end date was moved.

- To apply the date change to only the selected task, without changing the start and end dates of any successor tasks, click **Do Not Change Dependent Task Dates**.

- If your end date change impacts successor tasks that you do not have permission to view, these tasks are not changed.
- If your end date change impacts successor tasks that you have permission to view, but not to edit, these tasks are marked for change approval by an administrator. The start and end dates for tasks that you have permission to edit are changed.
- Successors of tasks that are marked for change approval are not changed until approved by an administrator.
- From this page, you cannot change the start and end dates of successor tasks selectively. You must change all or none of them. To edit the start and end dates of selected successor tasks only, select **Do Not Change Dependent Task Dates**, then edit each selected task manually.

Find tasks

Use the task manager to find a specific task or set of tasks quickly. You can filter tasks by predefined fields or search by keyword, date or other values.

Search for tasks

Use the Search Tasks option to refine your search if task filtering returns too many results.

1. Click **TASKS** in the project navigation bar.

2. On the **Task Summary** page, click **Search Tasks**.

3. On the **Task Search Criteria** page, enter the keywords for the search.

   ☞ **Tip:** Wildcards are allowed.

4. Search by the type of content that makes sense to you.

   - Search for keywords in the **Task Name** field.
   - Search for keywords in the **Task Description** field.
   - Search the task progress fields: **Accomplished**, **Planned**, and **Issues**.
   - If you are searching for a specific task, and you know its task ID, enter the task ID.
   - To search by date, select date ranges for start date or end date.
   - To search by project member, select one or more people.

     ☞ **Note:** Click the people icon to display a list of project members with permissions to edit or create tasks.
   - To search by location, select the Task Folder to search.
   - To search by Status and Priority, select those.
   - Click **Search**.

See the right tasks

Filtering and sorting allows you to find a task or group of tasks using a number of pre-defined fields.

The filter option is available on every item list view in TeamForge.

1. Click **Tasks** in the project navigation bar.

2. On the **Task Summary** page, click the title of the task folder that contains the tasks you are interested in.

3. Specify the filter criteria in one or more filter fields (at the top of each column) and click **FILTER**.

   - You can find a filter field at the top of each column in most of the tables in the TeamForge application.
   - The filter field could be a text box or a drop-down list with multi-select check boxes.

- You can type your filter criteria in the text boxes. The search text is case-insensitive.
- You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the *FILTER_DROPDOWN_MAX_SELECTION* token in the `site-options.conf` file to increase or decrease the count.
- **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing "R" instantly shows all statuses having the alphabet "R". The search text is case-insensitive.

- Some search filters may not appear if your site administrator has not enabled them.

4. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.

5. Use the up-down arrow at the top of any column to sort your list by that column.

   - Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
   - Click the up-down arrow again to reverse the sort order.

Check your assigned tasks

It's a good practice to regularly look at all of your assigned tasks in the **My Tasks** section of your **My Page**.

Your **My Page** provides a central location where you can view and manage all tasks assigned to you from all the projects that you are a member of.

☞ **Note:** To see tasks assigned to you on a specific project, look at the **Tasks** section of that project.

1. Go to **My Page**.
2. If you have more than 20 tasks, click **View All** to list all of them.
3. Specify the filter criteria in one or more filter fields (at the top of each column) and click **FILTER**.

   - You can find a filter field at the top of each column in most of the tables in the TeamForge application.
   - The filter field could be a text box or a drop-down list with multi-select check boxes.

- You can type your filter criteria in the text boxes. The search text is case-insensitive.
- You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the *FILTER_DROPDOWN_MAX_SELECTION* token in the `site-options.conf` file to increase or decrease the count.
- **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing "R" instantly shows all statuses having the alphabet "R". The search text is case-insensitive.

- Some search filters may not appear if your site administrator has not enabled them.

4. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.

5. Use the up-down arrow at the top of any column to sort your list by that column.

   - Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
   - Click the up-down arrow again to reverse the sort order.

6. Click the task name to see the task details.

Update task status

When you have completed some work on a task assigned to you, you must update its status.

If you don't mark a task as completed, you may get an automatic reminder from the task's owner, its administrator, or other project members.

You can provide task status reports for all of your assigned tasks from a single page.

☞ **Note:** If you want to enter a status update for only one task, you can do so from the Task Progress section of the **Task Details** page.

1. Click **Tasks** in the project navigation bar.

2. On the **Task Summary** page, click the **MY TASK PROGRESS** tab.

   Your **My Task Progress** page displays a list of all tasks assigned to you.

3. Enter your task status updates in the **ACCOMPLISHMENTS**, **PLANNED**, and **ISSUES** fields.

   - **Accomplishments** - Record any accomplishments since your last update.
   - **Planned** - Describe work you plan to do on the task.

- **Issues** - Report any issues with the task.

4. If applicable, change the task status.

   ☞ **Note:** Project administrators may require approval for changes to task status. If change approval is required, you are notified when you submit your status report.

5. Save your changes.

   - Click **Save Page** to finish and save your changes.
   - If your tasks cover more than one page, click **Save Page and View Next** to proceed to the next page.

Schedule a task before or after another task

When one task cannot be done unless another task is done first, set up a dependency to track the relationship between the two tasks.

For example, if the deadline for a task changes, the tasks that depend on that task can automatically be rescheduled to reflect the change.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, find the task folder containing the task to which you want to add dependencies.
3. Click the task name.
4. On the **View Task Details** page, click the **DEPENDENCIES** tab.
5. On the **Task Dependencies** page, click **Add** in the **Predecessors** or **Successors** section.
6. In the **Add Task Dependency Wizard** window, choose a task.

   - **ENTER TASK ID** - If you know the task ID, you can enter it directly.
   - **ADD FROM HISTORY** - The History is a list of tasks that you have submitted or edited.

   ☞ **Note:** To create multiple dependencies, enter multiple task IDs separated by commas, or use the **ctrl** key and select multiple tasks from the history list.

7. Click **Next**.
8. On the **Add Task Dependency Wizard** that shows the dependency created, click **Finish**.

The **Predecessor** or the **Successor** section lists out the dependencies you have added.

View task dependencies

Task dependencies allow you to establish relationships between tasks. Task dependencies show what must be done before a given task and what will be done after the task is complete.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, find the task folder containing the task that you want to view.
3. Click the task name.
4. On the **View Task Details** page, click the **DEPENDENCIES** tab.

For each predecessor and successor task, you can see the person to whom the task is assigned, any issues with the task, the start and end dates, and the effort expended on the task. For details about a predecessor or successor task, including that task's predecessors and successors, click the task name.

Associate a task with other items

When a task is related to other TeamForge items such as documents, tracker artifacts, file releases, integrated application objects, or news items, you can connect the document to the other item by creating an association.

Creating associations between items helps you define relationships, track dependencies, and enforce work flow rules. Some example uses for task associations include:

- Associating tasks with a master document describing the project, such as a project plan.
- Associating a task with the deliverables it requires, such as a requirements document, completed code, or a file release.
- Associating a task with a related object in the integrated application.
- Associating a task with items on which it is dependent, such as closed tracker artifacts.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, find the task folder containing the task to which you want to add an association, and click the task name.
3. On the **View Task Details** page, click the **ASSOCIATIONS** tab.
4. Click **Add**.
5. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:

    * **ENTER ITEM ID** - If you know the item's ID, you can enter it directly.

        ☞ **Note:**
        * To associate an object in an integrated application from within TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked.
        * Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.

    * **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
    * **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.

6. Click **Next**.
7. You may add a comment in the **ASSOCIATION COMMENT** text box.
8. Save your work.

    * Click **Finish and Add Another** to add additional associations.
    * Click **Finish** to return to the **Details** page.

9. Click the **Associations** tab to view a graphical representation of all the associated items.

    Through the Association Viewer, you can choose to view associations in the form of a list or flip over to the Trace view to explore the layers of associations (including parent/child dependencies) laid out in a timeline. You can scroll across the Trace view by dragging the mouse over the association layer or use the 'Previous' and 'Next' arrows to view all the objects as events in a timeline.

    While the Associations tab shows the count of the total number of associations, you can only view the most recent 500 associations when you click the Associations tab in case the artifact has more than 500 associations. You can, however, browse through the Association Viewer to view older associations.

Manage tasks

Project members with the Task Admin permission can perform all task administrative functions such as copying and moving tasks. They can also create and administer task folders.

Copy a task

Copying a task is useful if you want to use an existing task as a template for creating a new task with many of the same attributes.

When making a copy of a task, reference to the source task is maintained in the task name and history and in the **Associations** section of the **Task Details** page.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, find the task folder containing the task that you want to copy.
3. Select the task that you want to copy, and click **Copy**.
   CollabNet TeamForge creates a copy of the the task and places it on the clipboard.
4. Go to the folder where you want to place the copy and click **Paste**.

    You can place a copy in the same task folder or a different task folder.

The task copy is now placed in the selected folder, with the name `Copy of <Task Name>`.

Move a task

When a task's focus or status changes, it may be useful to move it to a more appropriate task folder.

You can move a task to another folder in the same project or in another project.

To move a task between projects, you must have task administration permission in both the source and destination projects.

If the project members to whom the tasks are assigned have the appropriate permissions in the destination project, the tasks are automatically reassigned to the same project members. If not, you must reassign them.

👉 **Important:** If you move tasks across projects, the dependencies, resources, and change approval settings might be changed.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, go to the task folder containing the task you want to move.
3. Select the tasks you want to move and click **Cut**.
4. Go to the folder to which you want to move the tasks, and click **Paste**.

   👉 **Note:** To move the tasks into another project, first find the destination project using your **Projects** menu.

   • If the destination folder is in the same project, the tasks are now moved.
   • If the destination folder is in a different project, the **Paste Tasks** window appears.
5. If the project members to whom the tasks are assigned don't have the appropriate permissions in the destination project, reassign the tasks to another project member or to yourself.
6. Click **Next**.

The tasks are now in the destination folder. The history of the move is recorded in the task history.

Organize tasks

Use task folders to organize tasks according to work group or project phase.

Create a task folder

Set up a task folder to contain tasks that are related to each other.

Task folders, like file directories, let you organize your tasks in a logical, hierarchical manner with labels that are meaningful to you and your organization.

You can create folders in the root folder or as subfolders in any other folder. There is no limit to the number of folders you can create.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, go to the location where you want to place the task folder.

   You can create a task folder in the root folder or in any other task folder.
3. Click **New**.
4. In the **New Task Folder** window, provide a name for the task folder.
5. Click **Save**.

The task folder is created.

Rename a task folder

When the focus or function of a group of tasks changes, you may wish to rename the folder they are in.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, find the task folder you want to rename.
3. Click the folder title and choose **Rename** from the **Edit** menu.
4. In the **Rename Folder** window, provide a new name for the folder.
5. Click **Save**.

The folder is now renamed.

Reorder task folders

To make it easier for project members to find things, you can arrange task folders alphabetically or otherwise.

By default, task folders are ordered according to when they were created. New task folders are added to the end of the task tree.

👉 **Note:** After you reorder folders, you cannot restore them automatically to their original order. You must reorder them manually.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, find the folder containing the subfolders that you want to reorder.

   You can reorder the task folders in the root directory, or the subfolders in any folder.
3. Choose **Reorder Subfolders** from the **Edit** menu.
4. In the **Reorder Subfolders** window, order the subfolders.

   - Click **Alphabetize** to sort in alphabetical order.
   - Click a folder and click **Move Up** or **Move Down** until the folder is where you want it.
5. Click **Save**.

Delete a task folder
If you no longer need a task folder, it's a good practice to delete it.

☞ **Important:** Deleting a task folder deletes all of the tasks within it. Delete a task folder only if you are sure that you no longer need any of the tasks within it. Move any tasks that you want to save before deleting the task folder.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, click the task folder that you want to delete.
3. Choose **Delete** option from the **Edit** menu.

The task folder is removed.

Document your work
The document management system is a centralized repository for creating, storing, and managing information about a project.

Project members with the Document Admin permission can create, edit and administer documents and document folders. In addition to the Document Admin permission, individual permissions to create, edit and delete documents, and document folders can also be set so that project members who do not have the Document Admin permission can still perform these tasks.

Find and view a document
You can find and view a document by navigating to it, by searching for it, or by viewing it from a URL or unique identifier. You can also open a document from TeamForge directly into Microsoft Office applications.

Go to a document
If you know where a document is posted, or would like to browse to see what documents are available, you can find a document by navigation.

You can get to documents from your project home page or by searching the project's documents page.

1. Choose a way to find documents:

   - Look for a **Documents** window on your project home page. If your project manager has added a document component, you will find all your project documents there (see *Create a project page component* on page 106 to know how to add a "Documents" component to the project home page).
   - If there is no Documents component, click **DOCUMENTS** in the project navigation bar. The **List Documents** page appears.
2. Select a documents folder.

   If a documents folder contains sub folders, you can expand the document tree by clicking the arrow to the left of the folder name.
3. To view or save a document, click the file type icon to the left of the document name.

   Use the filter options to locate the document you want. Documents with file types that can be viewed in a browser, such as HTML and text, are displayed in a TeamForge window. For other file types, use the appropriate viewing software.
4. To view additional information about the document, click the document name.

Search for a document

Search for a document by its title, its content, the project or folder in which it is located, its file type, or its creation date.

If you don't know which project to search, or want to search for documents and other types of items simultaneously, use global search.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Click **SEARCH DOCUMENTS**.
3. On the **Document Search Criteria** page, enter the desired search criteria.

   ☞ **Tip:** You can use wildcards.

   • To search by location: If you know the document is located in the current folder, check the Current Folder in the **SEARCH CATEGORY** field. Otherwise, the default Root Category searches all document folders in the project.
   • To search by name or description, enter the text in the **SEARCH KEYWORDS** field and select **Search Name and Description**.
   • To search by document content, enter the text in the **SEARCH KEYWORDS** field and **Search Within Document**.
   • To search by name, enter the text of the document name in the **DOCUMENT NAME** field.
   • To search all document versions, select **SEARCH ALL VERSIONS**.
   • To search by date range, enter the start and end dates for the search. Click the Calendar icon to select dates from a calendar.
   • To search by author, enter the user name in the **CREATED OR EDITED BY** field. Click the search icon to display a list of project members.

4. Click **Search**.

TeamForge lists the documents matching your search criteria.

Create a document

You can create a document by uploading a file or by entering the text directly into TeamForge.

You can also create a document in TeamForge. TeamForge automatically detects the file extension for many common file types.

   ☞ **Note:** To create a document, you must have the Document Admin permission, or 'Create/View' permission for document folders or documents.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder in which you want to create the document, and click **Add Document**.

   You can create a document in the root document folder or any other document folder.

3. On the **Create Document** page, enter a name and description for the document.
4. Identify the status of the document.

   • **Draft** - The document is not yet completed.
   • **Final** - The document is finished.

5. If you want to prevent others from editing or downloading the document, select appropriate check boxes in the **LOCK DOCUMENT** section.

   • To enable the edit lock for a document, select **Prevent others from editing**. The edit lock prevents other users from editing the document.
   • To enable the download lock for a document, select **Prevent others from downloading**. The download lock prevents other users from downloading the document. You must have enabled the edit lock to make this selection.
   • To enable the Document Admin to edit the document, select **Enable Document Admin to edit**. You can choose to allow the Document Admin to unlock and edit a document after enabling the edit lock.

👉 **Note:**

- By default, all the locks are disabled. You must have the edit lock enabled to select the download lock. Selecting the edit lock automatically enables the Document Admin to edit the document.
- To choose locks in the **LOCK DOCUMENT** section, you must have the Document Edit permission.

6. Select the document creation method.

- To upload a document, select **Upload From File** and browse to the desired file.
- To enter the content as text, select **Create Online**.
- To upload a document from a specific URL, select **Enter Document URL** and provide the URL.

  👉 **Note:** You cannot prevent others from editing or downloading the document if you choose this URL option. Edit and download locks are not effective in this case.

7. Write any comments you may have about this version of the document.

8. Save your changes:

- Click **Save** to save the document and return to the document list view.
- Click **Save and View Details** to save the document and display the document details.

Edit a document

When you are called on to help develop a document, you can edit the document online or by posting a new version.

To edit a document that is currently locked for editing, the user must have the Document Edit permission. To download a document that is currently locked for downloading, the user must have the Document View permission. A Document Administrator that have the option, **Allow Document Admin to edit** enabled in the **LOCK DOCUMENT** section or a Site Admin can also remove the edit and download locks.

Update a document

To update a document, you can upload a new document version or change the document's name, description, or lock status.

You can also update a document in TeamForge directly from one of the following Microsoft Office applications:

- Word
- Excel
- Project
- PowerPoint

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the document that you want to edit.
3. Select the document you want to edit by clicking the document name.
4. On the **Document Details** page, click **EDIT / UPDATE**. You can see this button at the top right corner of the page only if you have Document Edit Permission.
5. On the **Edit Document** page, make your changes.

- Update the name or description by entering new text in the appropriate field.
- Change the status value, if appropriate.
- To set locks, select appropriate check boxes in the **LOCK DOCUMENT** section.
- Update a binary document by selecting **Upload From File**, then browsing for the updated file.
- Update a text document by selecting **Create Online**, then editing the document text.

6. Enter a version comment to describe the changes made, if appropriate.

7. Click **Save**.

The document is now updated.

Change the active document version

You can specify which version of a document is the active version at any time.

TeamForge stores every document version that you or any other user posts. When you open a document from the document list view, you open the active document version. When you open a document's **Document Details** page, the information you see is about the active document version.

The **Versions** section of the **Document Details** page lists all document versions and indicates which one is currently active. By default, the most recently posted document version is the active version.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the document.
3. Select the document by clicking its name.

> ☞ **Tip:** On the **Document Details** page, in the **Versions** section, the active document is indicated with a check in the **Active** column.

4. Select the version that you want to make the active version, and click **Mark As Active**.

The selected document version is now the active version.

Lock a document

While you are working on a document, you can lock it to prevent others from editing or downloading it and then posting new versions to TeamForge .

You can lock a document when you create it or at any time afterward. When a document is locked, any user with the Document View permission can view or download the document. You can edit or download a locked document, if you are a Site Admin or a Document Administrator that have the option, **Allow Document Admin to edit** enabled in the **LOCK DOCUMENT** section.

1. On the **Document Details** page, confirm that the lock status is currently Unlocked.
2. Click **EDIT / UPDATE**. You can see this button at the top right corner of the page only if you have Document Edit Permission.
3. On the **Edit Document** page, select appropriate check boxes in the **LOCK DOCUMENT** section.
4. Click **Save**.
   The document is now locked. It is represented by the Padlock icon in the document list view.

Unlock a document

To work with a document that someone else has locked, you must unlock the document.

You can unlock a document that you have locked before. If you want to unlock a document that was locked by someone else, you must be a Site Admin or a Document Administrator that has the option, **Allow Document Admin to edit** enabled in the **LOCK DOCUMENT** section.

1. On the **Document Details** page, confirm that the document is currently locked.
2. Click **EDIT / UPDATE**. You can see this button at the top right corner of the page only if you have Document Edit Permission.
3. On the **Edit Document** page, clear the check boxes in the **LOCK DOCUMENT** section.
4. Click **Save**.

The document is now unlocked. All project members with the Document Edit permission can now update the document.

Get a document reviewed

There are a few things you have to do to get the most out of a document review.

Start a document review

To get a document reviewed, select the appropriate reviewers, set a due date, and explain to the reviewers what you need from them.

Before starting a document review, you must create the document in TeamForge .

1. Click **DOCUMENTS** from the **Project Home** menu.
2. In the document tree, click the title of the document you want input on.
3. On the **Document Details** page, click the **REVIEW** tab, then click **Start a Review**.

☞ **Note:** If a review is already in process, **Start a Review** does not appear, because you can only have one review at a time.

4. On the **Start a Review Cycle** page, give the review a descriptive name.

5. Enter a due date for reviewers.

   Click the Calendar icon to select dates.

   ☞ **Tip:** Comments are due at 12:00 midnight on the date selected. Therefore, if you want all document reviews to be completed by end of day Tuesday, select Wednesday as the review end date.

6. Select the required and optional reviewers.

   Click the Search icon to pick from a list of project members whose access permissions enable them to see this document.

7. Write a message to your reviewers describing the document and the type of input you need from them.

   Your reviewers will get this message in an automatic email that shows you as the sender.

   ☞ **Tip:** If you add yourself to the reviewers list, you will get a copy of this email for future reference.

8. Click **Save**.

The document is now submitted for review.

* All reviewers receive an email with the details of the review, your message, and a link to the **Review Details** page where they can download the document and post their responses.
* The document status is changed to **Review** and remains so until the review is closed.

☞ **Tip:** To help keep track of the development of your document, associate a tracker artifact with your document review.

Read review responses
After starting a document review, you receive email notifications whenever a reviewer posts review responses.

You can read the review responses from the email or from the **Review Details** page. The **Review Details** page is the starting point for managing all document review activities.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the desired document.
3. Click the title of the document.
4. On the **Document Details** page, click the **REVIEW** tab.
   A summary of all open and closed reviews appears.
5. Click the title of the desired review.
   Review comments are listed in the bottom section of the page.

You can also reach the **Review Details** page by clicking the link provided in any related email notification or by clicking the title of the review in the **My Document Reviews** section of your **My page**.

Edit review details
At any time after a review is started, you can change the review name, the required and optional reviewers, and the due date.

When you edit review details, affected reviewers get an email notification with the details of the change.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the desired document.
3. Click the title of the document.
4. On the **Document Details** page, click the **REVIEW** tab.
   A summary of all open and closed reviews appears.
5. Click the title of the desired review.
   The **Document Review** page appears.
6. Click **EDIT**.

7. On the **Edit Review Cycle** page, make your changes.

   - Click the Calendar icon to select dates from a calendar.
   - Click the Search icon to display a list of project members from whom you can select.

8. Click **Save**.

- When you add reviewers, new reviewers receive an email with the details of the review.
- When you remove reviewers, removed reviewers receive an email telling them that their reviews are no longer necessary.
- When you change the due date, outstanding reviewers receive an email with the new due date.

Send a reminder email

To help project members keep track of their review obligations, you can send a reminder email to reviewers who have not yet submitted their responses.

1. On the **Document Review** page, find the outstanding required and optional reviewers.

2. Click **SEND REMINDER EMAIL**.

3. On the **Send Reminder Email** page, write a message to include in the email.

4. Choose whether to send the document as an attachment to the email.

   ☞ **Note:** The URL to the document is included in the email regardless of whether the document is attached.

5. Click **Send**.

The email is sent to all reviewers who have not yet submitted responses.

Close a document review

After a document review is completed, close the review.

Reviews are never closed automatically. You must close any outstanding reviews on a document before you can start a new one on the same document.

   ☞ **Note:** You can close a review even if the due date has not yet passed.

1. Click **DOCUMENTS** from the **Project Home** menu.

2. Using the document tree, find the folder containing the desired document.

3. Click the title of the document.

4. On the **Document Details** page, click the **REVIEW** tab.
   A summary of all open and closed reviews appears.

5. Click the title of the desired review you want to close.

6. On the **Document Review** page, click **CLOSE REVIEW**.

7. On the **Close Review Cycle** page, change the document status to **Final** or **Draft**.

8. Click **OK**.

The review is now closed. Any reviewers with outstanding reviews receive email notifications that the review is closed. No additional review responses can be submitted.

Review a document

To contribute to a review, you look at the document, consider responses from other reviewers, and post your own responses.

When you are identified as a reviewer of a document, two things happen:

- You receive an email about the review. The details include the due date, whether you are a required or optional reviewer, and any message text entered by the review submitter. The email also includes a link to submit your response.
- A new entry in the **Documents Awaiting Review** section of your **My Page** provides a summary of the review, including the review due date, the project in which the document is posted, and the person who is requesting your review.

1. Go to the **Document Review** page by clicking the link provided.

   - Click the link in the email notification you received with the review details.
   - Click the title of the document in the **Documents Awaiting Review** section of your **My page**.

2. On the **Document Review** page, in the **Submit a Response** section in the center of the page, enter your response.

3. If appropriate, use the **Browse** button to attach a marked-up copy of the document, or another attachment.

4. Click **Submit**.

Your response is now submitted.

- The review submitter receives an email notification containing your response.
- Your response comments appear in the **Responses** section of the **Document Review** page.

Associate a document with other items

When a document is related to other CollabNet TeamForge items such as tasks, tracker artifacts, file releases, or news items, you can connect the document to the other item by creating an association.

Creating associations between items enables you to define relationships, track dependencies, and enforce work flow rules. Some example uses for document associations include:

- Associate a requirements document with supporting feature requests or bugs in the tracker.
- Associate a document, or a document review, with the task that requires it as a deliverable.
- Associate a master document, such as a product plan, with other supporting documents such as a test plan or beta plan.
- Associate a product announcement document with a file release.

1. Click **DOCUMENTS** from the **Project Home** menu.

2. Using the document tree, find the folder containing the document to which you want to add an association. Select the document by checking the box, then click **Details**.

3. On the **Document Details** page, click the **ASSOCIATIONS** tab.

4. Click **Add.**

5. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:

   - **ENTER ITEM ID** - If you know the item's ID, you can enter it directly.

     👉 **Note:**

       - To associate an object in an integrated application from within TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked.
       - Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.

   - **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
   - **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.

6. Click **Next**.

7. You may add a comment in the **ASSOCIATION COMMENT** text box.

8. Save your work.

   - Click **Finish and Add Another** to add additional associations.
   - Click **Finish** to return to the **Details** page.

9. Click the **Associations** tab to view a graphical representation of all the associated items.

   Through the Association Viewer, you can choose to view associations in the form of a list or flip over to the Trace view to explore the layers of associations (including parent/child dependencies) laid out in a timeline. You can scroll across the Trace view by dragging the mouse over the association layer or use the 'Previous' and 'Next' arrows to view all the objects as events in a timeline.

   While the Associations tab shows the count of the total number of associations, you can only view the most recent 500 associations when you click the Associations tab in case the artifact has more than 500 associations. You can, however, browse through the Association Viewer to view older associations.

Manage documents

Project members with the Document Admin permission can copy and move documents. They can also create and administer documents. However, they can delete documents, only if they have the specific delete permission.

Copy a document

If a given document belongs logically to two or more document folders, keeping a copy in each folder can help users locate it.

Copying a document can also be useful if you want to use an existing document as a template for creating a new document. When making a copy of a document, reference to the source document is maintained in the document title and the **Associations** tab.

☞ **Note:** Each copy must be maintained separately. Edits made to one copy are not propagated to other copies.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the document that you want to copy.
3. Select the document you want to copy, then click **Copy**.
   CollabNet TeamForge creates a copy of the document and places it on the clipboard.
4. Find the folder where you want to place the copy, then click **Paste**.

   You can place a document copy in the same folder or a different folder.

The document copy is now placed in the selected folder, with the name "Copy of <Document Name>". The reference to the source document is also maintained in the **Associations** section of the **Document Details** page.

Move a document

You can move it to another document folder within the project or to a document folder in another project at any time. This allows you to reorganize documents as needed.

You can move one or more documents in the same operation. You cannot move a document while it has an open review.

☞ **Note:** To move a document between projects, you must have document administration permission in both the source and destination projects.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. In the document tree, find the folder containing the document you want to move.
3. Check the box next to the document you want to move, and click **Cut**.
4. Find the folder to which you want to move the document, and click **Paste**.

   ☞ **Note:** To move the document into another project, first find the destination project using your Projects menu.

The document is now moved to the selected folder.

Delete a document

If you no longer need a document, you can delete it.

Deleting a document deletes all of the document's versions, plus the document's details and review history.

☞ **Note:** To delete documents, you must specifically have the delete permission ('Delete/View') for document folders or documents. This is irrespective of whether you have the Document Admin permission or not.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the document that you want to delete.
3. Select the document you want to delete.
4. Click **Delete**.

The document is deleted.

Delete document versions

Document versions can grow in number over time. You may choose to delete one or more versions of a document if you no longer need them. This saves a lot of disk space and maintenance overheads.

You can delete a document's version only if:

- you have the delete permission ('Delete/View') for documents. This is irrespective of whether you have the Document Admin permission or not.
- the document has more than one version available.
- the selected document version is not marked as an active version.

⚠ **Caution:** Deleting a document's version permanently removes the version and cannot be undone. Exercise caution while deleting document versions.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder containing the document that you want to delete.
3. Select a document and click **Details**. You can also click the document's link to see its versions.
4. Select one or more document versions you want to delete from the **VERSIONS** tab and click **Delete**.
   A confirmation message is displayed.
5. Click **OK** to delete.

The document versions are deleted.

Organize documents

Project members with the Document Admin permission can create, edit, move and reorder document folders. Similarly, project members with the relevant individual permission can create, edit, delete or view document folders.

Create a document folder

Document folders, like file directories, allow you to organize your documents in a logical, hierarchical manner using labels that are meaningful to you and your organization.

You can create folders in the root folder or as subfolders in any other folder. There is no limit to the number of folders you can create.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the location where you want to place the folder.
3. Click **NEW**.
4. In the **Create Folder** window, enter a name for the folder.
5. Click **Save**.

The folder is created.

Rename a document folder

When the function of a document folder has changed, it's a good idea to rename the folder.

1. Click **DOCUMENTS** from the **Project Home** menu.
2. Using the document tree, find the folder that you want to rename, and click the title to select it.
3. Choose **Rename** from the **EDIT** menu.
4. In the **Rename Folder** window, enter a new name for the folder.
5. Click **Save**.

The folder is now renamed.

Move a document folder

You can move a document folder within a project.

You can move a folder in the following ways:

- From the root folder to a subfolder.
- From a subfolder to the root folder.
- From a subfolder to another subfolder.

When you move a document folder, any documents and subfolders that it contains are also moved to the destination folder.

Project members monitoring the destination folder will continue to receive monitoring notifications after a folder is moved. Project members not monitoring the destination folder will not receive further monitoring notifications.

1. Click **DOCUMENTS** from the **Project Home** menu.

2. Using the document tree, find the folder that you want to move. Click the title to select it.

3. Choose **Cut** from the **EDIT** menu.

4. Using the document tree, find the folder into which you want to move the selected folder. Click the title of the folder to select it.

> ☞ **Note:** You can move a folder into the root folder or into any other folder.

5. Choose **Paste** option from the **EDIT** menu.
The folder is moved to the destination folder.

Reorder document folders
To change the order in which document folders are displayed, you can alphabetize them or reorder them arbitrarily.

By default, document folders are ordered according to when they were created. New document folders are added to the end of the document tree.

1. Click **DOCUMENTS** from the **Project Home** menu.

2. Using the document tree, find the folder containing the subfolders that you want to reorder. Click the title to select it.

3. Choose **Reorder Document Subfolders** from the **EDIT** menu.

4. Organize the subfolders:

- To sort them alphabetically, click alphabetize icon.
- To reorder a specific folder, select it by clicking the title, then click **Move Up** or **Move Down** until the folder is where you want it.

5. Click **Save**.

Delete a document folder
If you no longer need a document folder, it's a good idea to delete it.

Deleting a document folder deletes all of the documents within it.

> ☞ **Important:**
>
> - Delete a document folder only if you are sure that you no longer need any of the documents within it. Move any documents that you want to save.
> - Even if you have the Document Admin permission, you cannot delete a document folder unless you specifically have the delete permission.

1. Click **DOCUMENTS** from the **Project Home** menu.

2. Using the document tree, find the folder you want to delete.

3. On the document folder **EDIT** menu, click **Delete**.
If the folder contains one or more documents that you do not have permission to delete, you cannot delete the folder.

Work with documents from Microsoft Office
You can use the **File** menu of your Microsoft Office application to work with documents that are stored on your TeamForge site.

Open a TeamForge document in Microsoft Office
You can get to your Microsoft Word, Excel, Project, and PowerPoint documents in CollabNet TeamForge even if the CollabNet TeamForge web interface is not open.

1. Open your Microsoft Office application.

2. From the **File** menu, choose **Open from CollabNet TeamForge** .

3. In the **Login** window, provide the following information:

- Your TeamForge user name and password.
- The URL of your organization's TeamForge installation.

> ☞ **Note:** Do not enter the `http://` or `https://` before the server name.

4. Click **Log In**.
   The **Selection Dialog** window shows all of the projects, document folders, and documents that you have permission to view.

5. Select the document you want to open.

The document opens in your Microsoft Office application.

Save a document to TeamForge

You can store your Microsoft Office documents in CollabNet TeamForge even if the CollabNet TeamForge web interface is not open.

CollabNet TeamForge works with Microsoft Word, Excel, Project, and PowerPoint.

1. Open your Microsoft application.
2. Open the document you want to save.
3. From the **File** menu, choose **Save to CollabNet TeamForge** .
4. In the **Log in** window, enter the following information:

   • Your TeamForge user name and password.
   • The URL of your organization's TeamForge installation.

   ☞   **Note:** Do not enter the `http://` or `https://` before the server name.

5. Click **Log In**.
   The **Selection Dialog** window displays all of the projects and document folders in which you have permission to create documents.

6. Select the document folder in which you want to save the document.

   ☞   **Note:** Do not click on an existing document. This will prompt you to update that document.

7. In the **Document Properties** window, clear **Use Filename**, then enter a name in the **CollabNet TeamForge Document Name** field.

   By default, the file name is used as the document name.

8. Write a description for the document.
9. Choose whether the document is in draft or final status.
10. To lock the document and prevent others from editing it, select **Lock Document**.
11. Write a version comment, if appropriate.
12. Click **OK**.

The document is now saved in CollabNet TeamForge. You can see the document by going to the CollabNet TeamForge document folder that you selected.

   ☞   **Note:** After you save a document to CollabNet TeamForge, future saves will also save in CollabNet TeamForge. If you want to save a document to a local drive, choose **Save As** from your **File** menu.

Communicate with project members

You can communicate with project members via discussion forums, mailing lists, wiki pages, and project news announcements.

Participate in discussions and mailing lists

CollabNet TeamForge discussions provide workspaces where project members can discuss project-related topics online or by email.

Create a forum topic

Create a new forum topic to begin discussion of a new subject.

A topic starts a message thread to which other users can reply. A forum can have any number of topics.

A forum topic is similar to an email, in that you can use it communicate with other people subscribed to the forum, as if it were an email list. If the forum owner has enabled the forum to work as a mailing list, then you can post to the forum by email as well.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click the title of the forum in which you want to create a topic.
3. On the **Topic Summary** page, click **Create**.
4. On the **Create a Topic** page, describe the topic in the **Subject** field.
5. Write the message in the **Message** field.
   After the topic is created, other users can reply to this message.
6. If you want the message sent by email to people who are not members of the forum, add their email addresses to the **Other Recipients** field. If there is more than one, put commas, semicolons or spaces between them.

   ☞ **Note:** In a moderated discussion forum, addresses in the `Other Recipients` field get your message only after the moderator approves the message.

7. To add an attachment to the topic, click **CHOOSE FILE** and select the file.
8. Click **Save**.

   ☞ **Note:** If this discussion forum is moderated, the topic is held until a moderator approves or rejects it. (Except if it is from a trusted user. These messages don't require moderation.)

Search for posts
Search for a post by using keywords, specifying the forum it belongs to, selecting the sender of the post or entering relative date or date range.

You can search for posts either across all forums or within specific forums.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click **Search Posts**.
3. On the **Post Search Criteria** page, enter the desired search criteria.

   ☞ **Tip:** You can use wildcards.

   • To search by subject, body or attachments, enter the text in the **Search Text** field and select **Subject**, **Body** and/or **Attachments** options.
   • To search by forum name, select the forum in the **Forum** field.

     ☞ **Note:** When none of the forum options are selected, the system searches for matching posts across all forums.

   • To search by post-sender, select the user name in the **Posted By** field.

     ☞ **Note:** You can select **User running search** option to display your posts.

   • To search by time span, specify relative dates such as "Within the last 7 days".
   • To search by date range, enter the start and end dates for the search. Click the Calendar icon to select dates from a calendar.

4. Click **Search**.

All posts matching your search criteria are displayed in the **Search Results** page.
Reply to a forum message
You can post a message in any topic in any forum you have access to. You can also post a message in reply to another message.

If the forum is moderated, you must have posting permission. Contact the forum moderator.

   ☞ **Tip:** If you are getting your forum messages delivered as email, you can reply to a post by email too.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, find the topic in which you want to post a message.
3. In the section containing the message to which you want to respond, do one of the following:

   • Click **Quote** to quote the original message in your response message.
   • Click **Reply** to omit the original message from your response message.

4. Write the message.
5. To add an attachment to the message, click **CHOOSE FILE** and select the file.
6. Click **Save**.

The forum message is posted. Other project members can reply to it using the same process.

☞ **Note:** A message to a moderated discussion forum is held until a moderator acts on it. (Except if it is from a trusted user. These messages don't require moderation.)

Subscribe to a discussion or mailing list
When you monitor a discussion forum, you are notified of contributions to the forum by email. Monitoring a forum is the same thing as subscribing to a mailing list.

A bell icon in the **Monitoring** column indicates that you are subscribed to a forum.

It looks like this:

- If you prefer to do it by e-mail:
  a) To subscribe in message by message type subscription, send an email to `<Email address name in Mailing list>-<project name>-subscribe@<domain name>`
  b) To subscribe in digest type, send an email to `<Email address name in Mailing list>-<project name>-digest-subscribe@<domain name>`
  c) You can also change the subscription type from message by message to digest and vice-versa.
- If you prefer to do it through the web interface:
  a) Click **Discussions** from the **Project Home** menu.
  b) Click **Monitor**.
     Set the notification frequency using "Monitoring Preference" as it suits you.

Subscribe others to a discussion or mailing list
You can add other users to a discussion.

If you are a forum administrator, you can also add users to the discussion as a group.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. Select the forums that you want to add users to.
3. In the **Monitor** list, click **Users Monitoring Selected**.
4. On the **USERS** tab, Click **Add**.
5. On the **Find a User** window, move the users you want into the **USERS TO ADD** column.

   ☞ **Tip:** To add users as a group (assuming you are a forum administrator), do the same operation on the **USER GROUPS** tab.

Post to a forum by email
If the forum also works as a mailing list, you can create a forum topic by sending an email message to the forum. You can also reply to a post by replying to the email.

If the forum is moderated, you must have posting permission. Contact the forum moderator.

The forum's email address, if it has one, appears on the **Topic Summary** page.

☞ **Tip:** If a forum is not set up to work as a mailing list, you can still get posts by email when you monitor the forum. See *Monitor many items* on page 101.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click the forum in which you want to create a topic.

3. On the **Topic Summary** page, find the email address in the **Mailing List** field. Send your email message to that address.

CollabNet TeamForge maps your email to the forum topic like this:

Email Field	Forum Topic field
To	Forum email address
Subject	Title of forum topic
Body	Text of forum topic
Attachments	Attachments

Emails from the forum can be read in RTF (Rich Text Format) or HTML format. The format of the message is delivered as an attachment to the post. Embedded attachments, such as text or images, are also delivered.

A message to a moderated discussion forum is held until a moderator acts on it. (Except if it is from a trusted user. These messages don't require moderation.)

In a moderated discussion forum, if you add other addresses in the `cc:` field of your email, those addresses get your email only after the moderator approves the message.

Associate forum messages with other items
When a forum message concerns some other CollabNet TeamForge item, such as a document, a tracker artifact, a file release, a code commit, or a task, link the message to the item under discussion by creating an association.

Creating associations between items enables you to define relationships, track dependencies, and enforce work flow rules.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, find the forum message with which you want to create an association.

Any existing associations are displayed in the **Associations** section.
3. Click **Associate.**
4. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:

- **ENTER ITEM ID** - If you know the item's ID, you can enter it directly.

  ☞ **Note:**

  - To associate an object in an integrated application from within TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked.
  - Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.
- **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
- **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.

5. Click **Next**.
6. You may add a comment in the **ASSOCIATION COMMENT** text box.
7. Save your work.

- Click **Finish and Add Another** to add additional associations.
- Click **Finish** to return to the **Details** page.

Administer forums and mailing lists
Forum administrators create forums and do what is needed to keep them on track, such as editing or moderating forum posts.

You can choose to make a discussion forum either public or private.

You can be a forum or mailing list administrator without being a project administrator. Ask your project administrator or site administrator to grant you forum administration permissions.

Forum administrators can enable or disable forum moderation and add or remove moderators and trusted users. Any project member with forum post permissions can be a moderator.

Forum administrators can also make forums work like mailing lists.

Guest users can monitor a forum if email monitoring is set to "Allow all site users and guests". Additionally, guest users can email-post or subscribe to a forum if the mailing list is enabled and "Email Posting" is set to "Allow all site users and guests" from the discussion settings.

☞ **Tip:** Who can post by email to a discussion forum is controlled by the "Email Posting" options set for the forum. It does not depend on the permissions set for users of the Web forum.

Create a discussion forum

Encourage project members to work together by creating discussion forums to which project members with the appropriate permissions can post messages.

Discussion forums can also function as mailing lists.

As a forum administrator, you can choose to make a discussion forum either public or private.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click **Create**.
3. On the **Create Discussion Forum** page, enter a title and description for the forum.
4. To make it a private discussion forum, set the **TYPE** to Private.
   Private discussion forums restrict anyone without specific access permissions from posting to the forum. For example, you may want to restrict a preliminary planning discussion to your project's core team before sharing it more widely.
5. If you want the forum to work as a mailing list, select **Enable Mailing List**.
   a) Provide a name for the mailing list in the **EMAIL ADDRESSddress** field.

      ☞ **Note:** The mailing list name must be unique within a project.

   b) Set who can *post to the forum via emails*. Choose either User with Roles and Permissions (default) or Allow only forum admins from the **Email Posting** drop-down list.
   c) Set who can *subscribe to monitoring via emails*. Choose either User with Roles and Permissions (default) or Allow only forum admins from the **Email Monitoring** drop-down list.

      ☞ **Note:** Selecting Allow only forum admins for **Email Monitoring** will not restrict users with 'Discussion-View' permission from getting monitoring emails in case they choose to *monitor the forum via the web UI*.

   d) Choose how replies to posts are handled by setting the **REPLY BEHAVIOR**.

      ☞ **Tip:**
      • Many users are accustomed to having their replies go automatically to the whole list. Others are used to having replies go just to the original sender. You should check with your users to see what makes more sense for a particular mailing list.
      • When **REPLY BEHAVIOR** is set to "To the list," email replies are sent to the list as a whole, not to the individual post. This may be a change from what some users are used to.

   e) Specify a prefix for the subject lines of messages from this list.

      This can help users sort their incoming messages, if they are subscribed to multiple lists.

   f) You may limit the size of emails (including attachments, if any). Enter the size (in MB) in the **MESSAGESIZE** field.
   g) Under **FOOTER TEXT**, provide any information you want to show up at the bottom of each email that subscribers receive.

      For example, you may want to offer useful web locations or email addresses.

6. To make this a moderated forum, select **ENABLE MODERATION**.
   a) Click the Search icon to add moderators.

      A moderated discussion must have at least one moderator.

☞ **Note:** If your project includes members of a parent project, you can select those members too.

    b) Click the Search icon to add trusted users.

       Posts by trusted users do not need moderator approval.

7. On the **Create Discussion Forum** page, click **Save**.

If you set your forum to work as a mailing list, all project members monitoring the forum will receive notifications whenever a new topic or message is posted.

Rename or edit a discussion forum

    To help keep a forum or mailing list focused, try updating its title.

    As a forum administrator, you can enable/disable mailing list or moderation features or just update their settings for the discussion forum.

1. Click **DISCUSSIONS** from the **Project Home** menu.

2. On the **Forum Summary** page, select the forum you want to change, and click **Edit**.

3. On the **Edit Discussion Forum** page, make your changes.

4. Click **Save**.

☞ **Note:** All project members monitoring the forum receive notifications of the update.

Add or modify moderators

    As a forum administrator, you can add or remove forum moderators.

☞ **Note:**

    • If a forum is moderated, it must have at least one moderator.

    • When you designate a forum moderator, you also become a moderator yourself.

1. Click **DISCUSSIONS** from the **Project Home** menu.

2. On the **Forum Summary** page, select the forum for which you want to add or modify the moderators.

3. On the **Topic Summary** page, click **Edit**.

4. On the **Edit Discussion Forum** page, add or modify the forum moderators.

☞ **Note:** The existing moderators are listed.

5. Click the **Search** icon to add or remove forum moderators.

    You can select one or more moderators.

☞ **Note:** You can select the inherited project members also from the list.

    a) To add moderators, select the required users, click **Add** and click **OK**.

    b) To remove moderators, select the required users, click **Remove** and click **OK**.

6. On the **Edit Discussion Forum** page, click **Save**.

Moderate a discussion forum by email

    If your discussion forum is also a mailing list, you can approve or reject the post by email.

    A message to a moderated discussion forum is held until a moderator acts on it. (Except if it is from a trusted user. These messages don't require moderation.)

    As a moderator, you get an email when a message is awaiting moderation. The email contains the URL where you can approve or reject the message.

    See the notification email for your options.

Option	Description
To accept the message:	Send an email to <postId>-accept@<domain>, or just click **Reply**.

Option	Description
To accept the message and add the sender to the "Trusted Users" list:	Send an email to <postId>-allow@<domain>, or click **Reply-to-All**.
To reject the message:	Send an email to <postId>-reject@<domain>.
To add your comments to the message:	Include your comments in the **Start Comment** and **End Comment** blocks in your response email. Your comments will appear in the approved post.

Approve a forum post

If a post is appropriate, you can add it to the forum by approving it.

A message to a moderated discussion forum is held until a moderator acts on it.

As a moderator, you get an email when a message is awaiting moderation. The email contains the URL where you can approve or reject the message.

☞ **Note:** If your discussion forum is also a mailing list, you can approve or reject the post by email. See the options in the notification email.

☞ **Note:** If the sender consistently contributes useful and appropriate input, you can save the moderation time by designating that user as a "trusted" user. Posts from trusted users don't have to be moderated.

1. On **My Page**, click the **ITEMS AWAITING MY APPROVAL** tab.

   The number of posts awaiting approval are displayed against the project names.

2. To view forum details, click the hyperlinked **Number of Posts** for your project.

3. Click the **Number of posts awaiting approval** link on the **Forum Summary** page.

   ☞ **Note:** The **Forum Summary** page displays all the forum names and the corresponding number of posts awaiting approval, if any.

4. Select either of the three approval techniques in the **Posts Awaiting Approval** tab.

   As the moderator of the post, you will be able to view the topic title in the **All Topics** tab; and post title in the **Posts Awaiting Approval** tab on the **Topic Summary** page.

   ☞ **Note:** In the **All Topics** tab, the hourglass icons differentiate the topics that contain posts awaiting approval. To view posts nested within a topic, you can use the hyperlinked topic title.

Option	Description
To approve posts and senders individually:	Select the post and click **Approve** or **Approve and Trust**.
To approve multiple posts and senders at once:	Select all the posts to be approved and click **Approve** or **Approve and Trust** below the post details.
To view the post details and approve:	Click the post title and click **Approve** or **Approve and Trust** on the **Review Post Awaiting Approval** page after reading the details.

Reject a forum message

If a proposed message is not appropriate or does not contribute to the goals of your discussion forum, you can reject it.

A message to a moderated discussion forum is held until a moderator acts on it. (Except if it is from a trusted user. These messages don't require moderation.)

As a moderator, you get an email when a message is awaiting moderation. The email contains the URL where you can approve or reject the message.

☞ **Note:** If your discussion forum is also a mailing list, you can approve or reject the post by email. See the options in the notification email.

☞ **Note:** You can reject posts with or without comments or reasons for rejection.

1. On **My Page**, go to **ITEMS AWAITING MY APPROVAL** tab.
2. On the **Forum Summary** page, click **Number of posts awaiting approval**.
   Topics that contain posts awaiting approval have an hourglass icon.
3. On the **Posts Awaiting Approval** tab, choose your method of rejection.

Option	Description
To reject the posts individually:	Select the each post, then click **Reject** at the end of the post.
To bulk-reject the posts:	1. Select all the posts you want to reject, then click **Reject** below the post details table.
To view the post details and reject:	Click the hyperlinked post title, then click **Reject** on the **Review Post Awaiting Approval** page.
To explain your rejection with a comment:	Click **Reject With Comment** instead of **Reject**.
	☞ **Note:** The rejection comment is posted to the message sender.

Rejected messages are deleted from the posts awaiting approval list and the message senders are notified by email.

Delete a forum message

When a message in a forum is off topic or potentially harmful, you may want to delete it.

Before deleting a forum message, consider leaving it in place so that future users can consult it if they need to. Consider this even if the message does not seem very useful right now.

☞ **Important:** You cannot delete the topic starter's original message without deleting the entire topic.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, find the forum message that you want to delete.
3. Click **Delete** and confirm that you want to delete the forum message.

The forum message is deleted.

Delete a forum topic

If you no longer want a forum topic in your project, you can delete it.

☞ **Important:** Deleting a forum topic deletes all of the forum messages in the topic. Delete a forum topic only if you are sure that you no longer need any of the forum messages in it.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click the title of the forum containing the topic that you want to delete.
3. On the **Topic Summary** page, select the topic you want to delete.
4. Click **Delete** and confirm that you want to delete the topic.

The forum topic is deleted.

Stop moderating a forum

If a moderated discussion forum does not require posts to be moderated anymore, moderation can be disabled.

To change a moderated discussion forum to unmoderated discussion forum in CollabNet TeamForge, select the forum and turn off its moderation feature.

☞ **Note:** Users with forum admin permissions only can enable/disable moderation.

☞ **Note:** On disabling moderation on a moderated forum, the posts awaiting approval are automatically approved.

1. Click **DISCUSSIONS** from the **Project Home** menu.

2. On the **Forum Summary** page, select the moderated forum that does not require moderation anymore.

3. On the **Topic Summary** page, click **Edit**.

4. On the **Edit Discussion Forum** page, to disable moderation, de-select the **ENABLE MODERATION** check box.

> ☞ **Note:** The following message is displayed: "Any post awaiting moderation will be approved automatically."

5. Click **Save** to turn off the moderation.

   To keep moderation enabled, select Cancel.

> ☞ **Note:** The moderated discussion forum changes to an unmoderated discussion forum and any posts sent to the forum will be displayed without any restrictions.

Share project news

Regular project news announcements help members stay in touch with events that can affect their work on the project.

Post a news item

To help project members stay up to date, post regular news items on the project home page.

News items are posted and displayed on the project home page and on the CollabNet TeamForge home page.

1. Go to the home page of the project in which you want to post the news item.

   • From within the project, click **Project Home** in the project navigation bar.
   • From anywhere in TeamForge , choose the project from the **Projects** menu in the TeamForge navigation bar.

2. In the **Project News** section, click **CREATE NEWS POST**.

3. On the **Create News Post** page, provide a title for the news item.

4. Write the news item in the **BODY**.

   The news item can be up to 4000 characters long, including spaces.

5. Click **Create**.

The news item shows up on the project home page and the TeamForge home page immediately.

Delete a news item

It's a good idea to promptly delete a news item that is out of date or otherwise incorrect.

Deleting a news item from a project also deletes it from the TeamForge home page.

1. Find the project home page.

   • From within the project, click **Project Home** in the project navigation bar.
   • From anywhere in TeamForge navigation bar.

2. In the **Project News** section, click **Delete** next to the news item that you want to delete, and confirm that you want to delete the news item.

The news item is deleted.

Edit a news item

To keep the project news in sync with the developments, modify the news items as and when required.

1. Find the project home page where you want to update the latest news.

   • From within the project, click **Project Home** in the project navigation bar.
   • From anywhere in TeamForge , choose the project from the **Projects** menu in the TeamForge navigation bar.

2. In the **Project News** section, click **Edit** next to the news item that you want to modify.

3. On the **Edit News Post** page, modify the title and/or the content of the news item.

4. Click **Save**.

The news post appears on the project home page, along with your name and the time at which you modify the post.

Contribute to the project wiki

The TeamForge Wiki allows you to create an unlimited number of Wiki pages in each TeamForge project. Wiki pages are a tool for managing project information as unstructured, linkable content.

Start a Wiki

To start communicating with other project members via Wiki, create a new page in your project's Wiki.

Every TeamForge project starts with a blank Wiki. You do not need to create a Wiki before you can begin adding content. After a Wiki is started, any user with the appropriate permissions can add or edit content; however, you cannot delete all of the content to start over with a new, blank Wiki.

1. Click **WIKI** from the **Project Home** page.
2. On the Wiki home page, click **Edit**.
3. On the **Edit Wiki Page**, write your Wiki text.

   Wiki content is a combination of plain text, markup for font elements such as bold or italics, headers, bulleted and numbered lists, and links.
4. Customize your Wiki entry with any of these optional steps:

   a) Use the buttons at the top of the text area to add Wiki markup to your text in the **WYSIWYG Editor** mode.

   You can also enter Wiki syntax directly into the text area in the **Plain Editor** mode. For an explanation of Wiki syntax, click *Syntax Reference*.

   b) To insert a link to another TeamForge item, just type the item id.

   You do not need additional Wiki syntax to create the link.

   c) To change the size of the display window, drag the arrow available at the bottom-right of the window.

   d) To attach an external file to a wiki page, click **CHOOSE FILE**, then browse for the desired file.

   e) To add a version comment, write it in the **Version Comment** field.
5. Click **Preview Changes** to see how your Wiki content will look.

   You can make further edits from the Previewing Home page before saving your changes.
6. Click **Update** to save your changes.

Your Wiki is now started.

Add Wiki content

When the information you want to share with other project members does not fit neatly into a tracker comment or a document review, use a Wiki page for a more free-form communication flow.

After a Wiki is created, any user with the Wiki create, edit, or view permission can add or edit Wiki content. You can also add associations or multiple attachments, or create additional Wiki pages.

☞ **Note:** You can now add multiple attachments in a Wiki page without updating the Wiki each time to include another attachment. If attachments are not required they can be deleted before updating the Wiki page.

1. Click **Wiki** from the **Project Home** menu.
   Any existing Wiki content appears.
2. For more information about this Wiki, click **View Details**.
3. On the Wiki home page, click **Edit**.
4. On the **Edit Wiki** page, make your changes or additions to the Wiki content.

   • If you prefer to use buttons to do your text formatting, the way you would with a word processor, click [NAME OF BUTTON].

   • If your tastes run more to typing in your wiki formatting, see *Wiki syntax* on page 308 for the choices available.

   ☞ **Tip:** You can use a variety of preconfigured queries to generate up-to-date content for your Wiki page.

   For more preconfigured Wiki content, see *Wiki syntax* on page 308.
5. In the **Version Comment** box, note the reason for your change.

   It is optional, but advisable, to get in the habit of recording a version comment. If your project manager has made it mandatory, then you must record a version comment before you can save your changes.
6. Click **Preview Changes** to see how your Wiki content will look.

   You can make further edits on the **Previewing** page before saving your changes.

7. Click **Update** to save your changes.

Create a new Wiki page

A TeamForge site can have any number of Wiki pages. All Wiki pages are linked, and their relationships are traced on the **Back Links** tab of each Wiki page.

1. Click **WIKI** from the **Project Home** page.

2. In the large text field, insert the title of your new page between square brackets, like this:

```
Please post comments about [test results] here.
```

Then click **Update**.

The text between the square brackets becomes a link on your Wiki page.

☞ **Note:** You can also type the new link in CamelCase (each word starts with an uppercase letter, no spaces) and skip the square brackets.

3. Click the link to open the new Wiki page.
The new page is created. The title of the new page is the same text as the link, rendered in CamelCase.

4. On the **Create Wiki** page, write the content you need, then click **Save**.

☞ **Tip:** You can use a variety of preconfigured queries to generate up-to-date content for your Wiki page.

For more preconfigured Wiki content, see *Wiki syntax* on page 308.

The new Wiki page is created. The back link to the page from which it was created appears on the **Back Links** tab of the **Show/hide Details** section.

Search a wiki

Use the wiki search page to find content in a project wiki.

1. Click **WIKI** from the **Project Home** page.

2. On the **Wiki home** page, click **Search Wiki Pages**.

3. Try one of the predefined searches.

These can save you time by running some of the most widely used content searches with a single click:

• List the wiki pages that have changed in the last 15 days.
• List the wiki pages that no other wiki page links to.
• List all the pages in this project's wiki.

4. If you need to narrow your search beyond the predefined searches, enter some search terms under **Wiki Pages Search Criteria**.

Wildcards are allowed.

• To search active wiki content, enter the search keywords in the in the **SEARCH TEXT** field.
• To search active and inactive wiki page versions, select **Search All Versions**. (By default, searches are performed on active wiki page versions only.)
• To search wiki attachments, select **Include Attachments**.
• If you know approximately when the wiki content was created or last edited, enter the start and end dates for the search and click **Search**. Click the calendar icon to select dates from a calendar.
• To search by author, click the Search icon in the **CREATED OR EDITED BY** field that displays a list of project members.

5. Click **Search**.

A list of wiki pages matching your search criteria appears.

View a wiki page as HTML

One way to resolve messy or incorrect formatting on a wiki page is by converting the page to clean HTML.

1. Click **WIKI** from the **Project Home** page.

2. Find your wiki page by navigating or searching.

3. Click the **View HTML** button.

The wiki uses HTML Tidy to display the cleanest HTML it can manage. View the page source to see the results.
View a wiki page as a PDF
When you want to send a wiki page to someone outside the project, it can be handy to convert it to a PDF document.

1. Click **WIKI** from the **Project Home** page.

2. Find your wiki page by navigating or searching.

3. Click the **View PDF** button.

Depending on how your browser is set up, the resulting PDF document appears or the browser offers to download it for you.

Manage source code

Browse a project's source code repositories for detailed information about code commits, changed files, and associations with other CollabNet TeamForge items.

Get the code

Browse TeamForge to find the code you want to work on, then check out the code.

You can view the contents of each file in a repository, plus additional information about each file such as revision history, comments, date and time of submission, and branch and tag information. You can also view differences (diffs) between any two files.

👉 **Restriction:** You can see only those paths in the repository that the repository administrator has given you access to.

👉 **Note:** If you're getting code from a Subversion replica repository, the TeamForge account used while setting up the replica determines what's available to be checked out. This account could have been provided total access to the master repository, or restricted access using path-based permissions.

1. Click **SOURCE CODE** from the **Project Home** menu.

2. On the list of project repositories, click the name of the repository in which you want to view code.
   For each file, the revision number, time since check-in, author, and last log entry appear in the **Repository Browser**.

   • To view a file, or to view the diffs between two files, click the file name.
   • To view a specific version of the file, click **Download**.
   • To view the differences between two files, do either of these:

     • Click **[select for diffs]** next to each of the two files that you want to compare.
     • Enter the file revision numbers in the Diffs between boxes at the bottom of the page.

3. If you need to diff files, choose a display from the **Type of Diff** menu, then click **Get Diffs**.
   The differences between the two files are displayed.

4. Use your source control client to check out the code to your local machine.

Internal code browser

For Subversion and Git repositories, you have the option to use the TeamForge code browser which is turned on by default while integrating the source code server. Fore more information, see *Integrate a source code server*.

5. On the list of project repositories, click the name of a Subversion or a Git repository in which you want to view code.
   On the top right of code browser, you can select the branch/tag (for Git) or specify the revision (for SVN) you want to browse.

   • **View**: This tab allows you to do the following:

     • Browse through the folder hierarchy of the repository and view the content of specific files. For any folder or file you are viewing within a branch (Git) or revision (SVN), you can obtain the commit information pertaining to its last update.
     • While viewing a single specific commit or a file, you can see the paths that were modified in that commit, the associations including JIRA such as builds, code reviews and so on (from Orchestrate events) for the specific commit and the differences between files in that commit.

☞ **Important:** To view the associations, you must have installed Orchestrate and must have RBAC (role-based access control) permissions to use the "Pipeline Read" or "Reporting API" of TeamForge Orchestrate. If either of these requirements is not met, this section will not show up at all.

- While viewing a folder, if there is a file named `readme`, `readme.txt` or `readme.md` that file will automatically be rendered beneath the list of files in the folder. If the file contains markdown formatting, it will be rendered as rich text.
- **Changes**: This tab lets you view all of the commits that touched a specific path you are browsing within a branch or revision. Click a commit to view its details.
- **Download**: This tab allows you to download a copy of the required file.
- **Graph**: This tab provides a graphical representation of the changes made including branching and merging of repositories.
- **Branches** (for Git): This allows you to see all of the branches in the repository in their relation to the default (master) one. Using **Compare Branch** you can see the commits in the branch that do not exist in the default branch.

View code commits

To stay up to date with code development on a project, browse the code commits made to each repository integrated into your TeamForge site.

For each code commit, you can view the list of files that were checked in, the version history of each file, and any associations with other TeamForge items, such as tracker artifacts or tasks.

☞ **Restriction:** You can see only those paths in the repository that the repository administrator has given you access to.

1. Click **SOURCE CODE** from the **Project Home** menu.
2. From the list of project repositories, select the repository you want to look at, then click **View Commits**.

   The **Commits** section of the **Repository Details** page lists all the code commits in the repository. By default, it shows the commits made over the preceding seven days.

   ☞ **Tip:** To get an automatic email notification whenever code is committed to this repository, click the **Monitor** menu, then choose **Monitor Selected**. The emails will contain all the details you can see on this page, unless the repository's owner has chosen to send emails with only the essential information.

3. Specify the filter criteria in **Commit Name**, **Committed By** or **Committed On** date ranges and click **FILTER**.
4. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.
5. To view the details of a commit, click its title.

   The **Files** section of the **Commit Details** page lists all files that were checked in with the code commit, including the version number of each file and the last operation that was performed, such as modified, deleted, moved, copied, or added.

- To view the file information, click the file name.
- To view the latest version of the file, click the file version.

6. To look at other items related to this commit, click the **ASSOCIATIONS** tab.

Associate code commits with other items

Create associations between code commits and other CollabNet TeamForge items, such as tracker artifacts or documents, to help define relationships, track dependencies, and enforce work flow rules.

For example:

- Associate a code commit with the bugs, feature requests, or other tracker artifacts that the code addresses.
- Associate a code commit with the task requiring its completion.
- Associate a code commit with an object in an integrated application.
- Associate a code commit with a requirements document.

Associate code with other items while committing

When you commit files to your source code repository, use the commit comment to quickly link your commit with one or more tracker artifacts or other TeamForge items.

Associations track the links between code and the bugs, feature requests, or other tracker artifacts that the code addresses. You can also associate code commits with other TeamForge items, such as tasks or documents.

A project administrator can make associations mandatory for all code commits. When this is made mandatory, the following additional rules pertaining to code commit can also be set:

- Code commits can be performed only for open artifacts.
- To perform a code commit, the committer must be the owner of the specific artifact.

☞ **Note:** Once you enforce the above rules, validations are strictly enforced for commits against tracker artifacts only. In case you commit against any other TeamForge object, for example a wiki or a document, mere existence of the object ID ensures successful commit and association and no validations are performed against the status of the artifact or who it is assigned to.

You can create tracker artifact associations from whatever interface you normally use to check code into your SCM repository. You do not have to log into TeamForge.

Use the same syntax for commits to CVS and Subversion repositories.

When making a code commit, add the associate command in the commit message like this: `[<item id>]`, such as the TeamForge tracker artifact ID or task ID.

- TeamForge item IDs are always letters followed by four or more numbers, such as `task1029` or `artf10011`.
- To associate a commit with multiple TeamForge items, separate the item IDs with commas.
- All associations are displayed in the **ASSOCIATIONS** tab of the **Commit Details** page.
- The **Comment** section lists the comments made with each commit.

☞ **Note:** To associate an object in an integrated application, use the `[<prefix_objectid>]` format. Each integrated application displays its prefix on moving the mouse over the application name.

☞ **Tip:** To remind yourself of the details of the association later, look in the **CHANGE LOG** tab of the associated **View Artifact** page.

Create associations with code that is already committed

At any time after a code commit is made, you can associate the code commit with other CollabNet TeamForge items, such as tasks, integrated application objects or documents.

1. Click **SOURCE CODE** from the **Project Home** menu.
2. On The list of project repositories, select the repository containing the code commit with which you want to create an association.
3. Click **View Commits**.
4. On the **Repository Details** page, click the name of the commit with which you want to create an association.

5. On the **Commit Details** page, click the **ASSOCIATIONS** tab.

6. On the list of existing associations, click **Add**.

7. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:

   - **ENTER ITEM ID** - If you know the item's ID, you can enter it directly.

     ☞ **Note:**

        - To associate an object in an integrated application from within TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked.
        - Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.
   - **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
   - **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.

8. Click **Next**.

9. You may add a comment in the **ASSOCIATION COMMENT** text box.

10. Save your work.

    - Click **Finish and Add Another** to add additional associations.
    - Click **Finish** to return to the **Details** page.

11. Click the **Associations** tab to view a graphical representation of all the associated items.

    Through the Association Viewer, you can choose to view associations in the form of a list or flip over to the Trace view to explore the layers of associations (including parent/child dependencies) laid out in a timeline. You can scroll across the Trace view by dragging the mouse over the association layer or use the 'Previous' and 'Next' arrows to view all the objects as events in a timeline.

    While the Associations tab shows the count of the total number of associations, you can only view the most recent 500 associations when you click the Associations tab in case the artifact has more than 500 associations. You can, however, browse through the Association Viewer to view older associations.

Create a source code repository

Each project can have one or more source code repositories.

Before you can create a source code repository, a site administrator must first add one or more SCM servers to the TeamForge environment.

1. Click **SOURCE CODE** from the **Project Home** menu.

2. In the list of the project repositories, click **Create Repository**.

3. On the **Create Repository** page, enter the directory name for the repository.

   ☞ **Note:** For CVS repositories, the directory name is the name of the directory relative to the CVS server's repository root directory. A UNIX group by the same name is also created to enforce permissions.

4. Enter a name and description for the repository.

   If you plan to use an SCM server that requires approval for new repositories, use the **DESCRIPTION** field to provide your reason for asking to create this repository.

5. Choose the server on which you want to create the repository.

   ☞ **Note:** The menu contains all of the SCM servers that the TeamForge administrators have added to the TeamForge environment.

6. If you want to require that each code commit be associated with an artifact (or a task or some other work item), select **Required on Commit**. Selecting this check box enables the following options:

   - **Artifact must be in open state:** Select this check box to ensure that users cannot perform a commit for closed artifacts. Therefore, if a user attempts to commit changes for a closed artifact, an appropriate error message is displayed.

     ☞ **Note:** When the commit log includes multiple artifact ids, then all artifacts must be in the OPEN state for the user to perform a commit.

- **Committer must Own Artifact:** Select this check box to ensure that only an artifact owner can perform the relevant commit. If the committer is not the artifact owner, then an appropriate error message is displayed preventing them from committing the changes to the specific artifact.

  ☞ **Note:**
  - In the case of a Git repository, the pusher who pushes the commits must be the owner of the artifact.
  - Once you enforce the above rules, validations are strictly enforced for commits against tracker artifacts only. In case you commit against any other TeamForge object, for example a wiki or a document, mere existence of the object ID ensures successful commit and association; no validations are performed against the status of the artifact or who it is assigned to.

7. For security reasons, you may want to restrict email notifications to the essential information. If so, select **HIDE DETAILS IN MONITORING MESSAGES**.

8. By default, the **Available in Search Results** check box is selected. Clear the check box if you want to exclude the repository from searches.

   ☞ **Tip:** You have the ability to control whether source code is included in search results. For example, you may want to exclude source code from searches that have sensitive information. You may also want to omit some repositories, such as the branding and publishing repositories, from search results. The **Available in Search Results** setting in **Create Repository** or **Edit Repository** page allows you to configure this. By default, this option is selected for all repositories, except for the **branding** repository in the **look** project and for **Publishing** repositories.

9. Click **Save**.

Your request for a new repository is submitted. You will receive an email notification when your repository is created or if your request for a new repository is denied.

- If the SCM server that you chose does not require approval for new repositories, the repository is created.
- If the SCM server that you chose requires approval for new repositories, a TeamForge administrator must approve your repository before it is created.

☞ **Note:** By default, repository requests on unmanaged CVS servers require approval, because the SCM administrator must manually create the repository.

Delete a source code repository

When you delete a repository, a request is submitted to the administrator for approval.

You need to have the required permission to delete SCM repositories.

1. Click **SOURCE CODE** from the **Project Home** menu.

2. In the list of the repositories, select the repository you want to delete and click **Delete**.
   The following confirmation message appears: `All SCM data in this repository will be lost. Are you sure you want to delete this repository?`

3. Click **OK** to delete.

Your request for deleting a repository is submitted. You will receive an email notification when your repository is deleted or if your request for deleting a repository is denied.

- If the SCM server that you chose does not require approval for deleting repositories, the repository is deleted right away.
- If the SCM server that you chose requires approval for deleting repositories, a CollabNet TeamForge administrator must approve your request to delete a repository before it is deleted.

☞ **Note:** By default, repository delete requests on unmanaged CVS servers require approval, because the SCM administrator must manually delete the repository.

Replicate a repository

When a Subversion Edge replica has been successfully registered with a TeamForge SCM integration server, it is available to project administrators in projects using that server to house repositories. To replicate a Subversion repository, you need to add it to a replica server.

Before you can replicate a Subversion repository, an administrator must first add one or more replica servers. This involves converting a *Subversion Edge server*, and then *approving the replica* in TeamForge.

1. Click **SOURCE CODE** from the **Project Home** menu.
2. In the list of project repositories, select the one you want to replicate and click **Edit**.

    The **Edit Repository** page lists the available replica servers. Here's an example:

Replica servers hosting repository: Desktop				
Status	Name	Hostname/IP address	Description	Owner
ⓘ No results found.				

Available Replica Servers					
	Status	Name	Hostname/IP address	Description	Owner
☑	☀	Tokyo	cu126.cloud.sp.collab.net	Replica in Tokyo office	Subversion Replication
					Replicate repository

    If you don't see any available replica server listed here, it may be because none were created for this Subversion server, or there are pending replicas which haven't yet been approved by a TeamForge administrator.

3. In the Available Replica Servers section, select a replica server and click **Replicate Repository**.
   You will see it in the list of replica servers hosting the repository.

- To see details such as the hostname and the user managing the replica repository, and recent replication command history, click the **Status** icon for the repository.
- For the Subversion checkout command, see the project's **Source Code** page.

    ☞ **Note:** The TeamForge account used while setting up the replica determines what's available to be checked out from a replica repository. This account - the **Managed By** user in the **Repository Replica Details** page - could have been provided total access to the master repository, or restricted access using path-based permissions.

Check command history

Recent command history for a replica server or a specific repository allows you to check for errors and see whether there are pending commands.

- Maybe a repository revision is not showing up. You can check for errors to know it's not just because the repository is still synching.
- You can also check if there are commands waiting in the queue -- you'd be able to see whether the repository is truly in synch.

- To check the command history for a replica repository, follow these steps:
    a) Click **SOURCE CODE** from the **Project Home** menu.
    b) In the list of repositories, click the icon for the one you're interested in.
       You'll see repository details and command history. Here's an example:

> **Tip:** You can also check command history by clicking on the **Status** icon in the **Edit repository** page.

* You need to be a TeamForge administrator to check the command history across a replica server.
  a) On the site administration navigation bar, click **INTEGRATIONS**.
  b) On the **SCM INTEGRATIONS** page, click the name of the replica server you're interested in.
     The **Edit System** page displays the command history.

Check out Subversion code anonymously

When you want to experiment with the code, you can do an anonymous checkout from the Subversion repository.

The checkout command uses a unique system-created user called "guest" and works without authentication.

To make anonymous checkout possible, the project administrator must set public and repository view permission to "All Users."

The checkout command differs based on whether a user is logged in or not.

* When you are not logged in, use this command to check out: `svn checkout --username guest <domain>/svn/repo URL name`
* When you are logged in, use this command to check out: `svn checkout --username <logged in_username><domain>/svn/repo URL name`

Access CVS with a stored SSH key

If you don't want to enter your password each time you access a CVS repository, you can create a set of SSH authorization keys and use the public-key method for automatic authentication.

Secure shell (SSH) can use public-key cryptography to confirm your identity, authenticate you to the remote host, and encrypt data transmission. On many TeamForge sites, access to CVS repositories is authenticated using SSH.

> **Note:** These instructions will help you only if all the following are true:
>
> * You are using CVS.
> * The `CVSROOT` (or "`cvs -d`") info given you by your admin (or the CTF "Repositories" list) does not include `:pserver:`.
> * Your admin has set up SSH access for you.
>
> Check with your site administrator if you are not sure about any of these conditions.

1. Generate an SSH key pair according to the instructions in the documentation for your SSH client.
   (Each SSH client provides its own mechanism for key pair generation.)
2. Log into TeamForge .
3. On your **My Workspace** page, select **MY SETTINGS** from the **My Page** menu.

4. On your **User Details** page, click **AUTHORIZATION KEYS** tab.

5. On the **Authorization Keys** page, copy the public key from the `.pub` file in your SSH installation directory and paste it into the **Authorized Keys** field.

6. Click **Update**.

Your SSH public key is now saved. When you log into a CVS repository on your TeamForge site, SSH automatically checks your private key against this public key and authenticates you. You do not have to enter a password.

☞   **Note:** You can store multiple SSH public keys in TeamForge, but only one for each client machine.

Manage binaries

TeamForge 8.0 and later support Nexus integration.

Before you can create binary repositories and work with them, a TeamForge site administrator must set up one or more Nexus servers and integrate the servers with TeamForge.

☞   **Important:** TeamForge-Nexus integration is possible on RHEL/CentOS platforms only. It's not supported in SUSE. See *Software requirements for CollabNet TeamForge 8.2* on page 748.

An overview of binary artifacts and repository managers

Here is the design overview of integrating TeamForge, a full ALM suite with binary repository managers.

An important aspect of the end-to-end development lifecycle is the creation and storage of software packages that are often binary artifacts. In the Java world, these are usually reusable jars that are used by other projects. Binary artifact repository managers are software systems that manage, version, and store binary artifacts. Examples of such repository managers are JFrog Artifactory, and Sonatype Nexus.

What is a binary artifact repository?

A binary artifact repository stores binary artifacts along with the metadata in a defined directory structure, conceptually similar to a source code repository. The metadata describes the binary software artifact and includes information such as dependencies, versioning, and build promotions. Maven is the widely used tool for dependency management, especially for Java projects. Maven represents dependencies in an XML file called Project Object Model (POM). Other tools can use similar approaches to store documentation archives, source archives, Flash libraries and applications, and Ruby libraries.

How does a binary artifact repository manager help?

Some of the advantages of using a binary artifact repository manager are:

- **Dependency management**: Nexus can act as a Maven repository. Maven is a widely used Java dependency management and build tool.
- **Efficient builds**: With the help of a binary artifact repository manager, you can save the download time from public repositories as the artifacts once downloaded are cached locally.
- **Predictability and release stability**: Once published onto a release repository, the binary artifact and metadata do not change. It ensures predictable and repeatable builds.
- **Control and audit**: If you want to standardize libraries that are used in your software, the binary artifact repository helps track the versions of your software components. Also it enables you to audit the licenses of your third-party components used in your software.
- **Promotes collaboration**: The binary artifact repository enables you to share components with other teams.

How to integrate Nexus with TeamForge?

☞   **Important:** TeamForge-Nexus integration is possible on RHEL/CentOS platforms only. It's not supported in SUSE. See *Software requirements for CollabNet TeamForge 8.2* on page 748.

TeamForge supports integration with Nexus in both the ALM and SCM modes. For the Nexus and Nexus professional versions 2.9, 2.10, and 2.11, the Nexus integration has been tested by CollabNet.

To integrate Nexus with TeamForge:

1. Download and install the Nexus OSS if you do not have a Nexus instance running.
2. Download and install the CollabNet Nexus integration plugin.
3. Change your build system and use the CollabNet supplied Maven deploy plugin for end-to-end traceability.
4. Set up the TeamForge Orchestrate activity source to provide your teams with end-to-end visibility from requirements to source code all the way to deployed binary artifacts.

Installing the TeamForge Nexus integration plugin

You need to have the following information handy before you start off with the installation:

- Installation path of the running Nexus instance.
- URL of TeamForge.
- TeamForge site administrator credentials.
- A suitable name for your Nexus instance; the Binaries App in TeamForge refers to this name.

You must have a Nexus instance running for the integration. If you are upgrading from an earlier version of the plugin, ensure that the old plugin is completely removed from the directory and the new plugin is unzipped on the same directory before you restart the Nexus instance.

- To set up a Nexus server, see *Install Nexus* on page 737.
- To install the TeamForge-Nexus integration plugin, see *Install the TeamForge-Nexus integration plugin* on page 738.
- To upgrade the TeamForge-Nexus integration plugin, see *Upgrade the TeamForge-Nexus integration plugin* on page 739.

Accessing Nexus through TeamForge: You have to introduce a TeamForge project context in Nexus and allow authentication to use TeamForge credentials for logging into Nexus directly. Accessing Nexus through the TeamForge project toolbar provides you with Single Sign-on (SSO). It logs you into Nexus automatically with the project context. You can allow RBAC (Role Based Access Control) using TeamForge roles.

Authentication policies

Your site administrator can enable the integration with the following two authentication mechanisms:

- TeamForge and native Nexus login (default)
- TeamForge only

In both the cases, you can use your TeamForge credentials to log on to Nexus. If your Site Administrator has used the default setup, you can use your pre-existing Nexus credentials.

Roles and permissions

Following are the two administrative privileges in Nexus:

- Nexus admin (Site Admin in TeamForge will be a Nexus admin)
- Project admin (permissions to create, update, and delete binary artifact repositories.)

For all the other users, privileges are based on the TeamForge RBAC setup.
Known limitations with TeamForge-Nexus integration
Here is a list of known limitations of TeamForge-Nexus integration.

- While it is possible for users (all TeamForge users but site administrators) with 'create repository' permission to create binary repositories via the TeamForge **Binaries** application, such users cannot create repositories directly on the Nexus server (using the Nexus UI) as there is no TeamForge project mapping available in Nexus.
- A TeamForge user cannot be configured as an anonymous user in Nexus as TeamForge users are not available in the Nexus database.
- TeamForge broadcast messages and license notifications are not visible in Sonatype Nexus pages in TeamForge. This is due to a limitation with the TeamForge-Nexus integration plugin.

Create a binary artifact repository
With TeamForge—Nexus integration enabled, each project can have one or more binary repositories.

☞ **Remember:**

Before you can create a repository for binary artifacts, a site administrator must set up TeamForge-Nexus integration and add one or more binary servers to your TeamForge site.

1. Click **BINARIES** from the **Project Home** menu.
2. Click **Create Repository**.
3. Type or select the values for the following fields in the **Create Repository** page.

   ☞ **Tip:** Hover your mouse over the Help icon (question mark icon) for relevant tooltip.

   • **Repository Name** and **Repository ID**: Type a name and ID for the binary repository.
   • **Server**: Select a binary repository server from the servers available in the drop-down list.
   • **Repository Type**: Select one of the following types from the drop-down list: Hosted Repository or Proxy Repository or **Virtual Repository**.
   • **Provider**: Select the content provider of the repository from the drop-down list.
   • **Repository Policy**: Repositories can store artifacts of a release or snapshot or both. Select a policy from the drop-down list.
   • **Override Local Storage Location**: Leave this blank to use the default local storage location. To change the local storage location, type the path (URL) in the following format.

     • **Windows**: `file:/{drive-letter}:/`
     • **Other operating systems**: `file://`

4. Modify the default access settings such as **Deployment Policy**, **Allow File Browsing**, **Include in Search** and **Publish URL**, if required.
5. Modify the **Not Found Cache TTL** duration (in minutes), if required.
6. Click **Save**.

The binary repository is created.

After creating a binary repository, you can now configure traceability for the repository (TeamForge Nexus) so that your Nexus runtime events are published to Orchestrate. See *Configure traceability for binary artifact repositories* on page 85.

Link binary artifact repository to project

You can link existing binary artifact repositories, if any, to your project.

1. Click **BINARIES** from the **Project Home** menu.
2. Click **Link Existing Repository**.
3. Select a repository from the list of binary repositories and click **Link to Project**.

Configure traceability for binary artifact repositories

Once a binary repository is created, configuring the repository to publish runtime data to Orchestrate is the next step that lets you track and visualize binary artifacts in real time using TeamForge Orchestrate.

Configure traceability for a binary repository.

1. Click **BINARIES** from the **Project Home** menu.
2. Select the binary repository for which you want to configure runtime.
3. Click **Configure Traceability**.
4. Click **Create**.
5. Change the **Associated Source** from the drop-down list and click **Update** to change the associated source.
6. Clicking **De-Activate** and **Re-Activate** disables and enables traceability for the repository respectively.
7. Clicking **Force Delete** deletes the traceability configuration including all historical associations. Exercise caution before deleting the traceability configuration.
8. Click **Close** to exit the **Traceability Configuration** window.

Configure Maven build

CollabNet provides you with a Maven deploy plugin that completely inherits the default (vanilla) Maven deploy plugin. Additionally, it posts binary artifact creation event information on to TeamForge Orchestrate enabling end-to-end traceability.

To configure your Maven build:

1. Replace the standard deploy with CollabNet's plugin in your POM.xml.

```
<pluginRepositories>
  <pluginRepository>
    <id>collabnet</id>
    <name>Collabnet Public Repo</name>
    <url>http://mvn.collab.net/nexus/content/groups/public/</url>
  </pluginRepository>
</pluginRepositories>
<plugins>
  ....
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-deploy-plugin</artifactId>
      <version>2.8.2</version>
      <configuration>
        <skip>true</skip>
      </configuration>
    </plugin>
    <plugin>
      <groupId>net.collab.maven.deploy</groupId>
      <artifactId>collabnet-deploy-maven-plugin</artifactId>
      <version>8.2</version>
      <extensions>true</extensions>
      <executions>
        <execution>
          <id>default-deploy</id>
          <phase>deploy</phase>
          <goals>
            <goal>deploy</goal>
          </goals>
        </execution>
      </executions>
      <configuration>
        <amqpServer>amqp://host.with.amqp</amqpServer>
        <sourceAssociationKey>as-per-orchestrate-setup</
sourceAssociationKey>
        <associatedBuildNumber>${env.BUILD_NUMBER}</associatedBuildNumber>
      </configuration>
    </plugin>
  </plugins>
```

☞ **Important:** Make sure the **<skip>** tag is set to true to prevent more than one Nexus notification for a single Nexus artifact deployment. If <skip> is not set to true, notifications are sent by both the maven-deploy-plugin and the collabnet-deploy-maven-plugin for a single binary artifact.

2. Setup the Nexus and Orchestrate credentials in settings.xml. You may find this file in the Maven home directory.
For example, in the following illustration, your distribution management section has a repository id of the local-nexus and the amqpServer, amqp://host.with.amqp is setup as mentioned in Step 1:

```
<settings>
 <servers>
  <server>
   <id>local-nexus</id>
   <username>your_ctf_username</username>
```

```
   <password>xxxxxxxx</password>
  </server>
  <server>
   <id>amqp://host.with.amqp</id>
   <username>myproj_nexus</username>
   <password>xxxxx</password>
  </server>
 </servers>
</settings>
```

If you already have a set of credentials against an AMQP URL in `settings.xml`, all jobs including the one that posts events against different sources can use the same credentials.

Table 1: All available configuration items:

Configuration parameter	Description	Mandatory	Default value	Example
amqpServer	An amqp url for the form `amqp://host.domain` where Orchestrate is listening to.	True	None	amqp://example.com
sourceAssociationKey	The source association key from orchestrate custom event setup.	True	None	uuid format
orchestrateSchemaId	The schema id posted to Orchestrate. Do not set it in v1.4.1 and later.	True	binary_artifact	binary_artifact
orchestrateSchemaVersion	The version of the schema, usually 1. Omit to default to 1. Do not set it in v1.4.1 and later.	False	1	1
amqpUserName	The AMQP user name. It is recommended not to set it in the project POM but using in settings.xml. Take it from the custom event setup in Orchestrate.	False	None	proj_custom
amqpPassword	The AMQP password. It is recommended not to set it in the project POM but using in settings.xml. Take it from the Orchestrate settings.	False	None	af78aaee0c5610d21...
amqpQueueName	Do not specify. Defaults to orchestrate.custom.	False	orchestrate.custom	orchestrate.custom
associatedBuildNumber	Specify to the env variable depending on your build system process. Set to ${env.BUILD_NUMBER} for jenkins and teamcity.	False	None	${env.BUILD_NUMBER}
skipOrchestrateNotification	Set to true to disable notification.	False	False	False
dieOnOrchestrateError	Defaults to true. Set to false to treat orchestrate notification errors as non-fatal.	False	True	True
component	Used to identify a specific binary artifact as a component in a larger application.	False	none	An ALM platform has several components such as an application

Configuration parameter	Description	Mandatory	Default value	Example
				server, an indexer, an SCM integration server and so on. These components have their own build process. This property is used to uniquely identify such components in Orchestrate.
componentOf	Associated with the 'component' parameter to store the details of the component.	False	none	SCM as a component of Teamforge.

Delete a binary artifact repository

You need to have the required permission to delete binary repositories. As a project admin, you can use the delete option only to unlink the repository from a project. If you want to delete a repository from the Nexus, login as a Nexus admin.

1. Click **BINARIES** from the **Project Home** menu.

2. In the list of the repositories, select the repository you want to delete and click **Delete**.
   The following confirmation message appears: `This repository will no longer be accessible. Your site administrator can relink it later. Are you sure you want to unlink this repository?`

3. Click **OK** to delete.

The repository is deleted.

☞ **Note:** This deletion disassociates the repository from your project; only the Site Admin can reinstate the repository.

Provision hardware

Use TeamForge Lab Management to access virtual or physical machines where you can build and test your code.

☞ **Tip:** For more information about the new features in TeamForge Lab Management 2.2, see *this introductory video*. (You'll need a free open.collab.net membership to watch it.)

Get an account on this system

If you are new to this system and require access, contact a TeamForge (or CEE) administrator in your organization.

The administrator will add a role to your TeamForge (or CEE) account which will enable you to access resources on the site.

What Lab Management systems are assigned to me?

The easiest way to see what Lab Management systems are assigned to you is to check the **My Pages** page.

Each system assigned to you is listed on the **My Pages** page, organized by project.

For more details on a particular system, click the name of the system. On the host status page, you can get more information about the host's configuration and performance, release the system, and more.

Find an available host to use

To work in the Lab Management environment, you need to either allocate a system to yourself or log into someone else's system in your project.

1. Click **My TeamForge Lab Management** or the **Projects** tab and select your project.

☞ **Important:** It is usually considered impolite, and may even be a violation of your company's policies, to check out a system from one project that you are a member of in order to do work related to another project.

2. Click **Project Hosts**.

3. In the list of hosts assigned to the project, do what you need to do:

   • If the host is allocated, click the user's name to ask their permission to work on their host.

   ☞ **Tip:** You may also ask the user if they still need the host, and would they mind releasing the system so that you or someone else, could check out the host.

   • If the host is in the **Free** state, allocate it to yourself.

   ☞ **Note:** It is polite to ask the owner of a machine if you can use it to log in and run jobs on. TeamForge Lab Management assumes that all TeamForge Lab Management users in a project should be able to log in to all systems in a project, to facilitate developer and QA productivity. But, along with this freedom comes the responsibility to be a "good citizen" and not step on other people's resources.

Allocate yourself a Lab Management machine

When you have been given membership and an appropriate role in a TeamForge Lab Management-enabled project, allocate a host for your own use.

Your personal host can be rebuilt with any profile whenever you want.

You can install any software you want on it (subject to the CollabNet Terms of Service and your own organization's security policy, all depending on how and where your TeamForge Lab Management instance is deployed).

You can run any jobs you want on the server (subject to the same conditions as software installation).

1. Find an available host to allocate.

2. On your project's TeamForge Lab Management home page, click the host you've identified.

3. On the host status page, click **Allocate** to reserve the host for yourself.

4. Specify how long you wish to allocate the host.

   • Choose a specific time/date to deallocate the host using the calendar widget.
   • Choose a length of time to allocate the host.
   • Allocate the host indefinitely (until you or a project admin frees the host).

   ☞ **Note:** Your project administrator may have set a limit on the amount of time a machine can be allocated. If this is the case, then you will receive an error message if you try to allocate a host for a longer period.

You can always change the allocation time for your host later, if you decide that you need the host for a shorter or longer amount of time (subject to the maximum limits set by your project administrator).

Compare profiles or profile versions

To choose a profile for your TeamForge Lab Management system, compare different profiles and profile version in the Profile Library.

1. On your project's home page, select **Profile Library**.
   The table shows all profiles in your project.

2. Choose a profile from each of the sets of profiles and click **Compare selected profiles**.

   ☞ **Note:** Profiles must first be added to your project as "allowed profiles". It is not possible to compare profiles before you have added them to your project.

3. On the **Profile Compare** screen, select the versions of each profile to compare.

   The default version is the latest version of the profile, but any version of a profile can be compared against any version of any other profile.

   ☞ **Note:** Any two profiles can be compared, even if they are from two different operating systems. For example, TeamForge Lab Management will let you compare a Solaris profile and a Windows XP profile (but do not expect the results to be very meaningful).

Create a profile from a host

Let's say you've built a generic Lab Management host and customized it for your team. With Lab Management 2.4, you can create a profile based on the host image and store it in the Project Build Library (PBL) for your project.

☞ **Note:**

- You can only create profiles from ESX guests.
- These OS profiles are supported: RHEL 5, CentOS 5, SLES/11, Windows XP and And Windows Server 2003. Windows Server 2008 and Solaris profiles are not supported.

When creating a profile that contains user-specific software or services, here are a couple of things you should be aware of:

- If the software or service is run by a user who is a locally-created user (say, an 'oracle' user who was manually created on the node), the profile should be easily portable to other projects.
- If the software or service is run by a user who is a Lab Management user, the profile will only work in projects where that user has access. In projects where that user has no access, it's not likely that the profile will work well - it may not work at all.

1. Bring up your host's **Admin** tab.
2. In the **Create Profile from This Node** section, enter a name for the profile.

   Only alphanumeric characters, hyphens, underscores, and periods are allowed.

3. Provide a description for the profile.

4. Select the PBL location for the project.

   The profile will be stored here.

5. Click **Create Profile from <hostname>**.

Your host will be switched to a clone, and inaccessible for few minutes. When it gets back to the allocated state, the profile will be created. The profile will be available in your project and in any project that allows public profiles.

Rebuild a system

You can rebuild a host with a specific profile that you choose from within your project.

Usually you will want the latest (highest) revision of the profile you are working with. However, profiles are versioned, like source code. This means that you can always build with an older version of an operating system or package if you need to.

☞ **Note:** Profiles are assigned to projects by your project administrator.

1. On the **Project Host properties** screen, select the host to rebuild.
2. Click the **Configuration** tab.
3. Select a profile from the **Profile** drop-down list and click **Rebuild**.

You can check the status of the host's progress by refreshing your browser.

Log into a TeamForge Lab Management system

You can log into any machine that is assigned to a project you are a member of and that you have the requisite permissions for.

For security reasons, all TeamForge Lab Management systems are on a private network, inaccessible from anywhere outside of CollabNet. To access these systems, you must tunnel through the TeamForge Lab Management Manager.

If you need access to a system that belongs to a project you are not a member of, you have two options:

- Ask to join the project, or to get permission within that project to access TeamForge Lab Management hosts.
- Ask the owner of the system, or a project/domain administrator, to create an account for you on the machine.

> ☞ **Note:** This may violate your organization's security policy, so check with your site administrator before creating any accounts. If the account is temporary, you will be responsible for removing it.

Log into a UNIX system

To log in to UNIX systems, such as Linux and Solaris machines, you must use port forwarding over SSH.

Port forwarding over SSH is a secure way to tunnel traffic over the Internet.

1. Set up your SSH tunnel.

   a) From a Linux or Solaris system, or from a Windows system with Cygwin, the command to set up an SSH tunnel to "cu086.$vars.mgr_domain" through the TeamForge Lab Management Manager host "$external_host" would look like this:

   ```
   ssh -L22000:cu086.$vars.mgr_domain:22 $external_host -l username
   ```

   b) From a Windows system without Cygwin installed, set up the tunnel using PuTTY.

   All connections to "localhost" port number 22000 are now tunneled to "cu086.$vars.mgr_domain" port number 22 (SSH) through the TeamForge Lab Management manager host "$external_host"

2. Log into "cu086.$vars.mgr_domain" by issuing this command:

   ```
   ssh localhost -p 22000 -l username
   ```

Log into a Windows system

To log into a Windows system in TeamForge Lab Management, you must use SSH with the Remote Desktop Protocol (RDP), which renders a full Windows graphical environment on your desktop.

The TeamForge Lab Management environment includes Windows XP Professional and Windows 2003 Server machines.

> ☞ **Tip:** You can automate this whole process with a tool called Wissh. CollabNet does not provide support for this tool, but it is something which we recommend.

1. Set up your SSH tunnel to the TeamForge Lab Management machine.

   a) If you have Cygwin installed, use a command like this. For example, "cu087.$vars.mgr_domain", through the TeamForge Lab Management Manager host "$external_host"

   ```
   ssh -L3390:cu087:3389 $external_host
   ```

   b) If you do not have Cygwin, set up the tunnel using Putty.

2. Start Remote Desktop by clicking **Start** > **Programs** > **Accessories** > **Communications** > **Remote Desktop Connection**.

   > ☞ **Note:** If you are using Linux to access a Windows system, you can use the `rdesktop` program, which provides the same functionality.

3. In the Remote Desktop Connection dialog box, enter `localhost:3390` and click **OK**.

   This initiates a remote desktop connection to the TeamForge Lab Management host.

4. At the login prompt, enter your TeamForge Lab Management username and password.

If you are an authorized user on the machine, you are granted access and the correct privileges.

5. If your remote Desktop Connection to `localhost` does not work, try this workaround:

    a) Copy the two files `mstsc.exe` and `msctax.dll` to another folder.

        These files should be in the directory `C:\WINDOWS\system32`, or in `C:\Program Files\Remote Desktop` if you installed the Remote Desktop Client from the Windows 2003 CD.

    b) Right-click the `mstsc.exe` file and select **Properties**.

    c) On the **Compatibility** tab, select **Run this program in compatibility mode for Windows 98**.

    d) Start up the new copy of `mstsc.exe` from the new location.
        You should now be able to connect to `localhost` at IP address 127.0.0.1.

Set up a tunnel with PuTTY

Summary to come...

1. Launch the PuTTY application.

2. For **Host name**, enter the TeamForge Lab Management Manager host "$external_host".

3. For **Connection type**, enter "SSH".

4. In the left pane, open **Connection list**.

5. Open **SSH list** and click **Tunnels**.

6. In the right pane, enter "3390" for **Source port** and "cu087.$vars.mgr_domain:3389" for **Destination**, and click **Add**.

7. Select the forwarded port from the list and click **Open**.

8. At the login prompt, enter your TeamForge Lab Management username and password.

The RDP tunnel is now set up.

☞   **Note:** This procedure is equivalent to running this SSH command:

```
ssh -L3390:cu087.$vars.mgr_domain:3389 $external_host
```

Set up a VNC server on TeamForge Lab Management

To allow remote access to an X Windows display on Linux from a remote computer, set up a VNC console.

VNC is a convenient and effective way to access an X Windows display from a remote computer. Some applications require X, and since users don't have access to consoles on TeamForge Lab Management nodes, VNC might be the only way those applications can be accessed.

1. Set a VNC password using the `vncpasswd` command.

```
[grue@cu012 grue]$ vncpasswd
Password:
Verify:
```

☞   **Tip:** This doesn't need to be the same as your TeamForge Lab Management password, but keep in mind that this password will give the bearer access to an X session as your username, so keep it something reasonably hard to guess.

2. Edit the file `/etc/sysconfig/vncservers`. Add an entry for your session number, defaulting to 1, followed by your username.

    Use this as an example file:

```
VNCSERVERS="1:grue"
```

3. Start the VNC service.

```
[grue@cu012 grue]$ sudo /etc/init.d/vncserver start
```

```
Starting VNC server: 1:grue        [  OK  ]
```

☞ **Note:** To stop the VNC server on your machine, run the command like this: `[grue@cu012 grue]$ sudo /etc/init.d/vncserver stop`

4. If you want VNC to start up each time this server is rebooted, use the `chkconfig` command to turn it on.

```
[grue@cu012 grue]$ sudo /sbin/chkconfig vncserver on
```

☞ **Note:** To prevent the VNC server from starting up on your machine when it is rebooted, run the command this way: `[grue@cu012 grue]$ sudo /sbin/chkconfig vncserver off`

Connect to a VNC server from your desktop
To access an X Windows display from a remote computer, install and connect a VNC client.

VNC saves your desktop session and allows you to connect to it later. For example, you can start an X session at work, then log off your computer, go home, start up your computer again, and resume your same X session. In this sense it is very similar to the `screen(1)` utility, which allows users to create, disconnect and reconnect to text-based terminal sessions.

1. Install a VNC client on your computer.

☞ **Tip:** Either of these recommended packages is available for all major desktop operating systems.

- TightVNC
- RealVNC

2. Set up an SSH tunnel to port `5901` on the proper TeamForge Lab Management node.

For example, let's say you are the user `grue` and you have set up a VNC server on the machine `cu012. $vars.mgr_domain`. The port forwarding command would look like this:

```
ssh -L5901:cu012.$vars.mgr_domain:5901 $external_host
```

☞ **Note:** Multiple `-L` options can be combined on the same command line.

3. Start your VNC client software, and give the following information:

- Hostname: `localhost`
- Port: `5901`

Your VNC desktop session appears.

When you are done with the session, you can close the VNC client application on your desktop.

☞ **Important:** Keep in mind that closing the VNC client does not close applications that are currently running. You can connect to this session again at any time later and your complete session is saved.

For security purposes, we recommend that you do not keep any windows open which contain root shells or other software running as the `root` user. This will mitigate the risk of someone being able to access your VNC session without you knowing it.

Transfer files in and out of the system
You can use WebDAV, SFTP or Windows RDP to transfer files in and out of a TeamForge Lab Management system.
Transfer files with WebDAV
The most useful way to get files in and out of the TeamForge Lab Management environment is via WebDAV.
Connect to WebDAV with Windows Explorer on Windows XP
With a software update, Windows can be used to connect to a DAV share on a TeamForge Lab Management system.

Microsoft recommends you install this *Software Update for Web Folders* before using Web Folders. You may need this update for Web Folders to work properly.

1. Launch Windows Explorer.
2. On the **Tools** menu, click **Folder Options** and select the **View** tab. Clear **Hide extensions for known file types**. This will make file names that begin with a dot (".") to appear properly (for example, your `.bashrc` file).

3. Launch Internet Explorer.

4. On the **File** menu, click **Open**.

5. Enter the URL for your home directory, which will always end in `/home/USERNAME`.
   For example:

   `https://$external_host/home/user`

6. Select **Open as Web Folder**.

7. Enter your user name and password when prompted.

8. If the connection is successful, you will see a new entry added to *My Network Places*. You can re-connect to this folder in the future by going to *My Network Places* and selecting this folder.

As of this writing Microsoft does not provide a client that can mount a Web Folder as a drive letter using an encrypted, authenticated connection, but a third-party tool called Webdrive can do this.

Connect to WebDAV with Mozilla Firefox on Windows XP

Firefox isn't a WebDAV client, but it will interface with the Windows native WebDAV client to open up WebDAV shares.

1. Install the *openwebfolder* Firefox plug in and restart Firefox.

2. Log into TeamForge Lab Management and go to the **Summary** page of any host you have access to.

3. Under **Access Options**, right-click the link pointing to your home directory and select **Open As Webfolder**.

4. Enter your user name and password.

Your WebDAV home directory is now mounted.

Connecting to WebDAV with Mac OS X

To connect to a TeamForge Lab Management system from a Mac, open up your TeamForge Lab Management home directory via DAV access.

This procedure has been tested with Mac OS X 10.4.8 (Tiger) and should work for all Mac OS X versions.

1. In the **Finder**, click **Go** > **Connect to Server**.

   ☞   **Note:** You can also press **cmd-K** on the keyboard.

2. For **Server Address**, enter the URL corresponding to your home directory.
   For example, `https://$external_host/home/user` (where "`user`" is your user name).

3. Click **Connect**.

4. If prompted, click **Continue**.

5. In the **Verify Certificate** dialog, enter your TeamForge Lab Management user name and password.

6. Select **Remember this password in my keychain** if you want to store your user name and password locally.

7. Click **OK**

You should now see your folder.

   ☞   **Note:** Files and directories beginning with a "." are not shown in the graphical folder view, but are accessible via the command line in OS X.

Connect to WebDAV with Nautilus on Linux

The Nautilus file browser -- the default file browser on distributions like Red Hat, Fedora and Ubuntu -- has built-in support for DAV folders.

1. From the main desktop, go to **Places** > **Connect to Server**.

2. For **Service type**, select **Secure WebDAV (HTTPS)**.

3. For **Server**, enter the name of the TeamForge Lab Management server, e.g., `$external_host`

4. For **Folder**, enter `/home/user`, where "`user`" is your user name.

5. You can leave the **User Name**, **Port**, and **Name to Use For Connection** fields blank.

6. Click **Connect**.

This creates an icon on the desktop, which brings up your home directory when accessed.

Transfer files with SFTP
Secure File Transfer Protocol (SFTP) can be a useful way to transfer files between your desktop and TeamForge Lab Management.

SFTP is an extension to the Secure Shell (SSH) protocol, which emulates File Transfer Protocol (FTP). File transfers made with SFTP show up immediately in your TeamForge Lab Management environment. Any TeamForge Lab Management client machines you are logged into immediately reflect changes made to your home directory from your desktop.

1. On Linux, UNIX, or Windows with Cygwin, run the command-line SFTP program with a command like this:

```
sftp username@$external_host
```

☞ **Note:** There are also graphical clients available for Linux.

2. On Windows, we recommend using the Filezilla program to manage your SFTP transfers.

   Filezilla is available as a plugin for the Firefox web browser with an easy-to-use Explorer-like interface.

Connect to TeamForge Lab Management with Windows Remote Desktop
If you are connecting to a Windows host via Remote Desktop, you can share drives from your local system to the remotely connected system.

1. On the **Start** menu, click **All Programs** > **Remote Desktop Connection**.
2. Click **Options** and click the **Local Resources tab**.
3. Click **Disk Drives** > **Connect**

Find a build
You can search for file names (not file content) in the Project Build Library.

Search for files in Project Build Library using these fields:

- Project
- User
- File Name: Shell-like regular expressions are permitted. For example, for searching all RPMS you can type `*.rpm`.
- Description: Substring-like pattern matching is permitted. For example, to search all files added by the `pbl_fsck` script, type `pbl_fsck`.

☞ **Note:** Exclude directories from your search results with the **Exclude** option.

Connect to network services outside TeamForge Lab Management
TeamForge Lab Management systems are firewalled from the outside world, for our users' safety. However, you can use port forwarding to connect to machines outside of TeamForge Lab Management.

For example, you may need to connect to a web server holding the results of your nightly builds, or access an external web server where libraries are installed.

Reach a TeamForge Lab Management server from the outside
Use port forwarding to allow users outside the TeamForge Lab Management environment to access specfic ports on specific hosts inside the TeamForge Lab Management network.

Port forwarding must be set up by a TeamForge Lab Management Domain Administrator: neither you nor your project administrator can do this. This is necessary because port forwarding can open up security holes.

⚠ **Caution:** Exposing an insecure network service outside of the TeamForge Lab Management environment could put your whole project and the entire TeamForge Lab Management environment at risk, if that service is compromised.

Contact your Domain Administrator and give them the following information.

- Your name.
- Your project.

- The host you want to expose a port from.
- The port number you would like to expose.
- The service you would like to run on this port.
- Security precautions you have taken to insure that this service is safe to expose outside of TeamForge Lab Management.

☞ **Note:** Your Domain Administrators may have more stringent information gathering or signoff processes which you will be subject to.

Reach outside destinations from within TeamForge Lab Management

TeamForge Lab Management has no built-in features to allow outbound network access to be granted by an administrator. However, if you need access to external destinations from inside TeamForge Lab Management, you have a couple of options.

Try one of these approaches:

- Contact your Domain Administrator and ask CollabNet to open up a hole in the firewall for this purpose.
- Get the files you need into your home directory from outside CollabNet, and access them that way.

Allocate a host from a cloud

To allocate a host from a cloud into a project, specify its type, size and profile information.

In Lab Management, you can allocate from virtual machines on physical hosts and EC2 hosts. You cannot allocate a physical machine from a cloud

1. In your TeamForge Lab Management project, click **Hosts**.

2. Click **Allocate** for the cloud from which you want to allocate your host.

   The clouds you can choose from depend on the ones available in the Lab Management domain, and which of these the administrator has made available for your project.

3. In the **Clouds** page, select a host type and size.

   Different host types have different attributes such as CPU, disk size and RAM. They also have different costs and available profiles.

4. Select the system profile you want on your host.

   A profile in Lab Management is a collection of software packages that comprise a runnable operating system and accompanying set of applications.

   ☞ **Note:** Profiles have different costs. For example, a profile might include some licensed software that adds to the cost.

5. Select the number of hosts you want to allocate.

   The default is one.

6. To customize your host, click **Advanced options**.

   a) Enter a description for the host.

   b) Select whether you want the host allocated up to a certain date, or for a specific number of days.

      By default, the host is allocated indefinitely until you free it.

7. Click **Allocate Host**.

☞ **Tip:** You can see information about the host in your **My Pages**.

Build and test your software

When your project manager makes CollabNet Build & Test available, you can set up the CollabNet plugin for Hudson (or Jenkins) in your development environment.

For more information, please see:

- *Hudson wiki for the CollabNet plugin*
- *Jenkins wiki for the CollabNet plugin*

Release products

Publish the ouput of your project to selected audiences as packages and releases.

Download a release

Downloading a release brings all the release's files to your local machine in a single `.zip` file.

1. Click **FILE RELEASES** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release you want to download.
3. On the **List Releases** page, click the title of the release.
4. On the **View Release** page, click **Download Selected**.

TeamForge prepares the compressed file in `.zip` format.

You are prompted to open or save the file.

Create a package

A package is a folder into which one or more related releases are published.

For example, you might create a package to represent a product deliverable or major component. You can then create releases within the package for product builds or other groups of files.

☞ **Note:** A package must exist before you create the releases and individual files that will go into the package.

1. Click **FILE RELEASES** from the **Project Home** menu.
2. On the **File Release Summary** page, click **Create**.
3. On the **Create Package** page, provide a title and description for the package.
4. Click **Save**.

The package is created.

When you have created a package, you can publish a release into the package.

Create a release

A release is a group of one or more files that are published as a unit.

Before creating a release, you must have a package into which your release will go.

1. Click **FILE RELEASES** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package in which you want to create the release.
3. On the **List Releases** page, click **Add**.
4. On the **Create Release** page, provide a name and description for the release.
5. Set the status of the release.

Pending	Releases with pending status are not visible in the drop-down lists displayed when you set **Reported In Release** and **Fixed In Release** fields in an artifact. Use **Pending** status when you have created a release but have not yet finished adding files.
Active	Releases with active status are visible in the drop-down lists displayed when you set **Reported In Release** and **Fixed In Release** fields in an artifact.

6. Identify the maturity level of this release.
7. Click **Save**.

The release is created. You can begin adding files.

☞ **Tip:** To facilitate tracking, you may want to match this release to the planning folder that tracks the work that's going into the release. If you do that, the relevant work items will automatically appear on the **PLANNED TRACKER ARTIFACT** tab. See *Create a planning folder* on page 17.

Add files to a release

After you have created a release, you can add one or more files. All files in a release are published as a unit.

Project members can download the entire release in a single `.zip` file, or download only selected files.

1. Click **FILE RELEASES** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release.
3. On the **List Releases** page, click the title of the release.
4. On the **View Release** page, click **Add**.
5. On the **Add File** page, choose the desired file, then click **Save**.
   The file is added and you are returned to the **View Release** page.
6. Repeat the last two steps until all files are added.

After you have added your files, you might wish to change the status of the release from pending to active. This will allow users with appropriate permissions to select the release when entering or updating an artifact. You can also change the maturity level if needed.

Update files in a release

You can replace an existing file in a release with a new file.

1. Click **FILE RELEASES** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release.
3. On the **List Releases** page, in the **Releases** section, click the title of the release containing the file that you want to update.
4. On the **View Release** page, select the file you want to update, and click **Update**.
5. In the **Update a File** page, go to the new file with which you want to replace the current file.
6. Click **Save**.

The file is now updated.

Delete files from a release

If a file is no longer needed, it's a good idea to delete it from the release.

1. Click **FILE RELEASES** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the desired release.
3. On the **List Releases** page, in the **Releases** section, click the title of the release containing the file that you want to delete.
4. On the **View Release** page, choose the file you want to delete, and click **Delete**.

The file is deleted.

Update release attributes

As your release matures, you should update its maturity level and status.

1. Click **FILE RELEASES** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release.
3. On the **List Releases** page, in the **Releases** section, click the title of the release containing the file that you want to delete.
4. On the **View Release** page, click **EDIT**.
5. On the **Edit Release** page, choose a new status for the release.
   A release can be in Active or Pending status.
6. Select the maturity level of this release from the **MATURITY** field.
7. Click **Save**.

Change a package's name or description

When the purpose or the audience for a package shifts, you may want to rename the package or describe it differently.

1. Click **FILE RELEASES** from the **Project Home** menu.
2. On the **File Release Summary** page, select the package you want to edit, and click **Edit**.
3. On the **Edit Package** page, provide the new name or description.
4. To make it easier for users to download the package, select **SHOW DOWNLOAD LINK IN PROJECT LIST**. This makes the download link show up in the project's entry in the Project Categories system, if the project has been put in one or more categories. See *Categorize a project* on page 250.
5. Click **Save**.

Associate a release with other items

If a file release is related to other TeamForge items, such as documents, tasks, tracker artifacts, integrated application objects, or news items, you can connect the file release to the other item by creating an association.

Creating associations between items helps you define relationships, track dependencies, and enforce work flow rules. Some example uses of file release associations include:

*   Associate a release with a requirements document.
*   Associate a beta release with a task related to managing the beta program.
*   Associate a release with tracker artifacts representing bugs fixed in the release.

> 👉 **Note:** You can also associate tracker artifacts such as bugs and feature requests with the related file releases. You do this as part of working with the tracker.

1. Click **FILE RELEASES** from the **Project Home** menu.
2. On the **File Release Summary** page, click the title of the package containing the release with which you want to create an association.
3. On the **List Releases** page, click the name of the release.
4. On the **View Release** page, click the **ASSOCIATIONS** tab and click **Add**.
5. In the **Add Association Wizard** window, select the items with which you want to associate the artifact:

    *   **ENTER ITEM ID** - If you know the item's ID, you can enter it directly.

        > 👉 **Note:**
        >
        > *   To associate an object in an integrated application from within TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked.
        > *   Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.
    *   **ADD FROM RECENTLY VIEWED** - Select one of the last ten items you looked at during this session.
    *   **ADD FROM RECENTLY EDITED** - Select one of the last ten items you changed.
6. Click **Next**.
7. You may add a comment in the **ASSOCIATION COMMENT** text box.
8. Save your work.

    *   Click **Finish and Add Another** to add additional associations.
    *   Click **Finish** to return to the **Details** page.
9. Click the **Associations** tab to view a graphical representation of all the associated items.

    Through the Association Viewer, you can choose to view associations in the form of a list or flip over to the Trace view to explore the layers of associations (including parent/child dependencies) laid out in a timeline. You can scroll across the Trace view by dragging the mouse over the association layer or use the 'Previous' and 'Next' arrows to view all the objects as events in a timeline.

    While the Associations tab shows the count of the total number of associations, you can only view the most recent 500 associations when you click the Associations tab in case the artifact has more than 500 associations. You can, however, browse through the Association Viewer to view older associations.

Delete a package

If you no longer need a package, you can delete it.

Deleting a package deletes all of the releases and files within it.

☞ **Important:** Delete a package only if you are sure that you no longer need any of the releases and files within it.

1. Click **FILE RELEASES** from the **Project Home** menu.

2. On the **File Release Summary** page, choose the package that you want to delete.

3. Click **Delete**.

The package and all of the releases and files within it are now deleted.

Delete a release

If you no longer need a release, you can delete it.

Deleting a release deletes all of the files within it.

☞ **Important:** Delete a release only if you are sure that you no longer need any of the files within it.

1. Click **FILE RELEASES** from the **Project Home** menu.

2. On the **File Release Summary** page, click the title of the package containing the release that you want to delete.

3. On the **List Releases** page, in the **Releases** section, select the release that you want to delete, and click **Delete**.

The release and all of the files within it are now deleted.

Watch for changes

When you monitor an item, such as a task or a document, you are notified by email when that item is updated. You can also monitor a discussion forum or project news by subscribing to its Really Simple Syndication (RSS) feed.

Sort emails from TeamForge automatically

When you are involved in a lot of activities on your TeamForge site, you can start receiving many emails from project members and administrators. To save time, you can configure your email application to sort these emails automatically.

For example, you are automatically emailed whenever someone comments on an artifact that you created. You may also want to join the monitoring lists of items you are interested in, and other team members can add you to the monitoring list of items they create. All of these things can add up to a good number of emails after you have been on a project for a while.

Your email program, such as Microsoft Outlook and Mozilla Thunderbird, looks for the X-headers in incoming email messages and acts on them according to your instructions.

1. In your email client, create a message filter.

    *   In Mozilla Thunderbird, click **Tools** > **Message Filters**.
    *   In Microsoft Outlook, click **Tools** > **Rules and Alerts**.

2. Specify the type of information you want to filter.

    For example:

    *   Use `X-TeamForge-ProjectId` to identify the project that the incoming email is related to.
    *   Use `X-TeamForge-FolderId` to identify messages related to a specific tracker or document folder.
    *   Use `X-TeamForge-AssignedTo` to flag tasks or artifacts that are assigned to you.
    *   Use `X-TeamForge-CommitPath` to watch for commits path in a Subversion repository where you are working.
    *   Use `X-TeamForge-PlanningFolder` to watch for artifacts to be added to or removed from a planning folder you are working with.

3. Tell your email program what to do with the messages with the X-header that you specified.

    For example:

    *   Automatically move emails into a special "To Do" folder if they are about tracker items assigned to you.
    *   Color emails red if they are about commits to the Subversion repository path you specified.
    *   Forward all build-related announcements to your mobile phone for reviewing later.

You can use as many X-headers as you need. If you have set them up judiciously, this can decrease the amount of time you need to work through your incoming email.

Monitor an item

To get notified by email whenever an item changes, monitor the item.

In TeamForge, an item is something you produce with the site's tools. These are some examples of items:

- Documents
- Tasks
- Tracker artifacts
- Discussion forum topics
- Files in a release

1. Go to the folder containing the item or items that you want to begin monitoring.
2. Select the item or items that you want to begin monitoring.
3. Click the **Monitor** down arrow, then choose **Monitor Selected**.

You are now monitoring all selected items. The icon, 🔔 is displayed in the item list view and the monitored items appear in the monitored items list on your personal **Monitoring** page.

☞ **Note:** When you update two or more artifacts at a time, each user who is monitoring any of the changed artifacts gets a single email describing all the updates.

To stop monitoring an item, select it, then roll your mouse over the **Monitor** down arrow and choose **Stop Monitoring Selected**.

☞ **Note:** You can also stop monitoring any item from the monitored items list on your personal **Monitoring** page.

Monitor many items

To get notified about a whole class of items whenever one is created or changed, monitor the folder that contains the items.

In TeamForge, a folder is a container for multiple items. Any container can be considered as a folder, even if it is not explicitly called a folder.

- A document folder contains documents.
- A task folder contains tasks.
- A tracker is a folder that contains tracker artifacts.
- A forum is a folder that contains discussion topics.
- A repository is a folder that contains code commits.
- A package is a folder that contains files.

1. Select the folder that you want to begin monitoring.
   For example, in a project where you are a member, click **SOURCE CODE** from your **Projects Home** menu and select one of the code repositories in the project.
2. Click the **Monitor** down arrow, then choose **Monitor Current Folder**.

You are now monitoring the folder. The icon, 🔔 is displayed in the item list view and the monitored items appear in the monitored items list under the **MONITORING** menu available in the **My Workspace** page.

☞ **Note:** You do not receive monitoring notifications for changes that you yourself make to an item in a monitored folder.

To stop monitoring a folder, click the **Monitor** down arrow, then choose **Stop Monitoring Folder**.

☞ **Note:** You can also stop monitoring any item from the monitored items list under the **MONITORING** menu available in the **My Workspace** page.

Monitor an application

To monitor the entire application, such as all trackers or all documents in a project, select it from the **MONITORING** tab available in the **My Workspace** page.

Monitoring an application keeps you updated on all items and folders within the application. Unlike individual items, lists of items, and folders, you don't monitor entire applications from within a project.

1. Click **MONITORING** from the **My Page** menu.
2. From the **Edit Monitoring Subscriptions and Preferences** pane, on the **My Workspace** page, choose the project in which you want to monitor an application.
3. Click the **MONITORED APPLICATIONS** tab and select the applications that you want to begin monitoring.
4. Click **Save**.

You are now monitoring the selected applications.

Monitor discussion forums as RSS feed

To get an update in your RSS feed reader each time there is an exchange of ideas in a discussion forum, subscribe to the forums's RSS feed.

In TeamForge, you can subscribe to discussion forums via RSS feeds.

1. Click **DISCUSSIONS** from the **Project Home** menu.
2. On the **Forum Summary** page, click the name of the forum that you want to subscribe through RSS feed.
3. On the **Topic Summary** page, click the RSS feed icon .

You can now monitor all topics in the selected discussion forum using your RSS feed reader.

☞ **Note:** You can access the project messages via the RSS reader without logging in to TeamForge, so long as you have the discussion view permissions.

Monitor project news as RSS feed

To get an update in your RSS feed reader each time something important is announced in your project, subscribe to the project news as RSS feeds.

In TeamForge, you can subscribe to project news for all the projects via RSS feeds but not to the news of any specific project.

On the **NEWS** tab of **My Page**, click the RSS feed icon.

☞ **Note:** You must be a member of a project to view its news.

You can now monitor all project news announcements regarding your projects using your RSS feed reader.

☞ **Note:** You can access the project news via the RSS reader without logging into TeamForge, so long as you have project membership in any project.

What am I monitoring?

All of your monitored items appear under the **MONITORING** tab available in the **My Workspace** page.

From this list, you can view or stop monitoring any item you are currently monitoring. You can also monitor entire applications from this page.

1. Select **MONITORING** from the **My Page** menu.
   Your personal monitoring page lists all items you are currently monitoring.
2. Specify the filter criteria in one or more filter fields (at the top of each column) and click **FILTER**.

   • You can find a filter field at the top of each column in most of the tables in the TeamForge application.
   • The filter field could be a text box or a drop-down list with multi-select check boxes.

- You can type your filter criteria in the text boxes. The search text is case-insensitive.
- You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the *FILTER_DROPDOWN_MAX_SELECTION* token in the `site-options.conf` file to increase or decrease the count.
- **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing "R" instantly shows all statuses having the alphabet "R". The search text is case-insensitive.

- Some search filters may not appear if your site administrator has not enabled them.

3. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.
4. Use the up-down arrow at the top of any column to sort your list by that column.

   - Your primary sort column is identified by a superscript **1** next to the up-down arrow, and your secondary and third-level sort columns, if any, are likewise marked.
   - Click the up-down arrow again to reverse the sort order.

To stop monitoring an item from your personal monitoring page, select the item you want to stop monitoring, then click **Stop Monitoring**.

Who is monitoring this item?

To see who is monitoring an item or folder, check the **Users Monitoring This Item** list.

1. Go to the page where the item appears.
2. Click the item to view the artifact.
3. On the item's **View Artifact** page, click **Users Monitoring**.

The **Users Monitoring This Item** window displays a list of all users who are monitoring the item.

Choose monitoring email frequency

How often do you want to receive monitoring email notifications for all the applications, folders and items you are monitoring?

1. Select **MY SETTINGS** from your **My Page** menu.
2. On your **User Details** section, in the **USER PREFERENCES** tab, choose an email notification preference.

   - **Email Per Change**: Get a separate email notification for each change to a monitored item.

- **Daily Digest Email**: Get one email notification each day containing a digest of all changes made to monitored items in the preceding twenty-four hours.
- **Don't Send Email**: Get no email notifications for changes to monitored items. This can be handy when you are on vacation.

3. Click **Save**.

After you have set your global email frequency, you can further customize the frequency of application monitoring emails.

☞ **Note:** When you update two or more artifacts at a time, each user who is monitoring any of the changed artifacts gets a single email describing all the updates.

Choose how often to get emails from specific applications

How many email notifications do you want to get about the applications you are monitoring, and how often?

To further personalize your monitoring preferences, you can set the frequency of email notifications for each monitored application on a project level. If you don't set your application monitoring email frequency settings, your global setting applies.

For example, suppose you are contributing code to the "Widgets" project, but your role in the "Gizmos" project is of a more advisory nature. When you monitor an item in the "Widgets" project you'll want more detailed updates than you will want from items you've monitored in the "Gizmos" project.

1. Set the global default email frequency from your **MY SETTINGS** page.

   See *Choose monitoring email frequency* on page 103.
2. Click **MONITORING** from the **My Page** menu.
3. From the **Edit Monitoring Subscriptions and Preferences** pane, choose the project in which you want to configure monitoring email frequency.
4. On the **EMAIL NOTIFICATION PREFERENCES** tab, specify how often you want to be notified and click **Save**.

Any email notification preferences you set here will override the default preferences that you set on your **MY SETTINGS** page.

☞ **Note:** When you update two or more artifacts at a time, each user who is monitoring any of the changed artifacts gets a single email describing all the updates.

Monitor an item for someone else

To alert another user to an item, add that user to the list of people monitoring that item.

After a user is added to a monitored item, the user can configure their own monitoring preferences for the item, or stop monitoring the item.

1. Go to the item to which you want to add a user.
2. Click the item to view the artifact.
3. On the item's **View Artifact** page, click **Users Monitoring**.
4. In the **Users Monitoring This Item** window, click **Add**.
5. From the list of available project members, select the user or users that you want to add to the monitored item.

   ☞ **Tip:** Press and hold the **Ctrl** key to select more than one user.

6. Click **Add**.

   ☞ **Tip:** You can also click **Add All** to select all users.

7. Click **OK**.

The users are now added to the monitored item.

Lead a project team with CollabNet TeamForge

Managing a project means providing tools and processes to give each project member's contributions their greatest possible impact.

Create a TeamForge project

Create a new project when you have identified work to be done that has its own distinct character, dependencies or schedule.

What constitutes a project depends on your organization. Some organizations favor a small number of big, centralized projects. Others prefer a larger number of smaller, specialized projects. Your site administrator can help you decide if your work should be part of a larger project or a project of its own.

1. Access the **Projects** page through either of the following ways:

   - Go to **My Workspace** > **View All Projects**
   - Go to **My Page** > **Projects.**

2. Click **Create New Project**.

3. On the **Create Project** page, give the project a name and a brief description.

   - The name will appear in project lists and on the project's home page.
   - A terse description is recommended. There will be unlimited room to discuss the project's aims and methods in detail on the project pages themselves.

4. Provide a URL name for the project, if you want the URL for the project to be different from the internal project name.

   If you do not enter a URL name, the project URL will be the same as the project name.

5. If your site administrator has provided project templates, select the appropriate one for your new project.

   Project templates give you ready-made artifact types, work flow support, user roles and other start-up content appropriate to the kind of project you are creating.

   👉 **Tip:** The **DEFAULT PROJECT ACCESS** and **PROJECT ACCESS EDITABLE** options are displayed based on the project settings. If you would like to change these settings, ask your site administrator.

6. Click **Create**.

The project is submitted to the TeamForge site administrator for approval. You will receive an email notification when the site administrator approves or rejects your project. When your project is approved, you can get to it from your **MY PROJECTS** tab available under **PROJECTS** menu in the **My Workspace** page or from the **Projects** menu in your navigation bar.

Build your project web site

Project members and other users need to know how to interact with your project. You can give them the information and tools they need by creating and maintaining a project web site tailored to them.

You can build your project home page from scratch or assemble it from building blocks provided by TeamForge.

Assemble your project web site

For quick, flexible site building, use the ready-made web page components that come with your TeamForge project site.

There are components for showing text, wiki pages, charts and graphs, and other purposes. These components make it easy to put together a sophisticated project site in little time.

Create a project page

To provide information and functionality to people viewing your project, build one or more project pages.

1. Go to the page to which your new page will belong.

   👉 **Tip:** Any project page can have subpages belonging to it. A page that belongs directly to the project home page is called a "top-level page."

2. Click **Configure: On**.

3. Choose where your new page will fit in your project's structure.

- To create a page just under the project home page, click **Add Top-level Page**.

   A top-level page's title is always visible in the navigation tree at left.
- To create a page under the page you are on right now, click **Add Sub-page**.

   ☞   **Tip:** You can create a free-form page only as a subpage.

4. Give your new project page a title.

   Keep the title brief and descriptive.

5. Choose who can see this page.

   ☞   **Note:** Your choice will apply to all subpages that you create under this page.

   - To show this page to anyone with the necessary permissions, select **Visible**. For example, if you have defined a group of users who have access to your project, your new project page is visible only to those users. If your project is open to the public, anyone in the world can see it. Use this option when the information on this page is ready for a wide audience.
   - To show this page only to users with the project administrator role, select **Hidden**. Use this option if you are drafting content that you aren't ready to share yet, or want to share only with other project managers.

6. Select the **Freeform** check box to create a free-form page.

   ☞   **Note:** A free-form page's title is not visible in the navigation tree at left.

   ☞   **Tip:** You can not create free-form subpages when the top-level page is 'Hidden'.

7. Click **Save**.
8. Click **Configure: Off** .

Now you are ready to build the functionality into your project page with components such as text, news or tracker queries.

Create a project page component

Put information and resources in your users' hands with project page components.

For example, to let people know about important new developments, create a news component. To enable project members to find tracker items quickly, create a query component.

1. On the project page, click **Configure: On**.
2. Click **Add New Component**.
3. On the **Create Component** page, give your new component a title.

   Keep the title brief and descriptive.
4. Select one of the following component types that suits your need.

Text	Write free-form messages, reports or rants, in plain text or HTML.
Reports	Add various reports and charts to your project page.
Documents	Let project members exchange and review documents from the project page.
Wiki Page	Open up your project page to two-way communication.
Tracker Search Results	Make saved search results available from the project page.
Tracker Metrics	Add charts about tracker metrics to your project page.
Project News	Maintain a journal or blog about your project, share information and make announcements.

Project Statistics	Show visual measures of your project activities on the project page.
Subprojects	Add a list of subprojects to your project page.

5. Choose who can see this component.

- To show this component to anyone with the necessary permissions, select **Visible**.

  For example, if you have defined a group of users who have access to trackers in your project, a query component will be visible only to those users. If your project's trackers are open to anyone, all users who view this project page will see the query component. Use this option when you are sure the component is ready for general use.

- To keep this component under wraps until you are ready to show it, select **Hidden**.

  Now only users with the project administrator role can see this component. Use this option if you are drafting content that you aren't ready to share yet, or want to share only with other project managers.

6. Select one of the locations, Top of page or Bottom of page, where the component shows up on the project page.

7. Depending on the component type you selected, set the properties of the component.

Component type	In the Properties of this component section...
Text	1. Type your free-form messages, rants or announcements in the text box. 2. Click **Save**.
Reports	For more information about adding reports, refer to the topics: *Add reports to a project page* on page 108 and *Reporting in TeamForge* on page 151.
Documents	1. Select a folder from the list to display its contents on the project page. 2. Click **Save**.
Wiki Page	1. Type a title for the wiki page. 2. Click **Save**.
Tracker Search Results	Make sure one or more shared tracker searches are available to add to the project page. For more information about sharing saved tracker searches, see *Share a saved tracker search*. 1. Click **Add Saved Tracker Searches**. 2. Select one or more shared tracker searches from the **Select from Shared Tracker Searches** window. 3. Click **Add Selected**. 4. Select the number of rows of the search results to display from the **Display Rows** drop-down list. 5. Click **Save**.
Tracker Metrics	1. Select the number of charts from the drop-down list. You can add up to three tracker charts to your project page. 2. Select one of the chart types: Burndown, Open by Priority or Open vs Closed. 3. Select a data source for your chart, a tracker or a planning folder. 4. Click **Save**.
Subprojects	1. Select the number of subprojects to be displayed on the project page. 2. Click **Save**.

8. Click **Configure: Off**.

Add reports to a project page

Add one or more reports to your project home page to publish your project status to other project members. You can add only reports of type 'Public' to your project home page.

For more information about TeamForge reports, 'Public' and 'Private' reports, see *Reporting in TeamForge* on page 151.

1. On the project page, click **Configure: On**.

2. Click **Add new component**.
   The **Create Component** page appears.

3. Type a title for the report component.

4. Select Reports component type.
   A list of 'Public' reports for the project in context is shown.

5. Select **Visibility** and **Location**.

6. Select one or more 'Public' reports to add to the project page.

   ☞ **Note:**

   - You can add up to three reports per Reports component.
   - The 'Public' reports list may span over multiple pages if the number of reports exceeds **Page Size**. You can navigate through pages by clicking the page numbers at the bottom of the list.
   - However, you can change the **Page Size**:
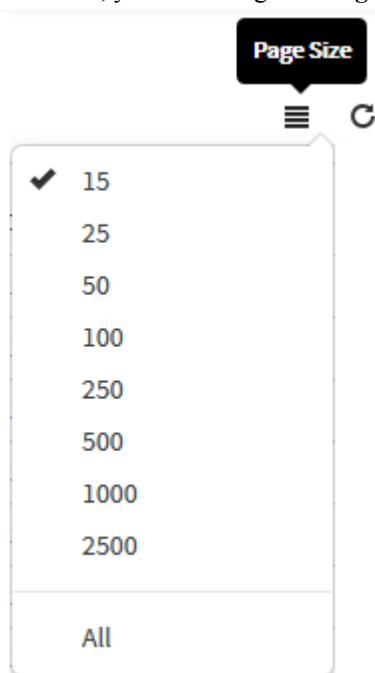
7. If you would like to add reports from other project(s) of which you are a member, select the required project from the **Select Public Reports** drop-down list.
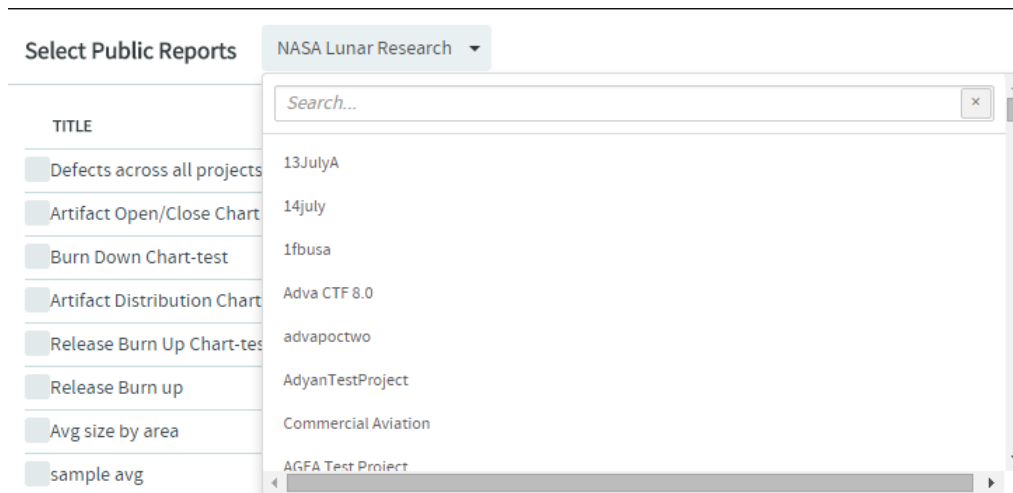
The list of reports existing in the selected project is displayed from which you can select the relevant reports.

☞ **Note:**

- Private reports and Table reports are not displayed in this list.
- You can add a maximum of three reports per Reports component.

8. Click **Save**.

The selected reports are added to the project page.

Reorder project page components

To make things easy for your project members, lay out your page components in a useful order.

1. On the project page, click **Configure: On**.
2. In the title bar of the project page component you want to move, click the up arrow ⬆ or the down arrow ⬇ .
3. Click **Configure: Off**.

Hide a project page component

While you are preparing a project page component, you may wish to keep it from being seen.

You can hide a page component from everyone except other project administrators.

1. On the project page, click **Configure: On**.
2. In the title bar of the project page component you want to hide, click the edit icon 🔧.
3. Select the **Hidden** option for **VISIBILITY**, and click **Save**.
4. Click **Configure: Off**.

Now no one who does not have the Project Admin role can see the component, even if they have access to the project and to this page. Users with the Project Admin role can see this component only when **Configure** is set to **On**.

Rename a project page component

As a page component's focus evolves, it's a good idea to rename it to match its changing function.

1. On the project page, click **Configure: On**.
2. In the title bar of the project page component you want to rename, click the edit button 🔧.
3. Update the title as appropriate to its current function, and click **Save**.
4. Click **Configure: Off**.

Edit a project page component

You may want to make some changes to your project page component to make it more useful.

1. On the project page, click **Configure: On**.
2. In the title bar of the project page component you want to edit, click the edit icon 🔧.
3. Update the title as appropriate to its current function, if required.

4. Select the **Hidden** option for **VISIBILITY** to hide the page while you work on it.

    You can hide a page component from everyone except other project administrators.

5. Change the following settings depending upon the project page component that you are editing:

    - For a **Sub-projects** component, you can change the number of projects to be displayed.
    - For a **Documents** component, you can change the folder location.

6. Click **Configure: Off**.

The project page component is displayed with modified content and/or settings.

Delete a project page component

When a component is no longer useful, remove it from the page to avoid distracting users.

1. On the project page, click **Configure: On**.

2. In the title bar of the project page component you want to delete, click the **Delete this Page** button.

3. Click **Configure: Off**.

Control access to a project page

Before you put information or functionality on your project page, make sure it is accessible to the people it is intended for.

☞   **Note:** This is only relevant if your page is not hidden. If your page is hidden, users who are not project administrators cannot see the page even if their role-based permissions would allow it.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Settings** page, click **Permissions**.

3. Click the role to which you want to give access to your project page.

    ☞   **Tip:** If the appropriate role does not exist, you must create it first.

4. On the **Edit Role** page, click **Project Pages**.

5. Under **Project Pages Permissions**, select the pages that users with this role can see and edit.

    - To enable users with this role to create, read and modify all project pages, select the **Project Pages Admin** permission.
    - To allow users with this role to see pages but not edit them, select the appropriate page in the **View** section.
    - To allow users with this role to modify the contents of a text component project pages, select the appropriate pages in the **Edit Text Content** section.

    ☞   **Note:** The project home page is always visible to any user who is authorized to see the project.

Reorder project pages

Facilitate your users' experience in your project by putting your project pages in an order that matches their needs.

1. Click **Configure: On**.

2. Select a project page and click **Edit Structure**.

3. Rearrange the structure of this page in any of these ways:

    - Select a page, and use **Cut** and **Paste** to place it in a different location.

      ☞   **Note:** When you move a page, all of its sub-pages will also be moved.

    - Drag and drop a page to a different position.
    - Click **Add New Page** to add a sub-page to the current page.
    - Select one or more pages and click **Delete** to remove them.

4. Click **Save Changes**.

5. Click **Configure: Off**.

Hide a project page

While you are preparing a project page, you may wish to keep it from being seen.

You can hide a page from everyone except other project administrators.

1. Click **Configure: On**.
2. Click **Edit Page Properties**.
3. For **Visibility**, select **Hidden** and click **Save**.

Now no one who does not have the Project Admin role can see the page, even if they have access to the project and to this page's parent page.

When you are finished building your page, don't forget to switch the page to "Visible" so that the intended users can see it.

Rename a project page

As a page's focus evolves, it's a good idea to rename it to match its changing function.

1. Click **Configure: On**.
2. Click **Edit Page Properties**.
3. Change the **Title** as appropriate to its current function, and click **Save**.
4. Click **Configure: Off**.

Hand-code your project web site

To fully custom control your project web site pages, build your own HTML and check it into the project's Publishing repository.

You can link to these pages as you would to any normal HTML page.

Create a custom project home page

If you decide to hand-code your project home page, the page works like any HTML page.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** page, select **Show custom web page** from **PROJECT HOME OPTIONS**.

   (**Show default project pages and components** is selected by default.)
3. Add the `index.html` file to the `publishing/www` Subversion repository.
   The content of your `index.html` shows up immediately.

   ☞ **Note:** When you switch to a hand-coded home page, you can still access any project pages you have created.
4. Click **Save**.
   Your custom **Project Home** page is now active.

   ☞ **Note:** If your project has any subprojects, the subprojects are also listed. To see them, make sure your `index.html` file exists and is in the right repository location.

See what's in your publishing repository

You can view the files in your TeamForge site's `publishing` repository through relative URLs.

For example:

1. Enter the URL: `https://forge.collab.net/sf/projects/sampleproject/index.html` in the address bar to display contents of the `index.html` file.

   In this case, your `index.html` file must be in the `publishing` repository under the `www/sampleproject` path.

   The contents of the `index.html` file are displayed in the **Project Home** page.
2. Enter the URL: `https://forge.collab.net/sf/projects/sampleproject/roles/roledetails.html` in the address bar to display contents of the `roledetails.html` file.

   In this case, your `roledetails.html` file must be in the `publishing` repository under the `www/sampleproject/roles` path.

   The contents of the `roledetails.html` file are displayed in the **Project Home** page.

Set up trackers

A tracker is a collection of related artifacts that describe work to be done or issues to be resolved. Every project should have one or more trackers.

When you start a tracker, you decide which fields will be used, who will use them, and how they will use them.

Create a tracker

Create a tracker whenever you need to report and track bugs, feature requests, support requests, or any other type of issue where ownership, status, and activity must be managed.

Individual tracker entries are referred to as tracker artifacts. A tracker is a set of tracker items with a common purpose, such as bug reports, feature requests, or tasks.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. Click **Tracker Settings**.

3. Click **Create**.

4. On the **Create Tracker** page, provide a name and description for the tracker.

   ☞ **Tip:** Descriptions help users learn how best to provide the information you want from them. To maximize your chances of getting useful data, make your description as informative as you can.

5. Select an icon that suggests the type of work the tracker is handling.

   This icon will appear with any artifact in this tracker, wherever it is viewed on the site. For example, if someone brings an artifact from this tracker into a planning folder, users of the planning folder can glance at the artifact's icon to see where it comes from.

6. Select the relevant unit from the **DISPLAY EFFORT IN** field. The units displayed here are configured based on the size of the artifacts in the tracker. Eg. Select the unit as **HOURS** for a tracker of small defects, **DAYS** for a tracker of tasks, and **WEEKS** for a tracker of epics.

   ☞ **Note:** Configure the units at the project level and not at the planning folder level.

7. Select **INCLUDE FOREIGN CHILDREN** to include points and efforts from children artifacts across the projects in TeamForge.

   ☞ **Note:** In a parent artifact, enabling **CALCULATE POINTS** field sums and rolls up the points from all its children artifacts within the project. In this total, if you want to include children artifacts from other projects across TeamForge, have the **INCLUDE FOREIGN CHILDREN** option enabled.

8. Click **Create**.
   The new tracker appears at the bottom of your list of trackers.

9. If necessary, drag the tracker to a place in your tracker list that makes sense.

   The order you set here controls the order of every tracker list in your project.

10. You'll probably need some custom fields to capture information that's specific to your project.

    See *Create your own tracker fields* on page 38.

11. To speed up the team's work, you may want to set up some rules for automatically reassigning artifacts when their contents change.

    See *Create a tracker work flow* on page 40.

Create your own tracker fields

To track data that is not captured by the default set of fields, create new fields that fit your project's purposes.

You can create the following user-defined fields in each tracker:

• Up to 30 text entry fields.
• Up to 30 date fields.
• Up to 30 single-select fields.
• An unlimited number of multiple-select fields.

   ⚠ **Caution:** Creating a large number of user fields and multiple-select fields may affect site's performance.

Create a text field

To let users type in data, create a text entry field.

A tracker can have up to 30 text fields.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. Click the tracker to which you want to add a text field.
4. On the **TRACKER FIELDS** tab, click **Add Field**.
5. On the **Create Field** page, provide a name for the field.
6. Configure the shape of the field with the **Field Width** and **Field Height** fields.
7. To help users enter the right text values, select **Use Text Validation** and supply a regular expression that describes the appropriate values.

   This can help reduce errors and keep your team's data as meaningful as it can be. For more detailed instructions, see *Validate text entries in a tracker artifact* on page 119.
8. Click **Save Field**.
   The new field is created.

Create a "Select" field

To let users choose values from a list that you define, create a "Select" field.

You can create up to 30 single-select fields and an unlimited number of multiple-select fields in a tracker.

☞ **Note:** Creating a large number of multiple-select and user fields may affect TeamForge's performance.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the tracker to which you want to add a "Select" field.
4. On the **TRACKER FIELDS** tab, click **Add Field**.
5. On the **Create Field** page, provide a name for the field.
6. Use the **Input Type** menu to specify whether users will be able to select one value or more than one.

   If you're going to make this a required field, pick one of the values to be the default value. This value is applied to existing artifacts and artifacts that are moved from another tracker.
7. Decide whether users *must* choose a value.

   • Required fields automatically appear on the **Submit Artifact** page.

   ☞ **Note:** If you make the field required, you must specify a default value.

   • If you make a **User** field required, specify one or more default users.
   • If you make a **Date** field required, the default is "today."

   • For optional fields, select **DISPLAY ON SUBMIT** if you want the field to appear when a user first creates an artifact.
   • To prevent the field from being used at all, select **DISABLED**. (By default, new fields are enabled.)
8. Use the **Values** section of the **Create Field** page to add more values for the user to choose from.
9. Keep adding values until you have the list of options you want, then click **Save Field**.

Create a people-picker field

To let users choose other users from a list, create a "people picker" field.

The **Assigned to** field is a people-picker field that is present in every tracker. An artifact can be assigned to only one user at a time. You can give yourself more flexibility by adding any number of custom people-picker fields.

For example, your QA team may want to assign a person to track progress on an artifact while it is assigned to a developer. You might create a people-picker field called "QA monitor" to specify who that person should be.

In people-picker fields you create, users can select multiple users.

☞ **Note:** Creating a huge number of user fields can slow down the site's performance.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the tracker to which you want to add a people-picker field.
4. On the **TRACKER FIELDS** tab, click **Add Field**.
5. On the **Create Field** page, provide a name for the field.
6. On the **INPUT TYPE** menu, select Select User(s).
7. In the **DEFAULT FILTER** field, choose whether the list of people available in your new field will include members of this project or everyone who is registered on the site.
8. Configure the size of the field with the **FIELD WIDTH** field.
9. Click **Save Field**.
   The new field is created.

Organize tracker fields

Most tracker artifacts ask the user for a lot of information. You can arrange the input fields in columns and rows to make it easier for users to find the fields they need.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of current trackers, click the tracker whose fields you want to organize.

   Click the **TRACKER FIELDS** tab if it isn't already showing.
4. If some fields seem to be logically connected to each other, create a section to bring them together.
   a) Click **Add Separator** and select **Section**.

      Give the section a short but descriptive label.
   b) In the list of fields, drag your new "Section Separator" row to a position that makes sense.
   c) Drag the appropriate fields under the Section Separator that you just created.
5. Within a section, arrange fields logically into columns.
   a) Click **Add Separator** and select **Column**.

      Give the column a short but descriptive label.
   b) In the list of fields, drag your new "Column Separator" to a position that makes sense.
   c) Drag the appropriate fields under the Column Separator that you just created.
   d) Create as many columns as you need.

      Drag a column separator above another column separator to move it to the left in the artifact entry form. Drag it below to move it to the right.
6. Within a column, group fields into rows if appropriate.
   a) Click **Add Separator** and select **Row**.

      Give the row a short but descriptive label.
   b) In the list of fields, drag your new "Row Separator" to a position that makes sense, then drag the appropriate fields under the Row Separator.

      ☞ **Note:** You can have rows and columns without sections.

Enable or disable fields

If a tracker field is disabled, it does not appear on the **Artifact** page. Most fields can be disabled.

Disabled fields are accessible only to tracker administrators on the **TRACKER FIELDS** tab.

You can enable or disable any field that is user-defined or configurable. However, you can't disable all configurable fields. For example, the **Status**, **Priority**, **Category,** and **Planning Folder** fields can't be disabled.

☞ **Note:** If your goal is to prevent users from entering data into a field when submitting an artifact, but still display the field on the **Edit Artifact** and **Tracker Search** pages, don't disable the field. Instead, clear the **DISPLAY ON SUBMIT** option on the **Edit Tracker Field** page.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of current trackers, click the tracker you want to configure.
4. On the **TRACKER FIELDS** tab, select the fields you want to enable or disable.

    •   Click **Disable** to remove them from the **Artifact** page.
    •   Click **Enable** to allow them to be configured and displayed.

    ☞ **Note:** Data in disabled fields is still search-able, but disabled fields do not appear as inputs on the **Search** pages.

Require users to complete a tracker artifact field

If a field is set as required, users cannot submit artifacts without completing it. Most tracker fields can be required or optional.

Each tracker can have its own required and optional fields. Required fields are marked with a red asterisk ⋆ on the **Submit Artifact** page.

☞ **Note:** When you make a field required, any field whose values depend on that field's values is also required. See *Help users select options in a tracker artifact* on page 118 for more about dependent field values.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of current trackers, click the one you want to configure.
4. On the **Tracker Fields** tab, click the name of the field you want to set as required or optional.

    By default, only the **Title**, **Description**, , and **Status** fields are set as required.
5. On the **Edit Field** page, select or clear the **Required** option to make a field required or optional.

    Required fields automatically show up on the **Submit Artifact** page.

    ☞ **Note:** System-defined fields and the **Status** field are always required.

6. For optional fields, select or clear the **DISPLAY ON SUBMIT** option.

    This specifies whether the field will appear on the **Submit Artifact** page.
7. Click **Save Field**.

Configure tracker "Select" field values

To help users provide meaningful information, supply them with useful field values to choose from in the input fields in the tracker entry form.

☞ **Tip:** Once a tracker has been created, you may:

    •   Create one or more user-defined single-select or multiple-select fields.
    •   Add predefined values to the fields. Remove values, if required.
    •   Enable or disable fields.
    •   Change the default values for fields.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, click the name of the tracker that you want to configure.
4. On the **TRACKER FIELDS** tab, click the name of the field whose values you want to edit.
5. On the **Edit Field** page, set up the field values you want users to see when they create a tracker artifact.

    •   To define a new value, click **Add**.
    •   To rename a value, edit the existing text. If you rename a value, the value is renamed in all existing artifacts.

- To remove a value, check the box and click **Delete**. If you delete a value, the value is changed to **None** in all existing artifacts.
- Select **DEFAULT VALUE** to set which option will be chosen if the user makes no selection. When you move a tracker artifact from one tracker to another, the default field value is the value that comes along.

  ☞ **Note:**
  - When you edit the values of the **Status** field, you are also asked to describe what each value's status means, as shown in the **Values** section of the **Edit Field** page. This status meaning is used in **Advanced Search** to define which values are returned when searching for open or closed artifacts.
  - As always, when you create a new tracker, the default value for the '**Priority**' field is set as '4 - Low'. However, you can change the default value by editing this configurable single select field, '**Priority**'. You cannot delete or disable the **Priority** tracker field.
  - When you change the tracker fields, the values in the existing artifacts remain unchanged.

6. Click **Move Up** or **Move Down** to order the list the way you want it.

7. Click **Save Field**.

All values are now available in the selection menu for the field.

Configure tracker units

You can estimate the value of efforts meaningfully in the form of units using the TeamForge tracker.

You can set any unit as the default for a tracker or planning folder and create any number of units. The default base unit is 'Hours' which you can rename, but not delete. You can also toggle between the burndown chart and effort values for any unit.

☞ **Note:** You cannot enter decimal values in the **Conversion** field. Deleting a unit will cause all the associated artifacts to express effort in the form of the base unit.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. Click **Tracker Settings**.

3. On the **UNITS** tab, click **Add**.

4. On the **Add Unit** page, enter the **UNIT NAME** and **CONVERSION** value for the unit.

5. Click **Save**.

Assign a tracker artifact automatically

You can configure the tracker to automatically assign newly submitted artifacts to specific project members.

You can assign artifacts to individuals based on the values in the **Category** or **Release** field.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. Click **Tracker Settings**.

3. From the list of existing trackers, click the tracker you want to configure.

4. On the **AUTO ASSIGNMENT** tab, select **Auto Assign By Category** or **Auto Assign By Release**.

5. For each category or release value, choose a project member from the **AUTOMATICALLY ASSIGN TO** drop-down menu.

   This menu contains all project members with the tracker edit permission. Click the Search icon to display a list of project members to whom you can auto-assign artifacts. Choose **None** if you do not want artifacts with a specific value automatically assigned.

6. Click **Save**.

Whenever a new artifact is submitted with a value in the relevant field, the assignee receives an email notification and the artifact appears on the asignee's **My page**.

Create a tracker work flow

To channel project members' work on tracker items, set up rules for how a tracker item can move forward.

Before creating a tracker work flow, see that these criteria are met:

- You have a tracker to work with.

- The tracker has a set of statuses defined, such as "In progress" and "Ready for QA."
- Roles exist, and you can assign project members to them.

A work flow is a sequence of changes from one status to another. You can define status transitions for any combination of tracker statuses in the tracker.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings**.
3. From the list of existing trackers, select a tracker.
4. Click the **WORKFLOW** tab.
   The **Workflow** page lists all of your status values and the tracker workflow that you have configured.
5. On the **Workflow** page, click the status value for which you want to create a workflow.
6. On the **Edit Field Transition** page, select a status value from the **Create Transition to Status** drop-down menu.

   ☞   **Note:** By default, the **Any** workflow is shown. This means that you are defining workflow rules for the transition from the status that you selected to *any* other status.

7. Click **Add**.
   A new work flow is added. The **Any** work flow is changed to **Remaining Statuses**.
8. In the **ROLES TO MAKE THIS TRANSITION** section, specify which users can make this change.
   For example, only users with the QA Engineer role are allowed to change artifacts from **Open** to **Cannot Reproduce**.
9. In the **REQUIREDFIELD(S) FOR STATUSTRANSITION** section, select the fields for which the user *must* provide values.
   For example, the user must assign the tracker item to someone and enter a comment.

   ☞   **Note:**

       - Fields that are already required do not appear in this list.
       - Fields whose values depend on a required parent field are automatically required. (See *Help users select options in a tracker artifact* on page 118 for more on parent and child fields.)

10. Click **Save**.

The work flow is now saved. When a user submits or edits the status of a tracker artifact, he or she sees only the options that are allowed by the work flow.

Set default tracker columns

Choose what information is visible by default to users of your project when they open a tracker or planning folder.

Users can also set their own personal default column configurations, which override the project defaults.

1. Click **TRACKERS** from the **Project Home** menu.
2. Select a tracker, planning folder or team and click **COLUMNS** > **Configure.**

   - If you've already saved a column configuration, click it and skip the rest of these steps.
   - To go back to the default column configuration, click **System (default)** and skip the rest of these steps.
   - To set up a new configuration, click **Configure.**

3. Choose your columns.
   a) Move the columns you want from **Available Columns** to **Selected Columns**.

      Artifact ID : Title, **Priority** and **Status** are required columns.

      ☞   **Note:** Selecting more columns can increase the time required to load the listing page.

   b) Remove any columns you don't need from **Selected Columns**.
   c) Use the move up and move down arrows to change the display order of the columns.
4. Apply your choices to your view of the tracker.

   - To use this arrangement this time only, click **Apply**. The next time you log in, you'll start with the default view again.

- To save your column layout for repeated use, click **Apply and Save**, then give your arrangement a name. The next time you log in, you'll see the column arrangement you just selected. (If you've sorted the records in your view, that sort order is saved too.)

  ☞ **Tip:** If you are editing a column configuration that already exists, you can rename it by saving it under a new name.

5. To make the same set of columns appear every time you come to this tracker, planning folder or team, click **COLUMNS** > **Save** and from **Save Column Configuration** page, select **Make this my default view**.

6. To make the same set of columns appear for every user the first time they see any tracker, planning folder or team in the project, click **COLUMNS** > **Save** and from **Save Column Configuration** page, select **Make this the default view for all project members**.

The project default configuration you set is now the default configuration for all project members, unless they have created their own personal default column configuration.

Change a tracker

A tracker's real-world uses often outgrow the name or description you gave it when you created it. When that happens, it's a good idea to update the tracker to reflect its changing role.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. Click **Tracker Settings**.

3. From the list of existing trackers, click the tracker you want to edit, and click **EDIT**.

4. On the **Edit Tracker** page, provide a new name or description for the tracker, and update the icon.

5. Update the units from **DISPLAY EFFORT IN** and click **Save**.

   ☞ **Note:** These units are configured in the **Units** page at the project level, and not at the planning folder level.

6. Select **INCLUDE FOREIGN CHILDREN** to include points and efforts from children artifacts across the projects in TeamForge.

   ☞ **Note:** In a parent artifact, enabling **CALCULATE POINTS** field sums and rolls up the points from all its children artifacts within the project. In this total, if you want to include children artifacts from other projects across TeamForge, have the **INCLUDE FOREIGN CHILDREN** option enabled.

7. If necessary, drag the tracker to a place in your tracker list that makes sense.

   The order you set here controls the order of every tracker list in your project.

Help users select options in a tracker artifact

You can help users cope with complex information by guiding them to eligible values in single-select fields.

Any tracker that manages real-world information will quickly become very complex. Users can be confused by a proliferation of "Select" fields. Confusion can lead to inconsistent data, which makes your job harder.

You can help relieve the complexity by showing users their eligible options in a given field based on values they have already selected in another field. You can create overlapping sequences of dependent fields, with as many levels as you need.

This simplifies things for the user, but for the tracker administrator it can quickly get complicated. So let's look at an example.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. Click **Tracker Settings** and create a tracker.

   For this example, let's call it the **Lunch Planning** tracker.

3. Create a single-select field and call it **Lunch type**.

   (For this example, we'll ignore the built-in fields, such as **Status** and **Assigned to**. We're just working with fields that you create.)

4. Create some values for the **Lunch type** field. Let's call them Buffet, Picnic, and Banquet.

Each type of lunch will make sense in some kinds of locations and not others. For example, you would not normally plan a banquet lunch in a park. We are now going to make it easy for users to avoid making such a mistake.

5. Create a single-select field called **Location type**, with Lunch type as the parent field, and give it some plausible values.

   a) Start by adding an option called Beach. In the **Parent Values** column, choose Picnic, because that's the kind of lunch you would have at the beach.

      (The **Parent Values** column lists all the values in the parent field you selected.)

   b) Add a Restaurant. In the **Parent Values** column, hold down the **Ctrl** key and choose Banquet and Buffet, because either of those could be held at a restaurant.

6. Create a single-select field called **Location**, with Location type as the parent field, and give it some values to choose from.

   a) Start with an option called Happy Food Restaurant. In the **Parent Values** column, select Restaurant, because that's the type of location that Happy Food Restaurant is.

   b) Add another option, Hanalei Cove. Under **Parent Values**, select Beach.

7. Save your work and go to the tracker whose settings you have been editing.

   Try selecting from the interdependent values you have just created.

Observe that the value you choose in the **Lunch Type** field controls which values are available to you in the **Location Type** field, and that selecting a value in **Location type** in turn controls the values that appear in the **Location** field.

• When a user selects Banquet for a lunch type, they can select Restaurant but not Beach in the **Location type** field. You will have less error correction to do, and users will avoid confusion.

• When a user selects Picnic for their lunch type, the **Location Type** field offers only Park and Beach. Now you will not have to go through and clean up after users who mistakenly choose to plan a picnic at Happy Food Restaurant, and the doorman at Happy Food Restaurant will not have to turn away users who mistakenly show up with picnic baskets.

☞ **Important:** If you are used to defining your own Tracker fields, the ability to make field values depend on other values may change how your trackers work in ways you didn't anticipate. Keep these points in mind:

  • Linking fields in this way doesn't modify existing data, but when users later modify fields that are linked, they will have to adhere to the relationships you set here.

  • If a field has a parent, and that parent field also has a parent, the top-most parent field must have at least one value.

  • When a field has a parent field that is required, the child field's default value is set to None. If that required parent field is deselected, the child field no longer has to be required, but Required remains the default.

  • If you require a specific field value before an artifact can be placed in a given status, that field's children are subject to the same requirements. See *Create a tracker work flow* on page 40 for more about controlling what status an artifact can be in.

  • If you delete a field that contains values that another field's values depend on, the dependent field becomes a standard single-select field on its own.

  • When you cut and paste an artifact from one tracker to another, only those field values that also exist in the new tracker come along with the artifact. If those values aren't valid under the dependency rules of the new tracker, they are still brought along.

Validate text entries in a tracker artifact

You can help users contribute useful information by testing their text entries against rules you configure.

Text fields can be error-prone because they invite free-form input. You can help users provide usable information by automatically rejecting values that don't match the needs of the tracker.

This simplifies things for the user, but for the tracker administrator it can be complicated. So let's look at an example.

☞ **Note:** If your goal is to require users to enter *some* value, whatever the value is, don't use text field validation. Select the **Required** option instead.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tracker Settings** and create a tracker.

    For this example, let's call it the **Bugs** tracker. It will be used to record entomological specimens in a collection.
3. Create a text field and call it **Legs**.

    This is for users to record the number of legs each specimen displays.
4. Select **USE FIELD VALIDATION** and supply a validation rule that requires the user to enter a number.

    For example, if a given insect has six legs, you'll want the user to enter the numeral **6**, and not a string such as "six" or "several."

    Try this regular expression:

    ```
    \d{1,3}
    ```

    This rule requires the user to enter a number with one, two or three digits. Now, a user who means to record a centipede with 100 legs but enters 1000 by mistake will not be able to save the artifact until the error is corrected.
5. Enter a sample string to test your regular expression.
    Any part of the sample string that matches your regular expression appears under **Match Results**. If nothing appears, rework your regular expression until you get a match.
6. Create another text field and call it **Location**.

    This is where users will record the geographical spot where they collected the bug.
7. Select **Use Text Validation** and supply a validation rule that requires the user to enter a pair of geographical coordinates.

    For example, if a given insect was found outside CollabNet's California headquarters, you'll want the user to enter a string like `37.674689,-122.384652`, and not something like "Brisbane" or "Out on the lawn."

    Try this regular expression:

    ```
    [-]?[0-9]*[.]{0,1}[0-9]{0,4}
    ```

    This rule requires the user to enter two numbers, separated by a comma, in the general format of a pair of mapping coordinates.

    > 👉 **Note:** This particular regular expression does not guarantee that the coordinates are valid, just that they look like coordinates.
8. Save your work. In the tracker whose settings you have been editing, try entering a number greater than 999 in **Legs**, or a street address in **Location**
    The red **X** next to the field indicates that the text entry is incorrect. A green check indicates that the value meets the requirements.

Notice that any field in which you are validating text entries is identified by **Text Entry (with Field Validation)** when listed on the **TRACKER FIELDS** tab.

Manage a product with CollabNet TeamForge
A product owner can use TeamForge to help define the scope of a product and contribute to its planning and implementation.
Define the scope of your project
Defining scope is an iterative, interactive process. As you go through it, you'll find elements of your scope expanding, shrinking or changing shape in response to feedback from analyzing and planning out the work.

> 👉 **Note:** It's a good idea to get the feature definition process under way *before* setting up planning folders (see *Create a planning folder* on page 17), because defining features gives you the raw material for your planning process.

1. Before you get started, you'll do some kind of user research, even if it's only a few phone calls, to get an idea of the needs of the customers you hope to satisfy. Express these needs and desires as stories about what a user can do with your product. Then create an artifact for each user story you identify.

    A user story describes the situation after your product has been launched: What can the user do now that they couldn't do before?

> 👉 **Tip:** For best results, your user story artifact should by limited to a clear description of the capability you want in the product. Remember that a user story is not an implementation plan. Details about the implementation will be recorded in the user story artifact, but when you write a user story, try to leave the implementation up to the developer who will be responsible for it.

2. You may want to define one or more fictional users who resemble the real-life users of your product. This can help simplify and focus your thinking.

3. Use the Priority field to express your opinion of how important the story is to the user. In general, the most eagerly desired capabilities will be addressed first. (During implementation planning, your Priority setting will be used as one input in summing up the effort involved in each priority level.)

4. Make it as clear as you can at the outset what degree of functionality is acceptable. For example, if your team is creating an airplane, how high must it be able to fly? How far must it go before refueling? How many passengers must it carry? Stating your acceptance criteria concretely helps reduce the time needed for ongoing reviews and changes.

Monitor project output

Regularly test and measure the features your team produces, involving real users as much as possible.

Accurately representing the needs and responses of real-life users is a key part of the product owner's role.

1. Schedule acceptance meetings throughout the iteration, either at regular intervals or whenever a given user story is completed.

   Use the latest working build of the product, and exercise the functionality live.

2. Test that the functionality implemented matches what is specified in the user story.

   Also review the user interface and user documentation associated with the user story, if any.

3. Enter any issues in the artifact's **Comments** field.

4. Update the user story artifact to reflect the outcome.

   - If a user story is accepted, change the artifact status to "Accepted" or the equivalent.
   - If a user story is not accepted, change the artifact status back to "In Development," or the equivalent, for further work. You can review it again in another acceptance meeting.

> 👉 **Note:** Sometimes a feature turns out not to meet the user's needs even though it has been implemented as specified. This is normal. It is not uncommon for the user's needs to change after the user story has been captured, or for the user research process to miss one or more important details. Just note the discrepancy in the user story and route the artifact back into the appropriate part of the process.

Organize your project with planning folders

Create and populate all the planning folders you need to capture the work you are planning.

A planning folder is a way to organize work into feasible chunks and monitor its progress. When you've thought about the general categories the work falls into, you can to set up planning folders that reflect those ideas.

A planning folder can represent:

- A set of tasks, such as Iteration 3, or "Initial infrastructure development."
- A period of time, such as "April," or "Q2-2010."
- A phase of development, such as "Testing" or "Deployment."
- A component of the product, such as "Chapter 12" or "Rear stabilizer."

> 👉 **Tip:** It often makes sense to set up planning folders *after* you have outlined and analyzed the features you plan to deliver. See

When you've set up your planning folders, you have four views available to work with them:

| LIST | PLAN | TRACK | KANBAN |

- **LIST**: The list view.
- **PLAN**: The planning board view.
- **TRACK**: The task board view.

- **KANABN**: The kanban board view.

For more information, see the related concept: *What is a planning folder?* on page 958

Create a planning folder

A planning folder is a way to organize work into feasible chunks and monitor its progress. As a project admin or as a user with the appropriate permissions, create and populate all the planning folders you need to capture the work you are planning.

When you've thought through your plan, express it in one or more planning folders.

☞   **Tip:** It often makes sense to set up planning folders *after* you have outlined and analyzed the features you plan to deliver. See *Define the scope of your project* on page 120.

For your agile projects, you have the option to create planning folders specific to iterations and releases, or you can create generic planning folders which you may customize later. When you've thought about the general categories the work falls into, you are ready to create planning folders that reflect those ideas.

A planning folder can represent:

- A set of tasks, such as Iteration 3, or "Initial infrastructure development."
- A period of time, such as "April," or "Q2-2010."
- A phase of development, such as "Testing" or "Deployment."
- A component of the product, such as "Chapter 12" or "Rear stabilizer."

1. Click **TRACKERS** from the **Project Home** menu.
2. Under **PLANNING FOLDERS**, click the folder in which you want to create a new planning folder. Click **NEW**. An appropriate planning folder creation page is displayed.
3. Select the type of planning folder you want to create: **Release**, **Iteration** or **Folder.**

   While "Release" and "Iteration" are specific planning folder types, "Folder" is your standard, generic planning folder. Though having multiple planning folder types help you logically organize your planning folders, there's no hierarchical restrictions on these different planning folder types. For example, as a user, you have the liberty to create a "Release" planning folder as a child of an "Iteration" planning folder, which may not be the ideal use case in an Agile project.
4. Enter a brief and descriptive name for your planning folder. For example:

- In an agile project, depending on your requirement, you can create two or more iteration folders called "Iteration 1", "Iteration 2" and so on within the Release planning folder.
- In a phased, waterfall-style project, you might name your first planning folder "Design," the next "Build," and so on.

   ☞ **Tip:** Don't worry if you don't have anything of interest to put into your planning folders yet. The parallel process of filling out the feature tree will provide plenty of material for this.

   ☞ **Note:** This field is mandatory.

5. Use the **DESCRIPTION** to briefly signal the kind of work that will be contained in this planning folder.

   Include enough information to help people get up to speed quickly when they join your project. But save most of the detail for the individual tracker artifacts, where project members will spend most of their time.

6. For **STATUS**, (available since TeamForge 5.4), select a value that communicates where the planning folder is in its life cycle.

   For a new planning folder, you'll probably want to select Not started.

   ☞ **Tip:** You can create more values to choose from in your Project Settings.

7. If you have set up a file release in the File Releases tool to deliver the work you are tracking here, you can identify it in the **FILE RELEASE** field (available since TeamForge 5.4).

   Any artifact you add to this planning folder will also appear in the Planned Tracker Artifacts tab when you look at that file release.

   ☞ **Note:** If you are creating an Iteration folder, this field displays the default file release value set at its parent folder level ("Release" folder) which you can modify.

8. For **CAPACITY**, (available since TeamForge 5.4), provide a number that expresses how much work your team thinks it can do in the period represented by this planning folder. This field is available only for "Iteration" and "Folder" types.

   ☞ **Tip:** When you create a new planning folder, it can be a good idea to set its capacity to zero until you get a feel for the artifacts and resources you are working with.

9. For **POINTS CAPACITY**, (available since TeamForge 7.1), provide a number that represents the amount of work (expressed as number of story points) that you think can be handled in that specific planning folder. This field is available only for "Release" and "Folder" types.

   ☞ **Note:** To assist project managers gauge the planning folder's points capacity, when a release or standard planning folder is selected in one of the *planning board* swim lanes, this information is displayed as a ratio of planned number of story points versus the actual number of story points. This information does not apply to Iteration folders and therefore does not appear on the planning board swimlanes when you select an Iteration planning folder. For more information, see *Use the Planning Board* on page 131.

10. Select or enter a start and end date for the work that this planning folder will contain.

   You can change these dates at any time. However, it's important to specify dates even if they're not firm, because useful visual aids and reports depend on them.

   ☞ **Note:**
   - The start and end dates of the child planning folder should be within that of its parent folder.
   - These dates are mandatory for an "Iteration" planning folder because the burndown chart for Iteration planning folders is enabled only when you specify the start and end dates of an Iteration planning folder. Also, these dates are not mandatory for a Release planning folder because the burndown chart of a Release planning folder takes into account the start and end dates of its child planning folders (Iterations) and not its own.

11. Select the unit that is suitable for the planning folder from the **DISPLAY EFFORT IN** field. Eg. if the planning folder represents a sprint, then select the unit as **Hours** and if it represents a collection of sprints, then select the unit as **Days**.

☞   **Note:** Units are configured at the project level and not at the planning folder level.

12. Select either **Effort** or **Points** from the **BURNDOWN VIEW BY** drop-down list to set the View Mode in the burndown charts (in the planning folder **List Artifacts** page). This field is available only for "Folder" type.

Depending on the planning folder type, the burndown chart is displayed.

- Burndown chart for the standard planning folder type which includes the View Mode:

- Burndown chart for "Iteration" planning folder type:

- Iteration-wise burndown chart for a "Release" planning folder type:

13. Click **Create**. The planning folder is created.

Update a planning folder

A planning folder is a dynamic representation of a changing situation. From time to time, you'll want to update its name, description and other parameters to reflect changes in the underlying work.

1. Click **TRACKERS** from the **Project Home** menu.

2. In the planning folder tree, click the planning folder you want to update.

3. Use the **EDIT** menu to make your changes.

Manage statuses for a planning folder

To help team members understand how to work with a planning folder, create statuses for it that correspond to the planning folder's life cycle.

When you have created a planning folder status, you can apply it to any planning folder in your project.

For example, at a given moment you might have one planning folder in "Under development" status, while another is in "Finished" status and another is in "Preliminary scoping" status.

☞ **Tip:** You may also want to move the planning folder to a relative position in the planning folder list that communicates where it stands in the order of work.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. Click **Planning Folder Settings**.

3. Under **Statuses**, click **Add**.

4. Configure your new planning folder status.

　　a) Give the status an unambiguous name that others will find easy to understand.

　　b) Specify whether the status counts as active or inactive.

　　　　Project members may want to exclude inactive planning folders from their view, to avoid overload.

　　c) Select which value is the default value for new planning folders and planning folders that originated in earlier TeamForge versions.

　　d) Arrange your planning folder statuses in the order that makes sense for you.

Reorder planning folders

When the sequence of work in your project changes, it's a good idea to reorder the planning folders so that they accurately reflect the real order of work.

For example, it's not unusual for product owners, who must constantly weigh the changing business value of features, to ask for one set of user stories to be put aside so that another set can be delivered first. When this happens, a project manager may want to move the corresponding planning folders so that their relative positions visually confirm their relationship.

☞ **Tip:** You may also want to set the planning folder's status to a value that communicates the change in the order of work.

By default, planning folders are ordered according to when they were created. New planning folders are added to the end of the folder tree.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the folder tree, click the folder containing the subfolders that you want to reorder.
3. Choose **Reorder Subfolders** from the **EDIT** menu.
4. Organize the subfolders:

   - To sort them alphabetically, click Alphabetize icon.
   - To reorder a specific folder, select it by clicking the title, then click move up or move down arrow until the folder is where you want it.

Reorder the contents of a planning folder
You can move artifacts in a planning folder up and down to reflect the order in which they should be addressed.

A set of user stories often includes dependency relationships that require that one user story be addressed before another. For example, suppose a backend database modification must be in place before a user interface component can be created to access it. Both user stories may have the same degree of priority, but one has to be done before the other.

One way to manage this is by ranking the user stories in your planning view so that they reflect the sequence you want. When you do this, team members have a better chance of selecting work efficiently and reducing the need for rework.

☞ **Remember:** Rank is not the same as priority. Two artifacts may have the same priority but different ranks, or sequence positions.

1. If you are not an administrator on this project, ask your project administrator or site administrator to assign you sequencing permission.
2. Click **TRACKERS** from the **Project Home** menu.
3. In the folder tree, open the planning folder containing the artifacts that you want to reorder.
4. Click **RANK**.
   You can now drag and drop the artifacts in your planning folder into whatever relative positions you want.

To go back to the sort order you had before (for example, sorted by priority), click **SORT**.
Move a planning folder
A planning folder's position may reflect scheduling changes in relation to the work contained in other planning folders.

For example, when plans change and the work represented in a later planning folder moves ahead of an earlier planning folder, move the folders around in the hierarchy to reflect the new relationship.

When you move a document folder, any documents and subfolders that it contains are also moved to the destination folder. Before promoting a planning folder to a higher level in your hierarchy of folders, make sure its member artifacts (and its members' parent artifacts, if any) are all assigned to the same planning folder, or to a planning folder that belongs to the same folder hierarchy.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the planning folder tree, find the folder that you want to move. Click the title to select it.
3. Choose **Cut** from the **EDIT** menu.
4. In the folder tree, find the folder into which you want to move the selected folder. Click the title of the folder to select it.

   ☞ **Note:** You can move a folder into the root folder or into any other folder, as long as no member artifact ends up having a parent belonging to a higher-level folder in the same tree.
5. Choose **Paste** option from the **EDIT** menu.
   The folder is moved to the destination folder.

Export planning folder artifacts

To use the contents of artifacts from a planning folder in other applications, export them to a `.csv`, `.xml`, or tab-delimited file.

For example, sometimes it can be useful to look at the status of a project by sharing a spreadsheet using Microsoft Excel or Google Spreadsheets.

1. Click **TRACKERS** from the **Project Home** menu.
2. On the **List Trackers, Planning Folders and Teams** page, click **PLANNING FOLDER**.
3. On the **Summary** page, click the desired planning folder.
4. From the **List artifacts** page, click **SORT** or **RANK** as required. You can also select specific set of artifacts to be exported.

   👉 **Note:** When you export artifacts in the rank mode, the parent artifacts are listed along with the children as ranked in the **List Artifact** page. Similarly the sort order in the list view is retained in the exported list as well.

5. Click **Export**.
6. In the **Export Data** window, select an export format that you can import into the other application.
   For example, to use the data in a spreadsheet program, select CSV.
7. Move the fields you are interested in from the **AVAILABLE COLUMNS** list to the **SELECTED COLUMNS** list, then click **Export** to complete the process.

Manage your project with Teams

The Team feature, as the name implies, enables you to create logical groups of team members to carry out project activities more efficiently in an agile environment.
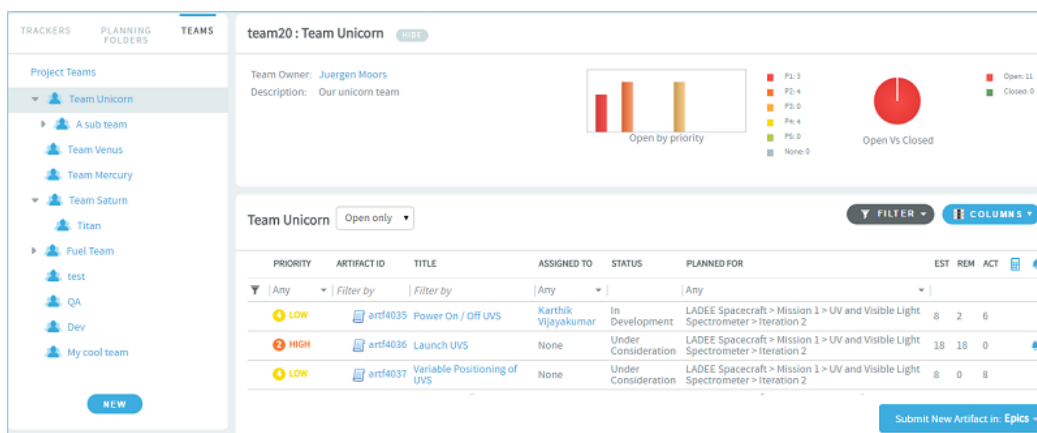
Teams - an overview

Using Teams, project activities can be planned, tracked, collaborated, reported and executed in a more organized and structured manner.

This feature facilitates effective communication among the team members who need to be aware of the changes and latest updates occurring in their projects and act accordingly. For example, using the team's view of backlogs, any project impediment can be communicated to the team and resolved quickly.

The Team list view consists of a tree structure of all the teams in a project, a summary and a filter table of the selected team. The filter table allows you to assign artifacts to a specific team. You can configure your team filter table and filter work items depending on the information you want. The filter table lists only the artifacts for the selected team. If you have selected a parent team, the filter table does not include the artifacts of the subteams (child teams).

👉 **Important:** To access the Team feature, you must have the Tracker 'view' permission.

Team roles and permissions

Other than the Tracker 'view' permission, there are no role permissions that you need to set specifically for Team on the **Permissions** page of **Project Admin**. Depending upon your project role, you can view the appropriate icons on the Team list view and perform Team related tasks.

- **Project administrator:** You can create, edit, delete and view teams.
- **Team owner:** Any team member can be designated as a team owner. You can be a team owner for more than one team. As a team owner, you can only edit and view your team details.
- **Team member:** Any project member can be added to a team. As a team member, you are only allowed to view the team members.

Permissions

The following table lists the various roles and their specific permissions associated with Team:

Permissions	Team member	Team owner	Project admin
Create a team			✔
Delete a team			✔
Add team members while creating a team			✔
Designate a team owner while creating a team			✔
Edit your team details		✔	✔
Add / delete team members while editing your team(s)		✔	✔
Designate a team member as the team owner while editing your team		✔	✔
View a team tree	✔	✔	✔
Assign artifacts to any team / team member	✔	✔	✔

Manage Teams

Using the Team list view, you can create, edit, delete and view teams at the project level.

You can access Teams only if you have the Tracker 'view' permission.

Create a parent team

As a project administrator, you can create a parent team at the root level, **Project Teams**.

1. Click **TRACKERS** from the **Project Home** menu.
2. On the left pane of the default list view, click **TEAMS**.
   All the teams in your project are displayed on the pane.
3. Select **Project Teams** and click **NEW** to create a new team.

   Alternatively, you can hover the mouse on **Project Teams** and click the **Create Team** button.

   The **Create Team** window is displayed.

4. Give a name and description to the new team.

   - The team name is a required field. You are allowed to enter a maximum of 64 characters including spaces.
   - No two teams on the same level in the Team tree hierarchy can have the same name within a project; however, a team can have the same name as that of a deleted team.
5. Add team members by selecting the users from the list displayed.

   Alternatively, you can search for specific users by typing the relevant alphabets and selecting from the list of matching names.

   - The user who creates a team becomes the team owner automatically.
   - A team member can be a part of more than one team.
6. Designate a team member as the team owner by hovering the mouse on the specific name and clicking the **Team Owner** button.

7. Click **Create**.
   A new team is created.

Create a sub team

A team can have many sub teams.

8. On the Team tree view, select the team for which you want to create a sub team and click **NEW** or hover your mouse on the specific team and click the **Create Sub Team** button.

The **Create Sub Team** window is displayed.

9. Provide all the required information, and click **Create**.
   A sub team is created.

Delete a team

As a project administrator, you can delete any team from your project. If a team has sub teams (child teams), you can delete the parent team only after deleting all its child teams.

10. On the Team tree, hover your mouse on the team you want to delete and click the **Delete Team** button.

Edit a team

As a project administrator, you can edit any team in your project. Similarly, if you are a team owner, you can edit the team(s) for which you are the owner.

11. On the Team tree, hover your mouse on the team you want to edit and click the **Edit Team** button.

The **Edit Team** window is displayed.

12. Make the required changes:

    • Edit the team name and description.
    • To add more team members, select the names from the list displayed.
    • To delete team members, hover the mouse on the specific name and click the **Remove Member** button.

    • To designate another team member as the team owner, hover the mouse on the specific name and click the **Team Owner** button.

      👉 **Note:** As a team owner, you may decide to designate a different user as your team owner. In that case, once you update your changes, you will only be a team member and you will no longer have the permission to edit your team.

13. Click **Save**.

View team members

14. On the Team tree, hover your mouse on the team whose members you want to view and click the **View Team Members** button.

👁

The **Team Members** window is displayed.

Team Members   ×

Design (3)

Graham Norton   🔑

Steven Moretti   William Cosby
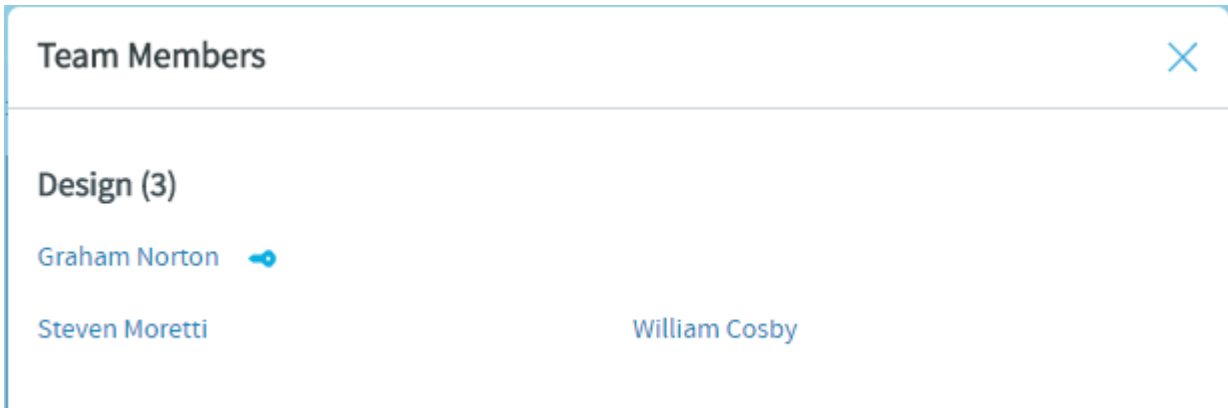
Manage your product release with the Planning Board

The Planning Board is an important tool for your TeamForge project's Agile planning activities. It enables you to plan and monitor the features that are required in each sprint (or iteration), and assign them from the product backlog to specific sprints.

When you've set up your planning folders, you have four views available to work with them:

| LIST | PLAN | TRACK | KANBAN |

- **LIST**: The list view.
- **PLAN**: The planning board view.
- **TRACK**: The task board view.
- **KANABN**: The kanban board view.

The planning board view complements the list view. While the latter offers you capabilities to accomplish various actions such as create, edit, and delete artifacts, planning folders and teams, the former offers product owners (or similar users) the ability to view, rank and move artifacts across the three planning folders (swimlanes) in a physical board-like user interface. In the Planning Board, planning folders are represented as swimlanes. In each swimlane, the tracker artifacts for the selected planning folders are represented as cards. You can also have a team's view of artifacts (backlogs and tasks), which is a swimlane representation of artifact cards for the selected team in the selected planning folder.

TeamForge user roles and permissions that are in place for planning folders apply to all the four views.

Use the Planning Board

You start populating the Planning Board by selecting a planning folder for each swim lane.

The drop-down list for each swim lane displays the hierarchy of active planning folders for all the sprints and scrum teams that are under the project's root planning folder. By default, inactive planning folders are not shown in the swim lane drop-down list. You can organize the Planning Board the way you want by populating the swim lanes with planning folders of interest to you: for example, you may select the planning folder corresponding to the product backlog in the leftmost swim lane and various teams working on the release in the other two swim lanes, or a different sprint in each swim lane. Your selections are remembered for the current session.

In the Planning Board, artifact cards are displayed in ranked sequence within each swim lane. The tasks that you can accomplish while you work with the Planning Board include:

- Adding artifacts (quick add) in select planning folders with minimum required artifact information.
- Editing artifacts.

- Moving artifact cards within a swim lane and ranking them in select planning folders.
- Reassigning (move) artifacts from one planning folder to the other.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the **List Trackers, Planning Folders and Teams** page, click **PLAN**.
   A Planning Board for the current project context is displayed.
3. For each swim lane in the Planning Board, select a planning folder from the drop-down list. The tracker artifacts contained by the planning folders are displayed as story cards within the corresponding swim lanes.

   - If a selected planning folder has subfolders, the artifacts within the subfolders are not displayed.
   - Cards have a color-coded bar to visually identify the priority. In addition, the open and closed cards' background is color-coded to uniquely identify the status.
   - For each card, the artifact ID, title, priority, status, assigned to and points (story points) are displayed. For an artifact, the estimated effort (in hours) is shown only if its points = 0. Relevant tooltips appear when you hover your mouse over these data elements.

   Assistive information such as the number of cards in the selected planning folder and planning folder points capacity are displayed at the top of the swim lane.

   ☞ **Note:** The planing folder points capacity is not displayed for "Iteration" folder types.

   For example, in the following screenshot, the planning folder points capacity is shown as `36 / 100 Pts`. Which means, you planned for 100 points, whereas the current points capacity (the sum of all open and closed artifacts) is 36 points. In other words, you have room for taking up more work in that particular planning folder.
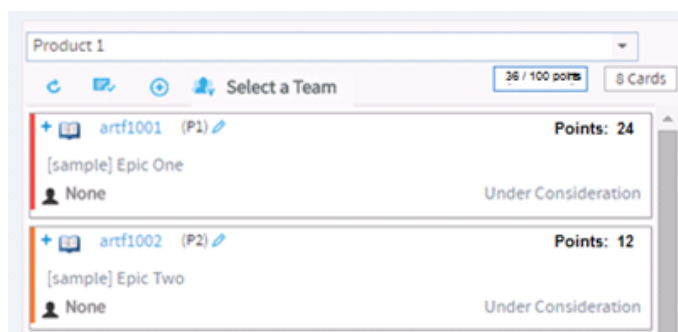
   You can also see `2 Cards` at the top of the swim lane, which is the count of the number of cards in the selected planning folder (by default, only open cards are counted).

   Click the Show/hide closed artifacts toggle button to see the count of both open and closed artifacts.

   Show/hide closed artifacts toggle button

   ☞ **Note:** You must have set the **Points Capacity** while creating planning folders to be able to see the current points capacity against the planned points capacity. For more information, see *Create a planning folder* on page 17.

4. Click the swim lane refresh button to refresh individual swim lanes.

   Swimlane refresh button

5. To know more about an artifact, click the artifact ID link.
   The **View Artifact** page is displayed.
6. By default, closed artifacts are not shown in swim lanes. Click the Show/hide toggle button to show or hide closed artifacts in swim lanes.

   Show/hide closed artifacts button

7. To expand an artifact that has child artifacts, click the "+" symbol in the artifact card. The first, second and third-level child artifacts are displayed.

When you expand an artifact, the parent and child artifacts are visually outlined by means of colored boxes as shown in the following screenshot.
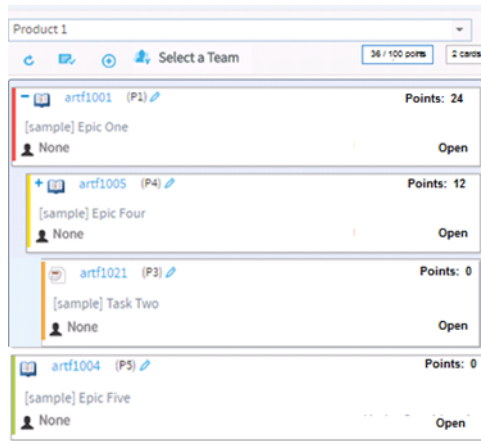
Work with artifact cards

In the Planning Board, artifact cards are displayed in ranked sequence within each swim lane. Using Planning Board, you can quickly add new cards (quick add), edit cards, move cards within a swim lane to rank them and move cards between swim lanes to reassign them to other planning folders.

To quickly add artifacts:

You can add artifacts (quick add) in select planning folders with minimum required artifact information.

👉 **Important:**

- While you quickly add artifacts with data for just three fields such as the tracker type, title and description, the artifacts are, however, saved with default values for other required fields, which you may choose to update later.
- You cannot add artifacts from the Planning Board if a work flow is configured for the meta status change from **New** to **Open**.

1. Select a planning folder in one of the swim lanes. The quick add (+) button at the top of the swim lane is enabled.

👉 **Note:** To be able to add artifacts in Planning Board, you need to have the required permissions.

Quick add button

2. Click the quick add (+) button. The **Create Artifact** window appears.

Create Artifact window

3. Select a tracker type from the **TRACKER** drop-down list.

4. Type a title and description. You can select the **USE TITLE AS DESCRIPTION** check box to copy the title you type to the **DESCRIPTION** text box.

5. Click **Submit**.
   The artifact is added as the bottommost artifact in the swim lane (planning folder). You may choose to rank the artifact (if you have rank permission) or update the artifact with more meta data later. To update the artifact, click the artifact ID link.

To rank a card:

6. If you have the requisite permission, you can drag and drop a card above or below the other cards within a swim lane to position it in the order you want the artifact addressed.

   👉 **Note:** The positioning of cards is retained when you switch between the list and board views (planning, task and kanban).

   For related information on ranking artifacts in the list view, see *Reorder the contents of a planning folder* on page 126.

To reassign (move) a card:

7. In the course of planning your release, you may want to assign an artifact from the product backlog to a specific sprint or team, or move an artifact from one sprint to another. To do this, just drag and drop the artifact card from one swim lane to the other.

   👉 **Note:** When you drop the card, its rank is relative to its position in the new swim lane. However, if a child artifact is moved to a swim lane (planning folder) where its parent is already present, the child artifact is filed under its parent and its rank is relative to the parent's position.

   To be able to move cards between swim lanes, you need to have the permission to update planning folders and rank artifact cards.

To edit an artifact using the artifact ID link:

8. Click the artifact ID link.

   The **View Artifact** page appears.

9. Make the necessary changes and click **Update** to update the artifact and return to the Planning Board.

To quickly edit an artifact:

10. Click the edit artifact button in a card. The **Edit Artifact** window appears.

   👉 **Note:** To be able to edit artifacts, you need to have the required permissions.

Edit artifact button

Edit Artifact window

11. You can edit the following in the **Edit Artifact** window:

- **Artifact title**
- **Artifact description**
- **Primary and secondary attributes**

  You have two expandable frames, **Primary Attributes** and **Secondary Attributes**, that list fields you can edit. Primary attributes include Priority, Status, Assigned To, Team, Estimated Effort, Remaining Effort, Actual Effort and Points. The expandable frames and the fields listed in them depend on your application configuration that you set in Tracker Settings. For example:

  - Configurable fields which are enabled. For more information, see *Enable or disable fields* on page 38.
  - User-defined fields (flex fields) with the field type Text Entry and Single Select and set as **Required** (Tracker Settings > Tracker Administration > Create Field) or workflow configured. For more information, see *Create your own tracker fields* on page 38 and *Create a tracker work flow* on page 40.

  ☞ **Note:** You cannot change the status of an artifact in the Planning Board view, if 'Assigned To' is set as a mandatory field for status change per your tracker work flow settings (Tracker Settings). However, you can edit the status of an artifact in the List view with the same 'Assigned To' work flow constraint.

By default, the following field types which you select while creating a user-defined field (Tracker Settings), are non-editable from the planning board's **Edit Artifact** window: Multiple select, Date picker and User picker.

Non-editable field types

• **Comments**

In this expandable frame, in addition to adding comments, you can view the last five comments for an artifact.

12. After editing an artifact, click **Update**.

Team view in Planning Board
The Team view in Planning Board provides an option to filter artifacts for a specific team within the selected planning folder.

You can select the same planning folder in all three swimlanes. This enables an efficient and faster way of planning a sprint simultaneously for three different teams residing in the same planning folder. You can also select the same planning folder and same team in all three swimlanes for easy handling of artifact cards.

View artifact cards:

1. Click **TRACKERS** from the **Project Home** menu.

2. In the **List Trackers, Planning Folders and Teams** page, click **PLAN**. A Planning Board for the current project context is displayed.

3. Select a planning folder from the drop-down list.

☞ **Note:** You can select the same planning folder in all three swimlanes.

4. View the artifact cards based on your requirement:

   • Select a team from the **Select a Team** drop-down list to view artifacts of a specific team.
   • Select **None** to view unassigned artifacts.
   • Select **All artifacts** to view all the artifact cards residing in the selected planning folder.

   ☞ **Note:** If there are no teams created for a project, then the **Select a Team** drop-down list is not displayed at all.

Assign artifacts to a team:

5. You can select an unassigned artifact from one swimlane and drag it to another where you have a team selected.

   ☞ **Note:** Whenever you move an artifact card from one swimlane to another, all its child artifacts (if any) are also moved along with it.

Manage tasks with Task Board

Task board is an important tool in the Agile process. It helps the team to focus on the work at hand in the current sprint and feed progress data back into the system.

The Task Board is not a sprint-planning tool; it is more of a sprint-tracking tool. Use the *Planning Board* for planning the stories that will be dealt-with in the sprint. During the sprint, team members can use the Task Board to break down the stories into tasks and track them to completion.

For more information, see *What are Planning, Task and Kanban boards?* on page 959

Set up the Task Board
TeamForge project administrators can configure one Task Board per project. Once that is done, project members can use the Task Board.

• During Task Board configuration:

   • Select one or more backlog[#] trackers.
   • Select a task tracker. You can select only one task tracker for a project.
   • Select at least two and up to seven statuses. The number of task swimlanes in the Task Board is equal to the number of statuses you select.
   • Select the unit of effort for backlog and task trackers.

• For more information, see *What are Planning, Task and Kanban boards?* on page 959

1. Click **TRACKERS** from the **Project Home** menu.

[#] Backlog trackers are tracker items such as epics, stories, defects and so on, for which tasks can be created.

2. In the **List Trackers, Planning Folders and Teams** page, click **Track**.

A Task Board for the current project context is displayed with the following message.

```
Task Board is not configured for the project.
```

3. Click the Task Board configuration icon.

The **Settings** window appears.

4. Select one or more backlog trackers and a task tracker.

☞ **Note:** You can always revisit your backlog and task tracker settings for Task Board and change them if required at any future point in time.

5. Click **Next**.

6. Select at least two and up to seven statuses from the **Available** list box. Press and hold the **Ctrl** key to select multiple items.

7. Click the forward arrow to add the selected statuses to the **Selected** list box.

☞ **Note:**

- You can always revisit your status settings for Task Board and change them if required at any future point in time.

- Use the forward and backward arrows to move statuses to and fro the **Available** and **Selected** list boxes.
- Use the up or down arrows to move statuses up or down within the **Selected** list box.

8. Click **Next**.

9. Select the task effort field and backlog item size field.

    ☞ **Note:** If the effort field is disabled for the backlog and task trackers, it is not possible to select the task effort and backlog item size fields and the value is set to 'None'.

10. Click **Finish**.

    The following success message appears: `Task Board preferences saved successfully.`

You have now successfully configured the Task Board for your project. Project members can start using the Task Board to manage their tasks.

Use the Task Board

During a sprint, TeamForge project members can use the Task Board to view tasks, create tasks for backlog items, edit tasks and drag and drop tasks across swimlanes as they progress.

With the Task Board, you can:

- View backlog items of a selected planning folder in the **Backlog Items** swimlane, which is the leftmost swimlane of the task board.
- View tasks (belonging to backlog items) pinned to swimlanes based on the status.
- Edit backlog items and tasks by clicking the artifact ID link.
- Move tasks (cards) from one swimlane to the other as tasks progress.
- Add new tasks for backlog items.

1. Click **TRACKERS** from the **Project Home** menu.

2. In the **List Trackers, Planning Folders and Teams** page, click **TRACK**.

A Task Board for the current project context is displayed.

3. Select a planning folder from the drop-down list.

    The Task Board is populated with backlog items and their tasks.

    - If a selected planning folder has subfolders, the artifacts within the subfolders are not displayed.
    - Cards have a color-coded bar to visually identify the priority. In addition, the open and closed cards' background is color-coded to uniquely identify the status.
    - For each card, the artifact ID, title, priority, assigned to and effort (in terms of 'Points' for backlog items and 'Hours' for tasks) are displayed. Status information is shown on backlog cards only. Relevant tooltips appear when you hover your mouse over these data elements. You can also hover your mouse over the tracker icon in backlog and task cards to know the tracker name.

Add new tasks

4. Click the '+' icon.

The **Create Task** window appears.

5. Type a title and description and click **Submit**.

> 👉 **Tip:** To use the title you type as the description, select the **Use Title as Description** check box.

To edit a backlog item or task using the task ID link:

6. On the **List Trackers, Planning Folders and Teams** page, select the planning folder.

7. From the listed backlog items and tasks, click the artifact ID link you want to edit.

The **View Artifact** page appears.

8. Modify the backlog item or task and click **Update** to update the artifact and return to the Task Board.

To quickly edit a task

9. Click the edit artifact button in a task. The **Edit Task** window appears.

> 👉 **Note:** To be able to edit tasks, you need to have the required permissions.

Edit task button

Edit Task window

10. You can edit the following on the **Edit Task** window:

- **Artifact title**
- **Artifact description**
- **Primary and secondary attributes**

  You have two expandable frames, **Primary Attributes** and **Secondary Attributes**, that list fields you can edit. Primary attributes include Priority, Status, Assigned To, Estimated Effort, Remaining Effort, Actual Effort, Points and Team. The expandable frames and the fields listed in them depend on your application configuration that you set in Tracker Settings. For example:

  - Configurable fields which are enabled. For more information, see *Enable or disable fields* on page 38.
  - User-defined fields (flex fields) with the field type Text Entry and Single Select and set as **Required** (Tracker Settings > Tracker Administration > Create Field). For more information, see *Create your own tracker fields* on page 38.
  - Fields which are workflow configured. For more information, see *Create a tracker work flow* on page 40.

  By default, the following field types, which you select while creating a user-defined field (Project Admin > Tracker Settings), are non-editable from the task board's **Edit Task** window: Multiple select (**Multiple Select**), Date picker (**Select Date**) and User picker (**Select User(s)**).

- **Comments**

In this expandable frame, in addition to adding comments, you can view the last five comments for an artifact.

11. After editing an artifact, click **Update**.

Move tasks across swimlanes

12. To move tasks across swimlanes, drag a task card from the source swimlane and drop it on the destination swimlane.

☞ **Note:** You need to have task tracker 'edit' permission to move cards across swimlanes.

View both 'open' and 'closed' backlog items

13. ☞ **Note:** When you select a planning folder, the Task Board shows only 'open' backlog items by default.

Click the 'Show/hide closed artifacts' toggle button to view both 'open' and 'closed' backlog items.

Show/hide closed artifacts toggle button

Refresh the Task Board

14. Click the following 'Refresh' button to refresh the Task Board.

Auto assign task

If a task is not assigned to anyone, that is, if it is assigned to **None**, then it can be assigned to the user who has logged in using the **Auto Assign Task to Me** feature. This check box appears only if:

• The Task Board is configured.
• The user who logged in has the edit permission to the task tracker.
• The project is not locked.

15. Select the **Auto Assign Task to Me** check box and drag the artifact from one swimlane to another (from one status to another) and the artifact is automatically assigned to you.

Position artifact cards

16. If you have the requisite permission, you can drag and drop a card above or below the other cards within a swim lane to position it in the order you want the artifact addressed.

☞ **Note:** The positioning of cards is retained when you switch between the list and board views (planning, task and kanban).

For related information on ranking artifacts in the list view, see *Reorder the contents of a planning folder* on page 126.

Team view in Task Board

The Team view in Task Board provides an option to filter backlog items for a specific team within the selected planning folder.

This team's view of artifacts applies only to backlog items. When you filter and view the backlog items for a specific team, the tasks (if any) within the filtered backlog items are also displayed in the relevant status swimlanes. However, you cannot filter tasks alone for a team.

You can move the artifact cards between swimlanes. If the artifacts are workflow configured, appropriate error messages are displayed when a workflow rule is broken.

1. Click **TRACKERS** from the **Project Home** menu.

2. In the **List Trackers, Planning Folders and Teams** page, click **TRACK**.
A Task Board for the current project context is displayed.

3. Select a planning folder from the drop-down list. All the backlog items and their tasks in the selected planning folder are displayed.

4. Select a team from the **Select a Team** drop-down list.

☞ **Note:** If there are no teams created for a project, then the **Select a Team** drop-down list is not displayed at all.

All the backlog items and their tasks in the selected team are displayed.

Manage your project workflow with Kanban Board

The Kanban Board is a project management tool, which gives you a snapshot of the work items, in which states they are, how they are progressing and if there is any bottleneck to be cleared for a smooth delivery of the product on time.

Set up Kanban Board

As a project administrator, you can configure your Kanban Board based on your project requirements.

Create Kanban Board states for the different stages of your project development process, specify workflow constraints for each state and map them to your tracker statuses.

1. Click **TRACKERS** from the **Project Home** menu.
2. In the **List Trackers, Planning Folders and Teams** page, click **Kanban**.

   The following message is displayed prompting you to configure a Kanban Board for the current project.

   ```
   Kanban Board is not configured for the project.
   ```
3. Click the Kanban Board configuration icon to open the **Settings** window.

   ☼

   ☞ **Note:** Use the configuration icon when you create a Kanban Board for the first time or when you want to modify the settings of your current Kanban Board.

   Alternatively, you can use **Manage Boards** to create a new board.

   ▦

   For more information about its usage, see *Manage Boards*.
4. Give a name to the new Kanban Board.
5. Select the trackers whose artifacts you want to view on your Kanban Board. (These are the trackers you created for your project. For more information on how to create a tracker, see *Create a tracker* on page 16.)
6. Click **Next**.

Kanban states and constraints

A kanban state is the status of a work item in a value stream (swimlanes). The constraints or the limits that you specify here dictates the behaviour of the board. These are checkpoints which eventually help you track the progress of a work item, identify bottlenecks and fix them. The default kanban statuses are 'Not Started', 'In Progress' and 'Closed'.

You can create your own kanban statuses, delete any state including the default one and rearrange the order in which they would appear as swimlanes on your Kanban Board.

☞ **Note:** If there are only three kanban statuses, the **Delete Status** icon is disabled disallowing you to delete any more because you must maintain a minimum of three statuses.
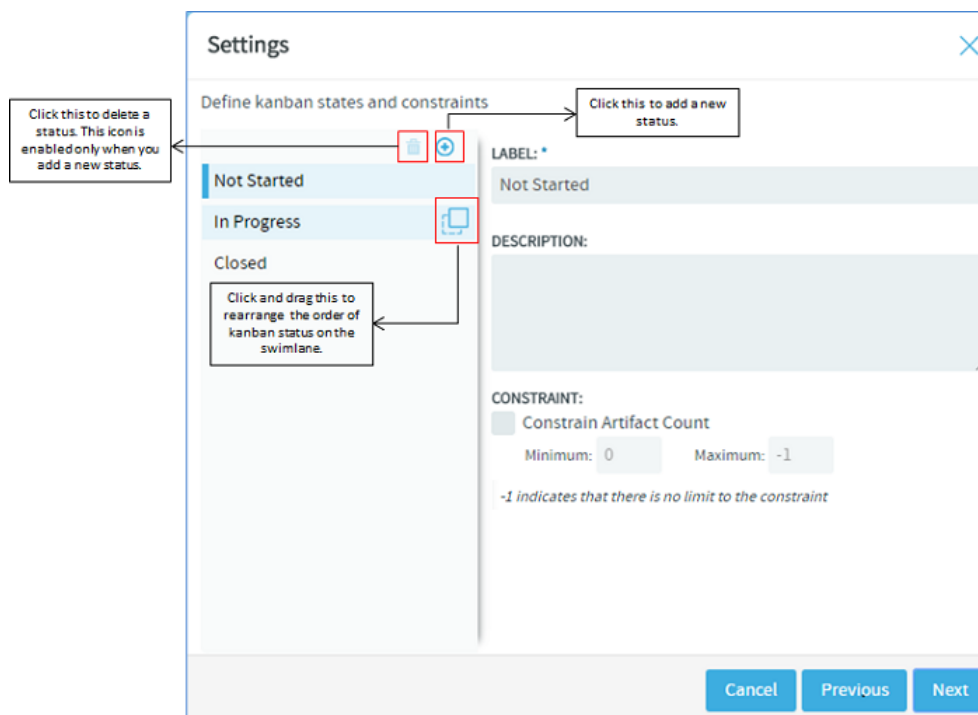
7. Create and configure kanban statuses.

   a) Click the **Add Status** icon. You can have a maximum of 20 statuses.

   b) Give a label and description to the new status.

   > 👉 **Note:** The kanban status label can have a maximum of 64 characters and the description, a maximum of 256 characters. Alpha numeric and special characters are allowed.

   c) Select the **Constrain Artifact Count** check box to specify the minimum and maximum workflow constraints, that is, specify the minimum and maximum number of work items (artifact cards) that ought to be present in each status. You may specify these values based on the number of work items in the queue versus the number of resources available.

   Remember the following when you set the minimum or maximum value to -1.

Constraint value	Logic	Example
Minimum value = -1	This is treated as zero. Therefore, though the maximum value should be greater than the minimum value, the following value range is allowed: Minimum = -1 Maximum = 0 and above	Let us assume that you are in the last iteration of your release. As a project manager, you would not want to see any artifact in the 'Impeded' status. So you set the minimum to -1 and maximum to 0, which translates to zero artifacts. When this constraint is applied, your Kanban Board flags a violation, if artifacts show up in the 'Impeded' status, thereby drawing your attention to address the issue immediately.
Maximum value = -1	This indicates there is no limit to the maximum value.	You may choose to set -1 as the maximum constraint for the 'Closed' status because you would want to see as many closed artifacts as possible.
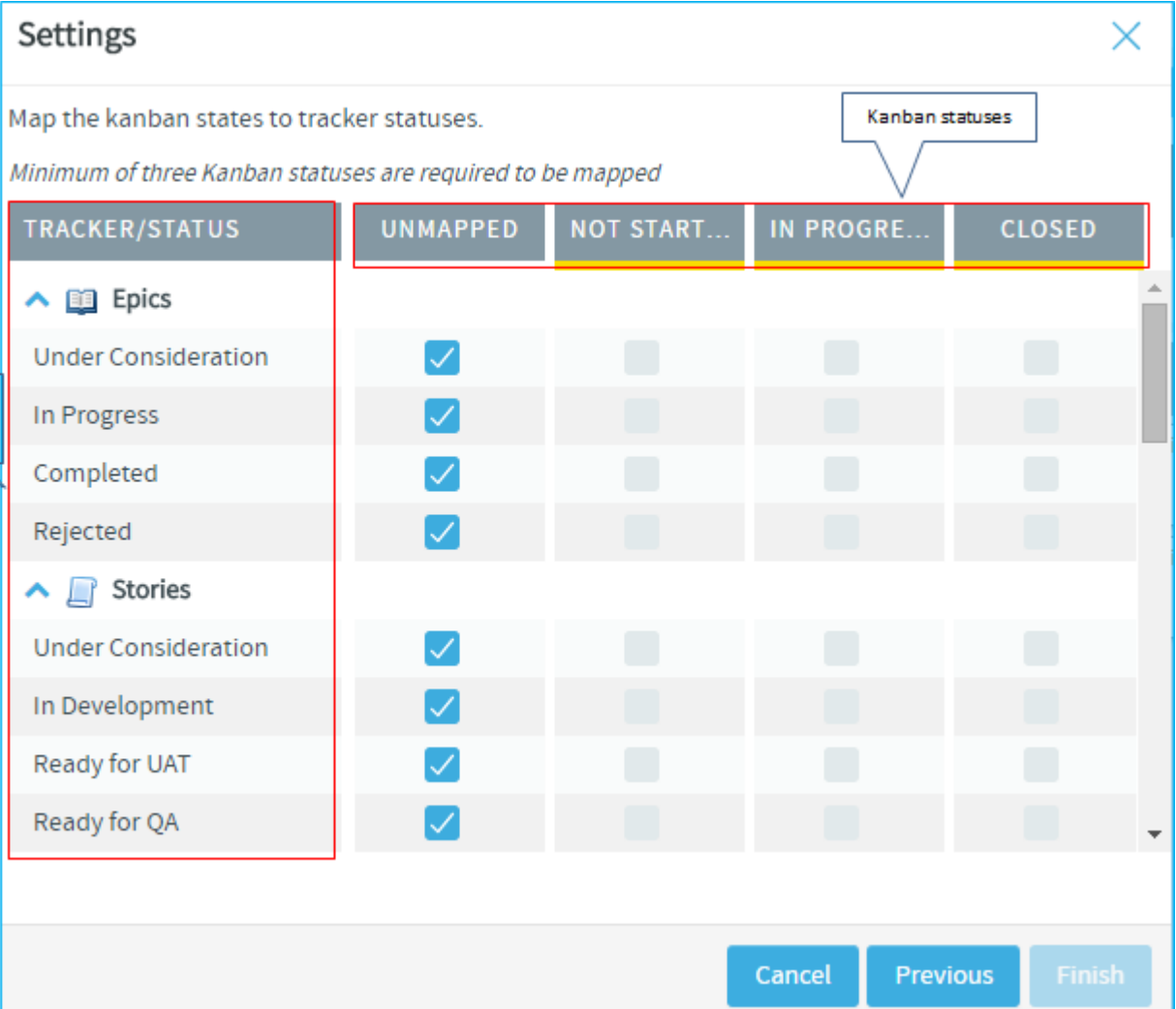
☞ **Important:** These constraints are applicable only at the planning folder level and not at the team level because the Kanban Board is expected to give an overall visibility to the work items for an iteration or release within a planning folder and not just within a specific team.

Map kanban states with tracker status

Next, make a logical mapping of the kanban states with the tracker statuses to view those tracker artifacts on your Kanban Board. For example, to view defects, you must first make a meaningful mapping of the kanban states with the defect tracker status(es). The mapped kanban states appear as swimlanes on the Kanban Board.

The trackers (you had selected in *Step 5*) and their statuses appear on the left. All the kanban statuses appear at the top. The tracker statuses are unmapped by default.

- You need to map a minimum of 3 kanban statuses.
- You can map a single kanban status to one or more tracker statuses, but you cannot map a single tracker status to more than one kanban status.

8. Click the appropriate check box. For example, for an Epic tracker, map 'Completed' and 'Rejected' tracker statuses with the Kanban statuses 'Closed' by selecting the appropriate check boxes as shown in the following screenshot.

9. Click **Finish** to complete the process.

Manage Boards

You can create and manage multiple kanban boards per your requirement. As a project administrator, you may want to create multiple boards at the iteration level, at each unit level, (Dev, QA and so on), at the hierarchical level or at each project team level, depending upon the needs of the users. For example, to get the big picture and make executive decisions you may want to create a kanban board at the management level; at the same time, at a project member level, you can create a board to focus on the immediate tasks to be completed.

10. Use the various icons available in **Manage Boards** to maintain multiple kanban boards in a project.

To do this...	Click this icon
To use **Manage Boards**.	▤
To create a new board	⊕
To edit a board	✎
To delete a board
👉 **Note:** You cannot delete the default Kanban Board. When you delete your current active board, the default one shows up automatically.	🗑

To do this...	Click this icon
To set a default board	☑

Use Kanban Board

After the kanban board is configured, you can use it to view, plan and track work items for the selected planning folder or the selected team within that planning folder.

1. Click **TRACKERS** from the **Project Home** menu.

2. In the **List Trackers, Planning Folders and Teams** page, click **Kanban**.

LIST	PLAN	TRACK	KANBAN

3. Select the required kanban board from **Manage Boards** (available only for project administrators). If you are a project member, select your kanban board from the **Select Board** list.
   The selected kanban board for the current project context is displayed.

4. Select the planning folder for which you want to view the status of the artifacts.

   If your project contains teams, the **Select a Team** drop-down list appears. From this drop-down, select **All artifacts** or if you want to view artifacts specific to a team within the selected planning folder, select the relevant team.

   Depending upon your selection, artifacts pertaining to the mapped trackers are displayed in the appropriate swimlane.

   - You can view a maximum of 5 swimlanes at a time on your Kanban Board. If there are more, use the carousel scroll to slide across the swimlanes. You can expand or collapse each swimlane. When there are no results to display in a swimlane, the **Expand/Collapse** button does not appear.
   - Based on the configuration values, if the constraints are violated, the relevant status headers are highlighted appropriately. A violation of minimum constraint is highlighted in orange indicating that resources are underutilized whereas that of maximum constraint is highlighted in red indicating resources being overloaded and bottlenecks to be fixed. For more information on how a kanban board is configured, see *Set up Kanban Board* on page 142.

   - Each swimlane header displays the status label, the total number of artifacts within the selected planning folder, and the minimum and maximum constraints that were configured for the kanban state.

   In the above scenario, for the 'In Progress' status:

'3' is the total number of artifacts in the 'In Progress' status within the selected planning folder.

'2' indicates the minimum constraint and '*' indicates the maximum constraint '-1'. (-1 as the minimum constraint translates to 0 whereas -1 as maximum indicates there is no limit and so represented by an *.)

> ☞ **Important:** These values and constraint validation are planning folder specific and not team specific. For example, in the above scenario, the total number of artifacts is the overall count of artifacts which are in the 'In Progress' status within the selected planning folder and not within the selected team.

- Each artifact card displays its points (story points) or estimated effort if these fields are enabled in your tracker settings. If both are enabled, only the estimated effort is displayed hovering the mouse on which the artifact's points is displayed. Similarly, if all the effort fields (estimated, remaining and actuals) are enabled, you can view them by hovering the mouse on the estimated effort.
- Cards have a color-coded bar to visually identify the priority. In addition, the open and closed cards' background is color-coded to uniquely identify the status.
- A parent artifact card displays the count of its child artifacts (**Show/Hide child artifacts** button). Click this button to view or hide the child artifacts. If there are more than 5 child artifacts, when you expand the parent artifact, you will see pagination arrows ('Previous'/'Next') at the bottom of the parent artifact card.

- If the tracker types of both the parent and child artifacts have been mapped with the kanban statuses, the child artifacts appear within their parent artifact card and also as an individual card. In the above scenario, Epic (parent artifact) and Story (child artifact) trackers have been mapped to the kanban status 'In Progress'. Story 'artf1014' is the child artifact of the epic 'artf1002'. So you will see 'artf1014' within the parent artifact card as well as outside of it as an individual artifact card.
- When a child artifact is closed, a 'Closed' tag appears next to the child artifact ID within the parent artifact card.

Edit artifacts

Use Kanban Board to edit an artifact using the Edit icon on the artifact card or move artifacts from one status to another appropriately. Based on the status changes you make, the swimlane headers get updated appropriately.

👉 **Remember:**

- You can edit an artifact only if you have the tracker edit permission. Otherwise, you can only view it.

5. When you drag the artifacts from one swimlane to another, note the following validations:

- The changes you see in swimlane headers do not apply to any specific team but takes into account the total count of artifacts within the selected planning folder.
- You cannot move an artifact card to a status that is not mapped to that particular tracker type. For example, a 'Ready for QA' kanban status may not have been mapped to any of the epic tracker statuses. So when you attempt to move an epic to that status, you will get an `Invalid status configuration` error.
- You can move a parent artifact to the 'Closed' status only if all its child artifacts are closed. However, a child artifact is independent of its parent with regard to the status change, that is, when you move a parent artifact from one status to another, the status of the child may still remain unchanged. For example, an epic may have many stories, the status of some may change from 'Not Started' to 'In Progress' whereas some may not. So when the status of an epic as one single unit may change, it need not apply to all of its child artifacts.
- When your move violates a constraint (minimum or maximum), a warning is displayed; but you can still move the artifacts.
- If a kanban status is mapped to more than one tracker status, when you move an artifact, you have an option to choose a status as shown in the following screen shot.

Create artifacts and child artifacts

Using the Create Artifact icon on the top right of your kanban board, you can quickly create an artifact. The Create Artifact icon appears only when you configure a kanban board and select a planning folder.

Similarly, using the Create Child Artifact icon available on the artifact card, you can quickly create a child artifact for any artifact card displayed on Kanban Board.

👉 **Note:** The Create Child Artifact icon does not appear on closed artifact cards.

You can create an artifact or a child artifact only if you have the required tracker permission.

6. Click the required icon: Create Artifact or Create Child Artifact.

➕

7. Enter the required information in the relevant window and click **Submit**.

- Only trackers configured for the kanban board appear in the Trackers drop-down list.
- While you quickly add artifacts with data for just three fields such as the tracker type, title and description, the artifacts are, however, saved with default values for other required fields, which you may choose to update later. If the default state of the selected tracker is not mapped for the newly created artifact or child artifact, the artifact card does not show up on the kanban board.

Show / Hide closed cards

8. When you have a large volume of closed cards in a planning folder, you can restrict the number of closed cards you want to view by toggling between the 'Show all closed artifacts' icon:

Or the 'Hide artifacts older than 60 days' icon using which you can hide closed cards older than 60 days. This is the default option:

☞ **Note:**

- The Show/Hide toggle icon appears only on configured kanban boards.
- Your selection is saved for the subsequent sessions as well.

Position artifact cards

9. If you have the requisite permission, you can drag and drop a card above or below the other cards within a swim lane to position it in the order you want the artifact addressed.

☞ **Note:** The positioning of cards is retained when you switch between the list and board views (planning, task and kanban).

For related information on ranking artifacts in the list view, see *Reorder the contents of a planning folder* on page 126.

Manage task work flow

As a task administrator, you can control how users move tasks along under various conditions.

Alert project members about overdue tasks

If a task is not completed by its proposed end date, you may want to send an automatic reminder to the task's owner, its administrator, or other project members.

You can choose which of the people involved with the overdue task get the alert, and how late the task must be to trigger an alert.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Task Manager Settings**.
3. Click the **ALERTS** tab.
4. Specify the number of days after a task's due date that an overdue task alert is sent.
5. Specify the recipients of overdue task alerts.

    - **Project Role Members** - All project members who are assigned that role.

        ☞ **Note:** If you select roles or project members that do not have the task view permission, these project members will receive Permission Denied errors when attempting to access overdue tasks.

    - **Specific Project Members** - Click the Search icon to select from a list of project members.
    - **Task Creator** - The person who originated the task.

        ☞ **Note:** The person to whom the task is assigned receives overdue task alerts automatically.

6. Click **Save**.

Overdue task alerts are now enabled for all tasks in the project.

Require approval for changing tasks

During times when changes to tasks carry a high risk, task administrators can prevent users' changes from taking effect until the administrator approves them.

Requiring change approval keeps you informed about changes to task status, dates, dependencies, and other changes that can affect the project plan.

You can select the fields for which you want require change approval. Whenever a user makes a change to a field for which change approval is required, the change request is placed in the **Change Requests** section of the **Task Summary** page, with a link in the **Items Pending My Approval** section of each task administrator's **My Page**.

☞ **Note:** Changes made by task administrators do not require approval.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Task Manager Settings**.
3. Click the **CHANGE APPROVAL** tab.
4. Select the fields for which you want to require change approval.
5. Click **Save**.

Change approval is now in force for all future changes made to the selected fields.

Handle a change request

When change approval is in force, changes made by project members to selected task fields must be approved or rejected by a task administrator.

Change requests are placed in the **Change Requests** section of the **Task Summary** page. Change requests are also displayed in the **Items Pending My Approval** section of each task administrator's **My Page**.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, click the **CHANGE REQUESTS** tab.
3. Respond to a change request in one of these ways:

   • If you do not need to view the request details, select the task and click **Approve Selected** or **Reject Selected**.
   • To view the request details or add a comment before responding to the change, click the task name and write your comments, then click **Approve Selected** or **Reject Selected** in the **Change Requests** section of the **Task Details** page.

   ☞ **Note:** If you approve a task end date change that impacts dependent successor tasks, you must confirm your changes for each of the successor tasks individually.

You can also approve or reject change requests from the **ITEMS AWAITING MY APPROVAL** section of your **My Page**. In the **Tasks Awaiting Approval** section, click the task name to view the request details, add a comment, and approve or reject the request.

Indicate task status with colors

The status of each task is represented by a color. The status color of a task folder is an accumulation of the status colors of the tasks in the folder.

When a certain percentage of tasks have turned a given status color, the color "rolls up" to the task folder. The status colors of the task folders also "roll up" to the project according to the same criteria.

The percentage in each case is chosen by the project admin.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Task Manager Settings**.
3. On the **STATUS ROLLUP** page, set your color-coding criteria.

   For example:

   • A task folder is represented by red when at least 10 percent of its tasks are red.
   • A task folder is represented by green when at least 10 percent of its tasks are green.
   • A task folder is represented by gray only when at least 80 percent of its tasks are complete.

   When one or more criteria are met simultaneously, the task folder is represented by the most critical of the applicable status values. For example, using the criteria above, if 10 percent of the tasks in a task folder are yellow and 75 percent of the tasks in the same folder are gray, the task folder is represented by yellow.

Measure task workloads

For help in planning and allocating resources, look at the project's task workloads.

You can measure the tasks assigned to each project member by status, due date, and estimated and actual effort.

1. Click **TASKS** in the project navigation bar.
2. On the **Task Summary** page, click the **WORKLOAD** tab.
3. Find the following information for each project member:

   - The number of assigned tasks, organized by status value.
   - The total number of incomplete tasks.
   - The total estimated and actual effort for all tasks.

Set the default task calendar

For each task, you can specify whether work is tracked son a five-day or a seven-day calendar.

For each TeamForge project, you can specify a default task calendar. This selection will be the default calendar for new tasks. You can override this value for individual tasks.

Five-day calendars count only Monday through Friday. Seven-day calendars also count Saturday and Sunday.

☞ **Note:** Changing the default calendar does not update existing tasks.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu** , click **Task Manager Settings**.
3. Click the **DEFAULT TASK CALENDAR** tab.
4. Choose a five-day or seven-day calendar as the default for the project.
5. Click **Save**.

Your default task calendar will be applied to all new tasks.

Reporting in TeamForge

Generate a report to get a snapshot of what is going on in a project. You can generate reports on data stored in both TeamForge's production (operational) database or datamart. Datamart, also known as the Reporting database, is built by extracting, transforming and loading (ETL) TeamForge's production data to a separate database (datamart) at regular intervals.

☞ **Important:**

Unless otherwise stated, you must have datamart enabled on your site to create reports in TeamForge. Note that a few *Distribution reports* use data from TeamForge's operational database and a few *Activity Reports* use data from the Orchestrate's event data store.

You can specify the time at which the reporting data is refreshed from the production database. By default, the extraction takes place daily at 2:30 a.m. in the TeamForge application server's time zone. See *Schedule data extraction for reporting* on page 577.

Configure a collection of variables in the `site-options.conf` file to use site-wide reporting functionality. See *Turn on site-wide reporting* on page 587 and the TeamForge Administration Guide for more details.

You can use reports to display data and group relevant information appropriately and specify intervals at which the datamart extracts TeamForge data from the datamart. For advanced reporting options and datamart information, see *Advanced reporting and datamart access* on page 197.

You can also use external reporting tools to connect to the datamart and generate customized reports. See *Datamart access using external tools* on page 207.

Reporting framework

Here's a list of some of the advantages of the new reporting framework:

- Reporting under one umbrella with a central dashboard of reports.
- Cross-project reporting capability.
- Tracker custom defined fields are in data-mart and can be used to filter.

- All events are captured via Orchestrate event data-store.
- You can query and write custom reports.
- All data, including event, associations, and traceability data, can be queried through an elegant API for custom reporting and data extraction.
- High charts-based interactive data visualization charts. You can hover over the charts to see data points, click legends in the chart to toggle specific data point in and out of the chart and so on.
- You can drill one level down on some of the activity reports using column charts to get more clarity on the data points of your interest.
- Categorization of reports.
- Improved usability: Hassle free report creation with new widgets for selecting report criteria such as planning folders, trackers and repositories.
- Ability to save 'Public' and 'Private' reports: While 'Public' reports are visible to all project members with view reports permission, 'Private' reports are only for your consumption. You cannot publish 'Private' reports in project pages.

Reports: Role based access control

In general, TeamForge site and project administrators, and users with tracker and task view permissions can see the **REPORTS** button on the project navigation bar.

- **Object-based access permissions**: If you are a TeamForge user, your ability to create, edit, preview or view reports depends on whether you have permission to access specific objects such as trackers and repositories in TeamForge. For example, to generate a report on the number of SCM commits in a repository, you must have permission to access the repository. Otherwise, a message such as `You do not have sufficient permission to perform this operation` is displayed.
- **Task and Tracker reports (Table reports)**: **Task and Tracker View** permission is required to generate task and tracker reports. You must have **View Activity** permission or object-level permission to view Activity reports such as SCM Commits, Build Activity, Build and Test Activity, Artifact Created and Artifact Closed reports.
- **Context-sensitive access permission**: Reports are shown or hidden from users based on the context. For example, a user must have **View Project Page** permission to view reports published on a project home page.
- **Deleting reports**: Site and project administrators can delete reports. In addition, you can delete your own reports.

Reports available in TeamForge
Reports in TeamForge are grouped under the following categories.
Activity reports
Here's a list of TeamForge activity reports such as the SCM commits, artifact created and closed reports.

These are reports to track activities such as SCM commits, artifact creation and closure, build and test activities and so on. Some of these reports are powered by data from Orchestrate's event data store.

> ☞ **Note:** You can also drill one level down on some of the activity reports using column charts to get more clarity on the data points of your interest.

SCM Commits
Shows the number of commits in one or more selected repositories in a given time period.
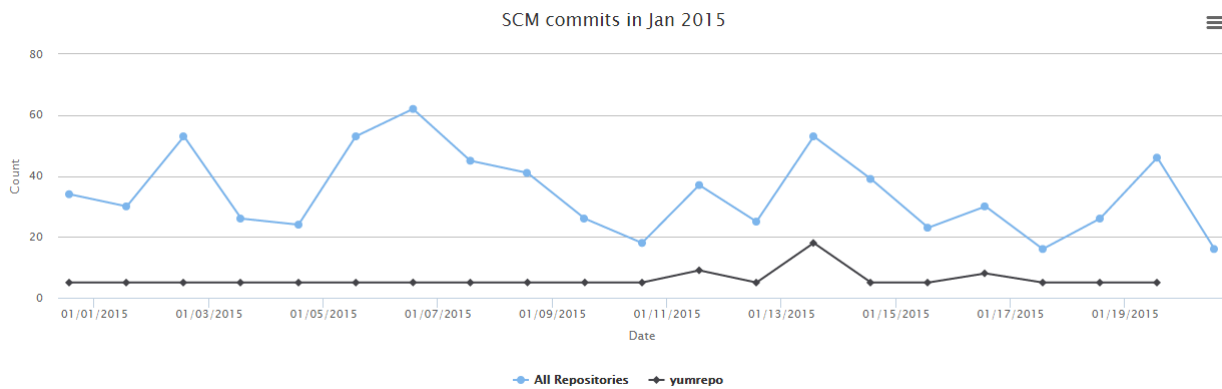
1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Scm Commits** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more repositories from the **SELECT REPOSITORY(S)** drop-down list.

6. Select date range and click **Apply**.

   To set the date range, select the start date first and then the end date. Use the month/year navigation arrows to select the month/year you want. If you don't have the exact dates at hand, you can also select one of the time periods from the **Select Range: Custom** drop-down list.

7. Select a chart type such as 'Line' or 'Bar' or 'Column' from the **CHART DISPLAY TYPE** drop-down list.
8. Select report visibility: Public or Private.
9. Click **Preview**.

10. Click **Create**.
    The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick
    function icon.

    Print chart
    Download PNG image
    Download JPEG image
    Download PDF document
    Download SVG vector image

11. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Created
    Shows the number of artifacts created in one or more selected trackers in a given time period.
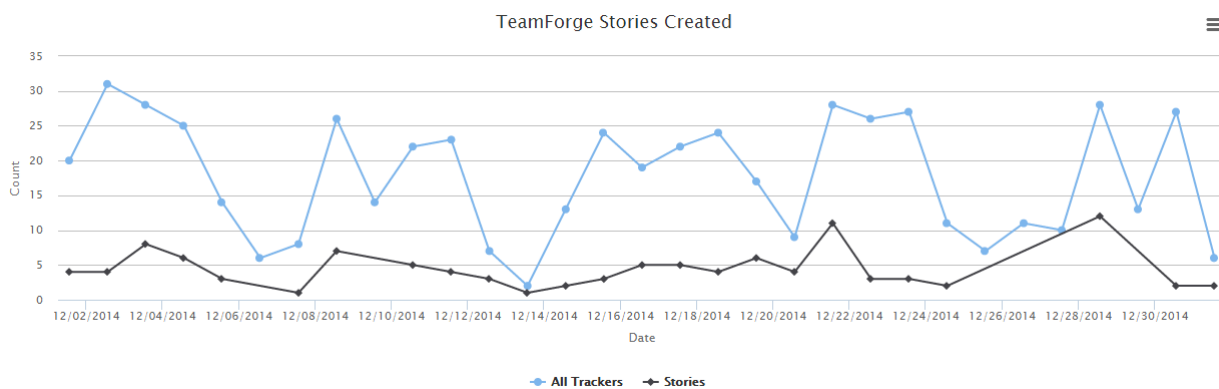
1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
    The **Select Report Type** page appears.
3. Select **Artifact Created** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.

6. Select date range and click **Apply**.

    To set the date range, select the start date first and then the end date. Use the month/year navigation arrows to select the month/year you want. If you don't have the exact dates at hand, you can also select one of the time periods from the **Select Range: Custom** drop-down list.

7. Select a chart type such as 'Line' or 'Bar' or 'Column' from the **CHART DISPLAY TYPE** drop-down list.
8. Select report visibility: Public or Private.
9. Click **Preview**.
10. Click **Create**.

    The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

11. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Closed

Shows the number of artifacts closed in one or more selected trackers in a given time period.

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Artifact Closed** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.

6. Select date range and click **Apply**.

   To set the date range, select the start date first and then the end date. Use the month/year navigation arrows to select the month/year you want. If you don't have the exact dates at hand, you can also select one of the time periods from the **Select Range: Custom** drop-down list.

7. Select a chart type such as 'Line' or 'Bar' or 'Column' from the **CHART DISPLAY TYPE** drop-down list.

8. Select report visibility: Public or Private.

9. Click **Preview**.

10. Click **Create**.
   The report is created and the **View Report** page appears.

   Print or download charts

   You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

11. Click **Back to Reports List** to go back to the Reports dashboard.

Activity Over Time
   Shows the count of various project activities over a period of time such as commits, reviews, builds, and so on. This report is powered by data from the Orchestrate's event data store.
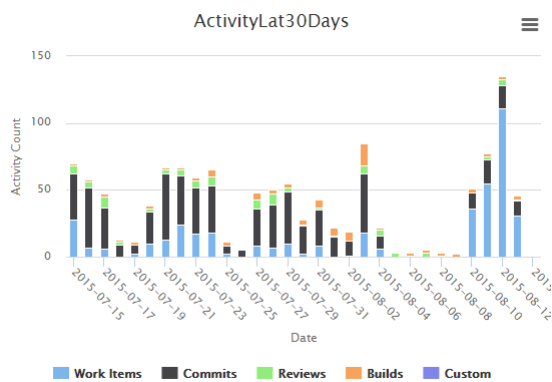
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.

3. Select **Activity Over Time** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more projects from the **PROJECT(S)** drop-down list.

6. Select the number of days from the **LAST N DAYS** drop-down list.
7. The default **Display Type** is **EventsDual**.
8. Select report visibility: Public or Private.
9. Click **Preview**.
10. Click **Create**.
    The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick
    function icon.

11. Click **Back to Reports List** to go back to the Reports dashboard.

Build Activity
    Shows the build activity over a period of time including the count of builds and the average build duration. This
    report is powered by data from the Orchestrate's event data store.
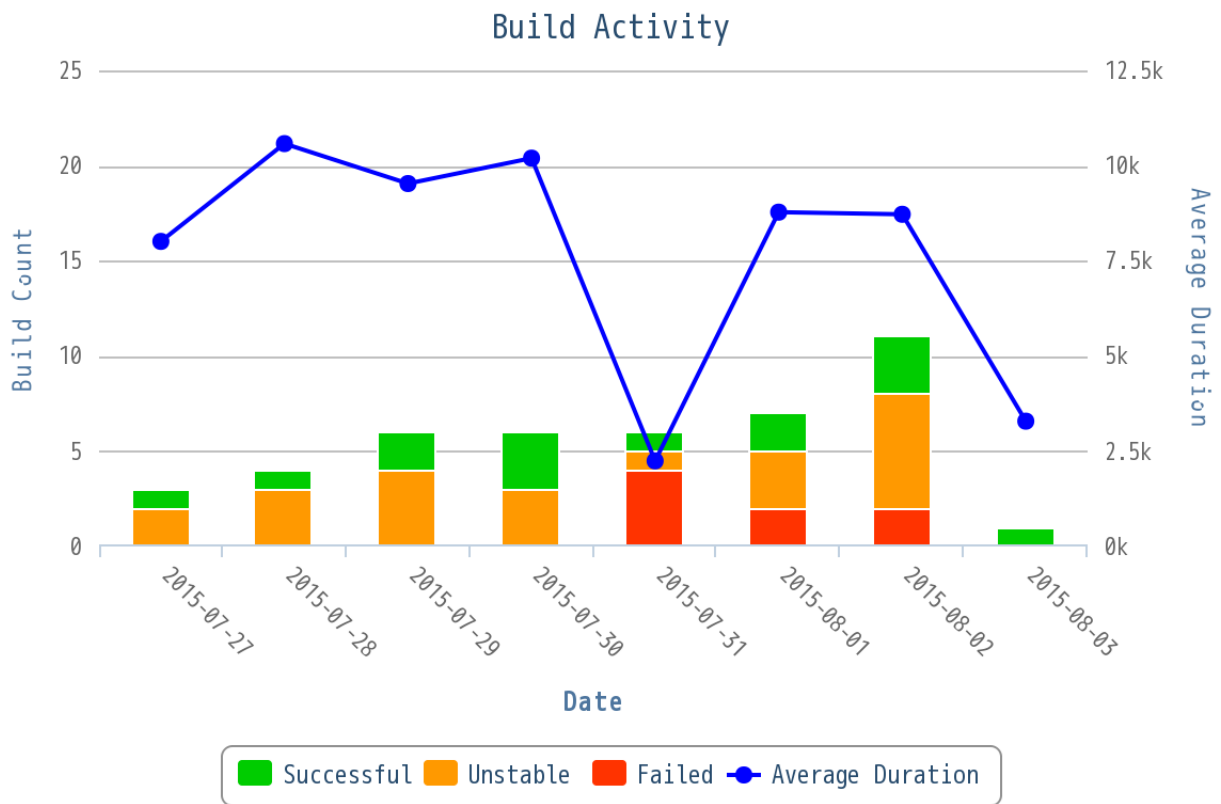
Build Activity

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Build Activity** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more projects from the **PROJECT(S)** drop-down list.

6. Select the number of days from the **LAST N DAYS** drop-down list.

7. The default **Display Type** is **EventsDual**.

8. Select report visibility: Public or Private.

9. Click **Preview**.

10. Click **Create**.

    The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.
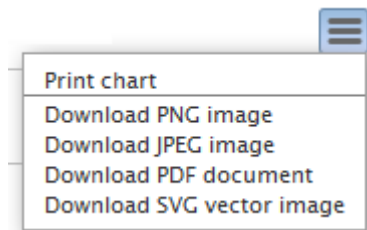
11. Click **Back to Reports List** to go back to the Reports dashboard.

Build Activity by Project

Shows the number of builds over a period of time. This report is powered by data from the Orchestrate's event data store.
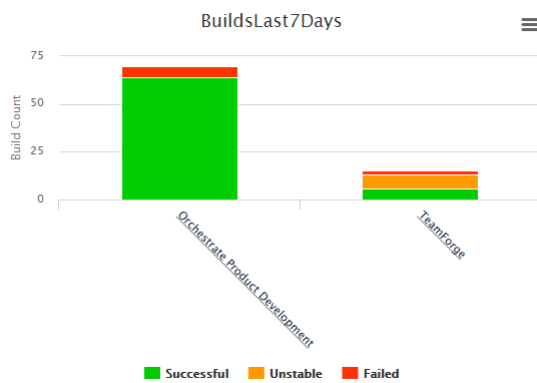
You can drill down this report.

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Build Activity by Project** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more projects from the **PROJECT(S)** drop-down list.
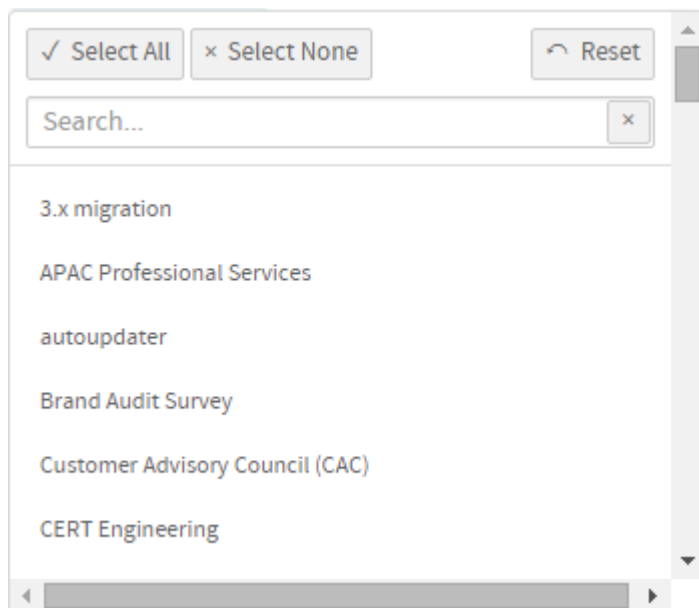
6. Select the number of days from the **LAST N DAYS** drop-down list.
7. The default **Display Type** is **EventsDrilldown**.
8. Select report visibility: Public or Private.
9. Click **Preview**.
10. Click **Create**.
    The report is created and the **View Report** page appears.

   Print or download charts

   You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick
   function icon.

Print chart
Download PNG image
Download JPEG image
Download PDF document
Download SVG vector image

11. Click **Back to Reports List** to go back to the Reports dashboard.

Build and Test Activity

Shows the total number of CI tests—including passed, failed, and ignored—performed for each build over a period of time. Also shows the total test duration for each build. This report is powered by data from the Orchestrate's event data store.

1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page.
The **Select Report Type** page appears.

3. Select **Build and Test Activity** from **Activity Reports**.

4. Type a report title and description.

5. Select one of the projects from the **PROJECT(S)** drop-down list.

6. Select the number of days from the **LAST N DAYS** drop-down list.

7. The default **Display Type** is **EventsDual**.

8. Select report visibility: Public or Private.

9. Click **Preview**.

10. Click **Create**.
The report is created and the **View Report** page appears.

Print or download charts

You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

11. Click **Back to Reports List** to go back to the Reports dashboard.

Test Activity

Shows the count of tests over a period of time including the average build duration. This report is powered by data from the Orchestrate's event data store.
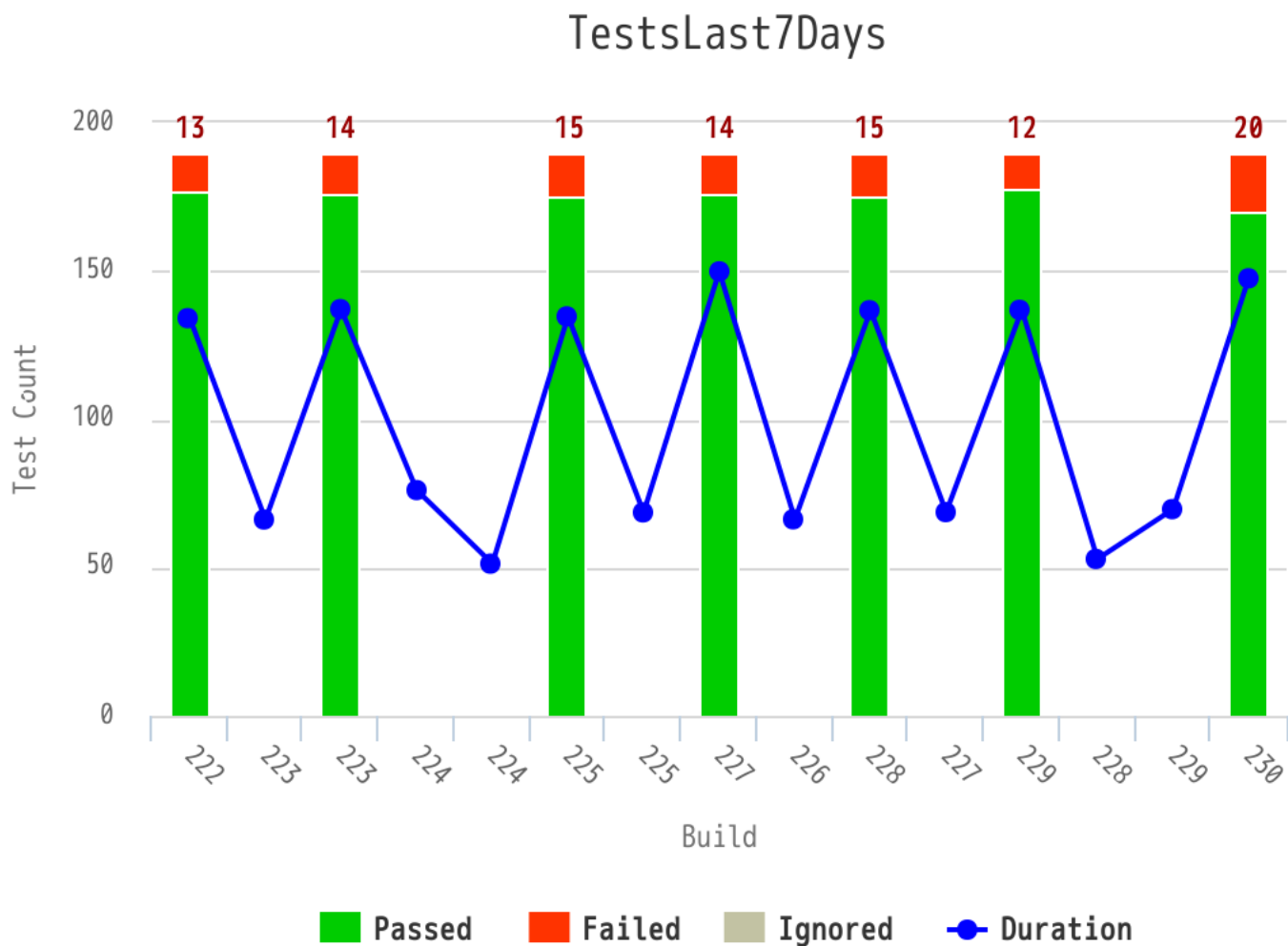
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.

3. Select **Test Activity** from **Activity Reports**.

4. Type a report title and description.

5. Select one or more projects from the **PROJECT(S)** drop-down list.

6. Select the number of days from the **LAST N DAYS** drop-down list.

7. The default **Display Type** is **EventsDual**.

8. Select report visibility: Public or Private.

9. Click **Preview**.

10. Click **Create**.

   The report is created and the **View Report** page appears.

   Print or download charts

   You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.
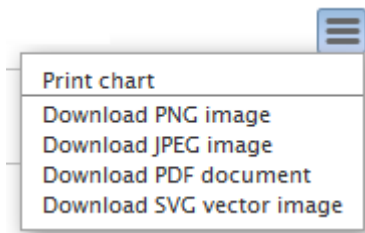
11. Click **Back to Reports List** to go back to the Reports dashboard.

Commit Activity

   Shows the count of daily commits over a period of time. You can group this report by the review status, if required.

   This report is powered by data from the Orchestrate's event data store.
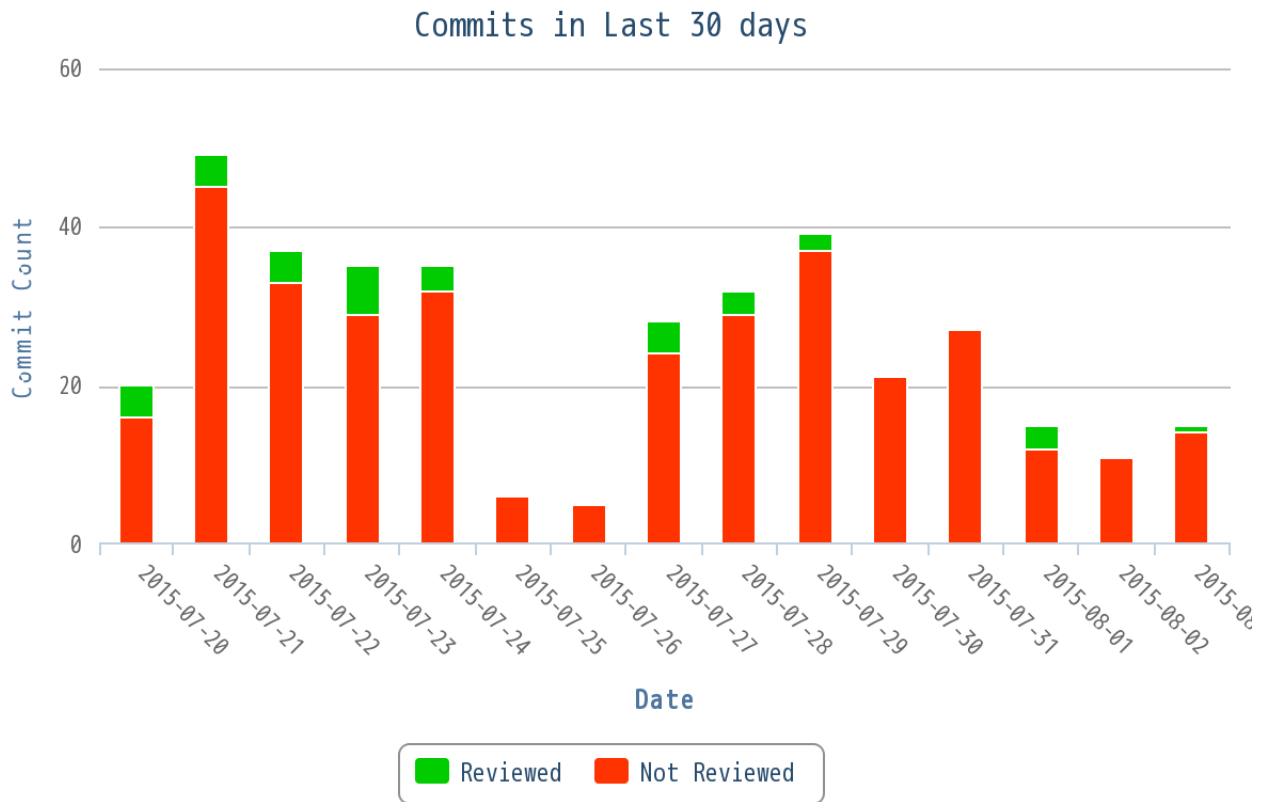
1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Commit Activity** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more projects from the **PROJECT(S)** drop-down list.
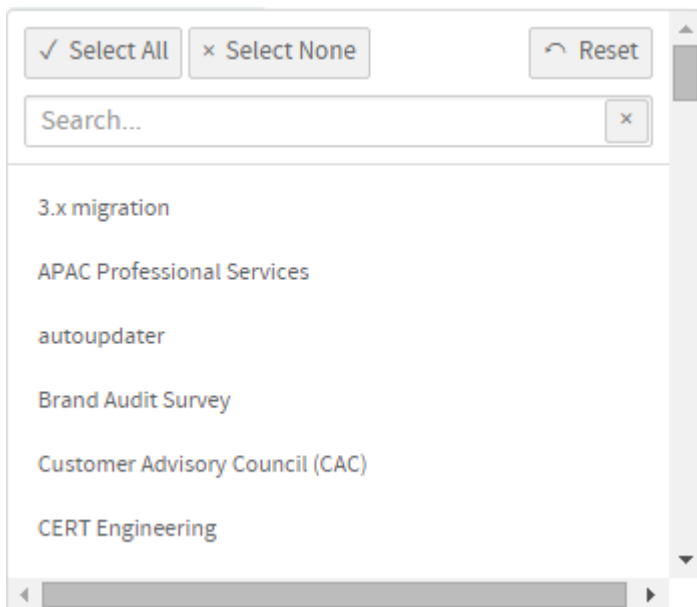
6. Select the number of days from the **LAST N DAYS** drop-down list.

7. By default, the **GROUP BY: Review Status** check box is selected. Clear this check box if you do not want to group the report by review status.

8. The default **Display Type** is **EventsDual**.

9. Select report visibility: Public or Private.

10. Click **Preview**.

11. Click **Create**.
   The report is created and the **View Report** page appears.

   Print or download charts

   You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.
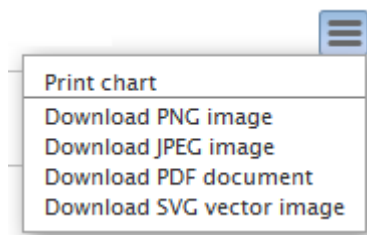
12. Click **Back to Reports List** to go back to the Reports dashboard.

Commits by User

Shows the number of commits by each user over a period of time. This report is powered by data from the Orchestrate's event data store.

1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.

3. Select **Commits by User** from **Activity Reports**.

4. Type a report title and description.

5. Select one or more projects from the **PROJECT(S)** drop-down list.
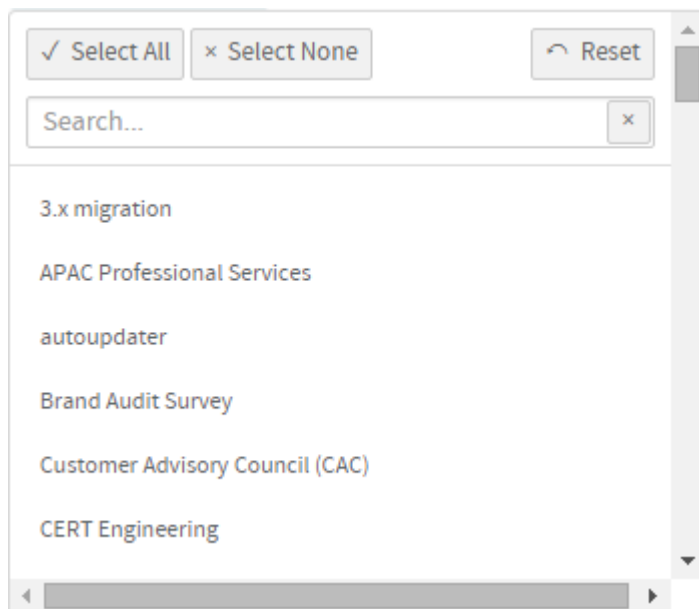
6. Select the number of days from the **LAST N DAYS** drop-down list.

7. By default, the **GROUP BY: Review Status** check box is selected. Clear this check box if you do not want to group the report by review status.

8. **EXCLUDE USERS (COMMA-SEPARATED LIST)**: Optionally, you can type a list of comma-separated user names that you want to exclude from the report.

9. The default **Display Type** is **EventsDual**.

10. Select report visibility: Public or Private.

11. Click **Preview**.

12. Click **Create**.
    The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.
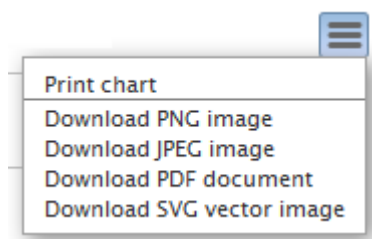
13. Click **Back to Reports List** to go back to the Reports dashboard.

File Changes Over Time
    Shows the file change statistics over time, including the total count of changes and the number of add, delete, copy and modify operations. This report is powered by data from the Orchestrate's event data store.
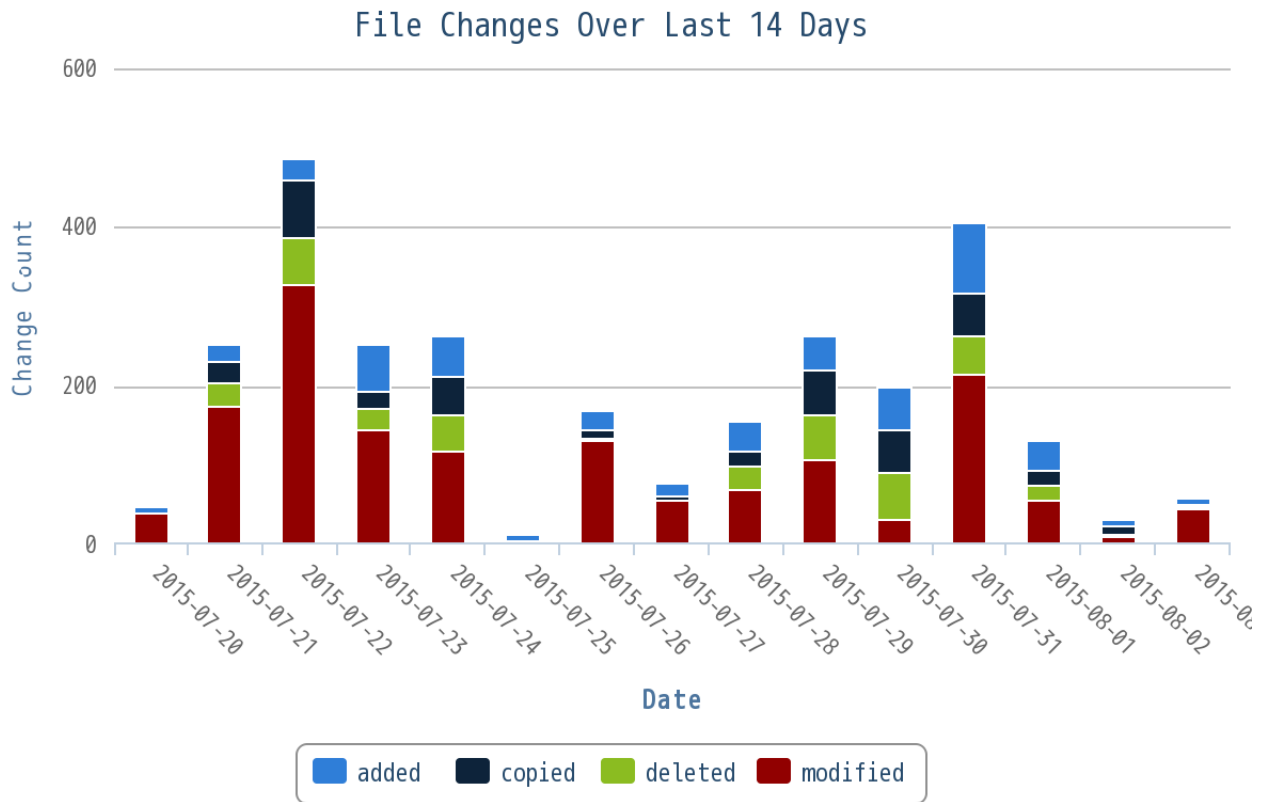
File Changes Over Last 14 Days

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **File Changes Over Time** from **Activity Reports**.
4. Type a report title and description.
5. Select one or more projects from the **PROJECT(S)** drop-down list.

6. Select the number of days from the **LAST N DAYS** drop-down list.

7. The default **Display Type** is **EventsDual**.

8. Select report visibility: Public or Private.

9. Click **Preview**.

10. Click **Create**.
    The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.
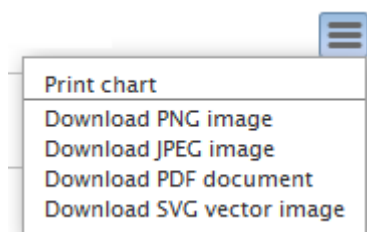
11. Click **Back to Reports List** to go back to the Reports dashboard.

Agile reports

Here's a list of TeamForge agile reports such as the release burn up and burn down reports.

Cumulative Flow Chart

The cumulative flow chart shows the progress of backlog items by status for a sprint or a release.
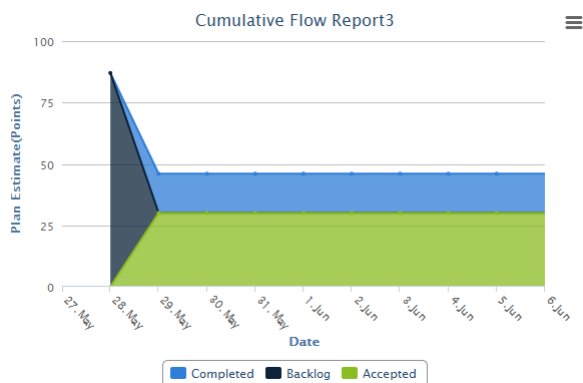
- Generate this report to see the rolled up status chart of scheduled work items in a release or a sprint.
- By viewing the rolled up status of backlog items by date, you can forecast whether you are on track or not, adjust the scope if required and identify bottlenecks in your release or sprint.
- You have 'Date' in the X axis and 'Plan Estimate' in terms of number of points in the Y axis.

1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.

3. Select **Cumulative Flow Chart** from **Agile Reports**.

4. Type a report title and description.

5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.
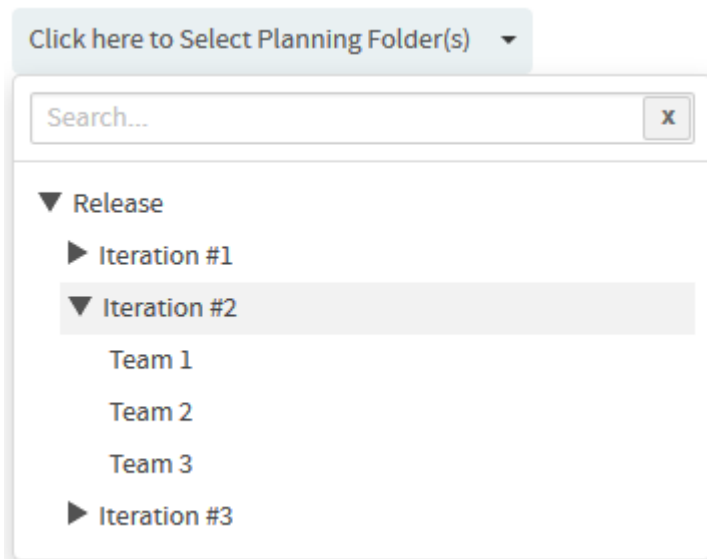
6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.
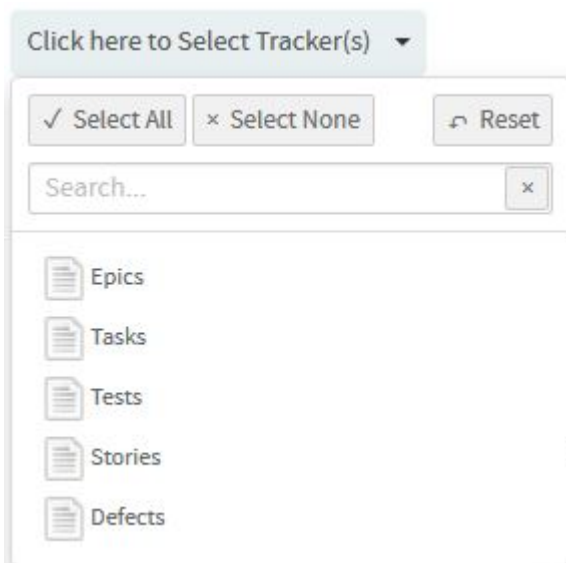
7. Leave the **CHART DISPLAY TYPE** as Area, which is the only available chart type for this report.

8. Select report visibility: Public or Private.

9. Click **Preview**.

10. Click **Create**.

The report is created and the **View Report** page appears.

Print or download charts

You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

Print chart
Download PNG image
Download JPEG image
Download PDF document
Download SVG vector image
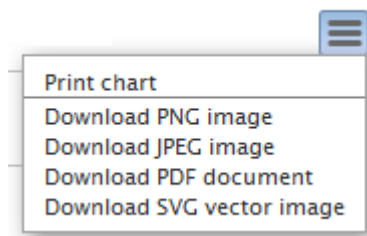
11. Click **Back to Reports List** to go back to the Reports dashboard.

Release Burn Up Chart

The Release Burn Up Chart shows the work progress to date against the total planned work. This chart shows the total planned work, total work completed to date and the rate of progress (velocity).

Release Burn-Up report
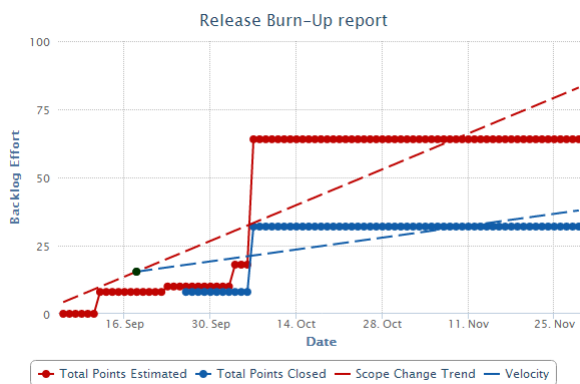
☞ **Important:** You must have the planning folder's start and end dates defined to generate this chart.

This chart includes a trend line to show the planned work scope change between a release's start and end dates (in other words, the start and end dates of the release's planning folder). Optionally, you can also include a trend line in the chart that forecasts when planned work might be completed depending on the average rate of progress (velocity).

1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page.
The **Select Report Type** page appears.

3. Select **Release Burn Up Chart** from **Agile Reports**.

4. Type a report title and description.

5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.
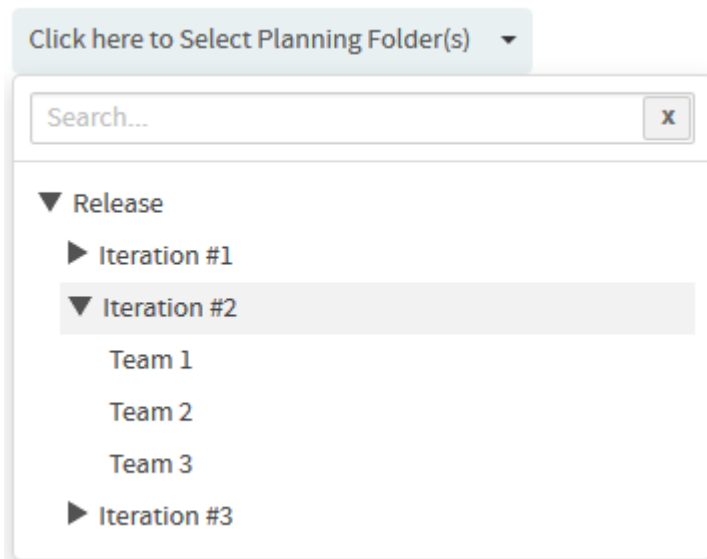
6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.
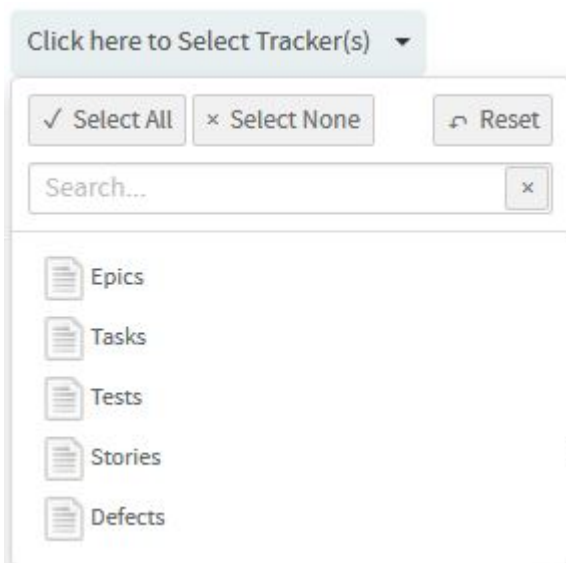
7. Optionally, select the **Include Forecast** check box.

8. Select either Points or Hours.

9. Leave the **CHART DISPLAY TYPE** as Trendlines, which is the only available chart type for this report.

10. Select report visibility: Public or Private.

11. Click **Preview**.

12. Click **Create**.

The report is created and the **View Report** page appears.

Print or download charts

You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

Print chart
Download PNG image
Download JPEG image
Download PDF document
Download SVG vector image
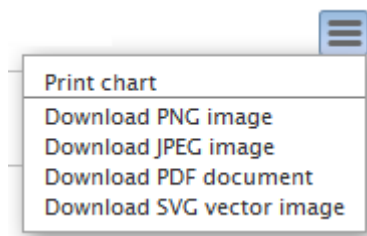
13. Click **Back to Reports List** to go back to the Reports dashboard.

Burn Down Chart

The Burn Down Chart shows your project's progress on a daily basis.

As work gets completed sprint-by-sprint for a release, your backlog tends to decrease. The burn down chart tells the story of how much work is left to be done versus how much time is left. With time along the X axis, the amount of work (backlog) measured in story points is on the Y axis. You can also see the average velocity per sprint in the release burn down chart. In addition to creating burn down charts by release, you can configure this report for a selected sprint planning folder to see its progress on a day-to-day basis.

- The story told in your burn down chart is only as reliable as the underlying data.
- You must have the planning folder's start and end dates defined to generate this chart.
- You can have the **POINTS** field enabled or disabled for a tracker. If a release planning folder consists of both "points-enabled" and "points-disabled" trackers, the release burn down chart shows grey-colored bars for all sprints, meaning, there are work items without points estimation data. This is because, the "points-disabled" tracker work items are considered as unestimated backlog items as they don't have points data. As a workaround, while configuring the release burn down chart, you can include "points-enabled" trackers alone to have the release burn down chart behave as expected.

Things to consider if you are generating burn down chart for a release planning folder
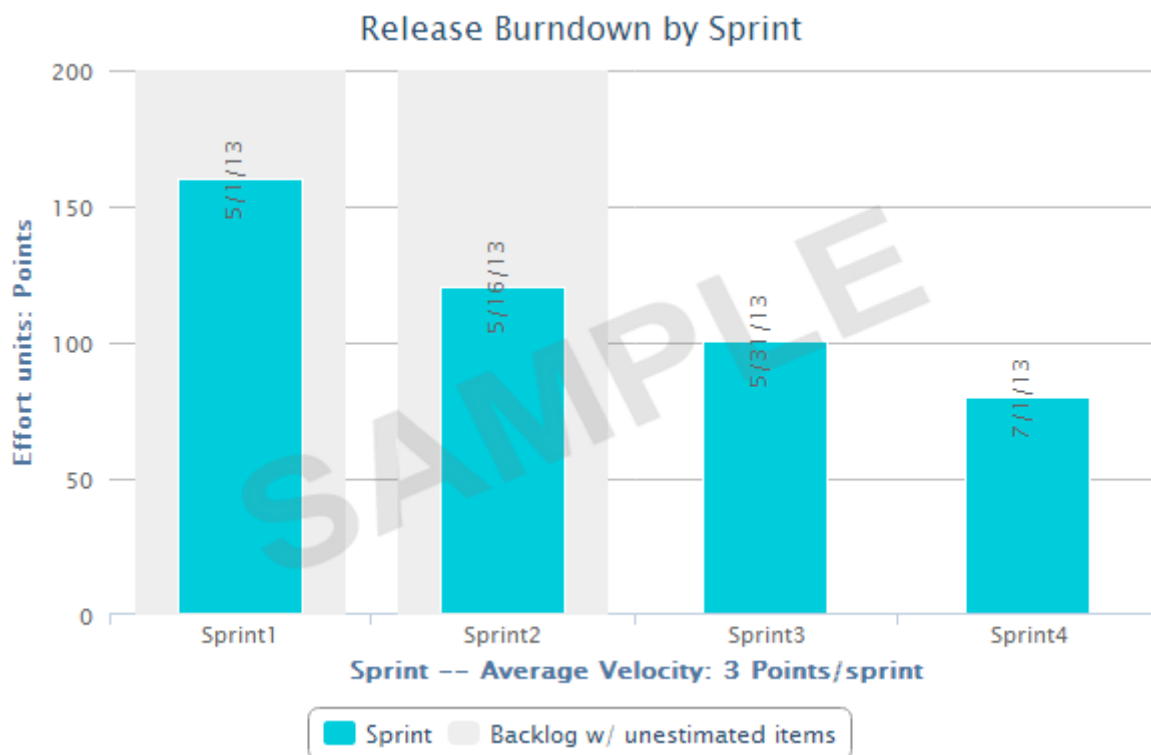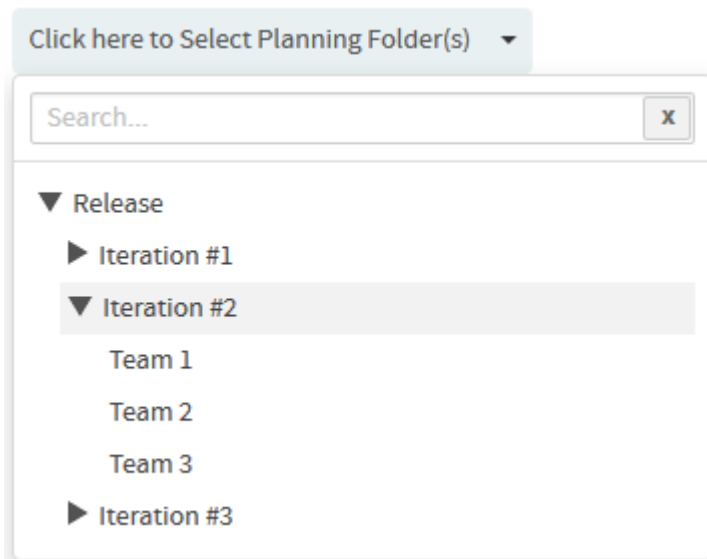
- It is recommended to have all your sprint planning folders organized hierarchically within your release planning folder. Release burn down chart can be generated only if that planning folder has child (sprint) planning folders.

- Within a selected release planning folder, all immediate child planning folders are considered as sprint planning folders.
- The bar chart shows the work left to be done (estimated as points) on the first day of every sprint.
- The release burn down calculation is based on (story) points and no effort data is considered. Meaning, the release burn down will not behave as expected if you have effort data alone and no points data.

Things to consider if you are generating burn down chart for a sprint planning folder

Sprint Burn down chart
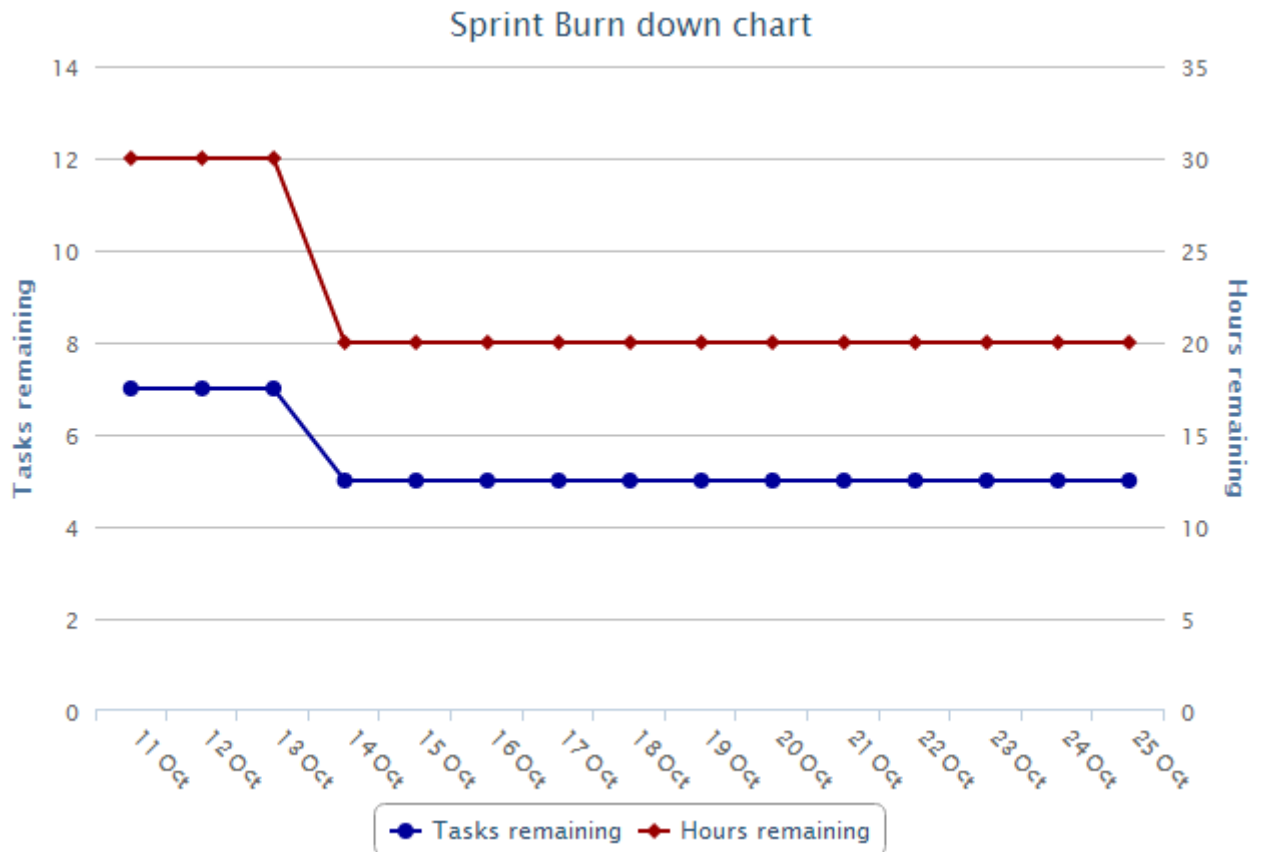
- The sprint burn down calculation is based on remaining effort data and no (story) points data is considered. Meaning, the sprint burn down will not behave as expected if you have points data alone and no remaining effort data.
- You can exclude weekends or specific days from your report if you are creating a burn down chart for a sprint planning folder.
- The sprint burn down chart shows the number of tasks and hours remaining for a selected sprint planning folder.

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Burn Down Chart** from **Agile Reports**.
4. Type a report title and description.
5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.

6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.
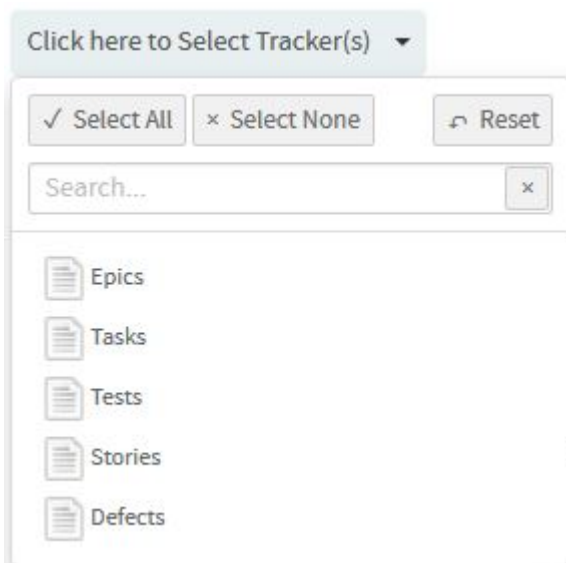
7. Select one of the options, **Release** or **Sprint**, to create burn down chart for either a release planning folder or a sprint planning folder.

☞ **Tip:** Select the Exclude **Weekends** check box to exclude weekends from the report.

8. Leave the **CHART DISPLAY TYPE** as Burndown, which is the only available chart type for this report.

9. Select report visibility: Public or Private.

10. Click **Preview**.

11. Click **Create**.

The report is created and the **View Report** page appears.

Print or download charts

You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

Print chart
Download PNG image
Download JPEG image
Download PDF document
Download SVG vector image
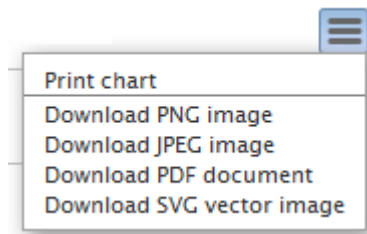
12. Click **Back to Reports List** to go back to the Reports dashboard.
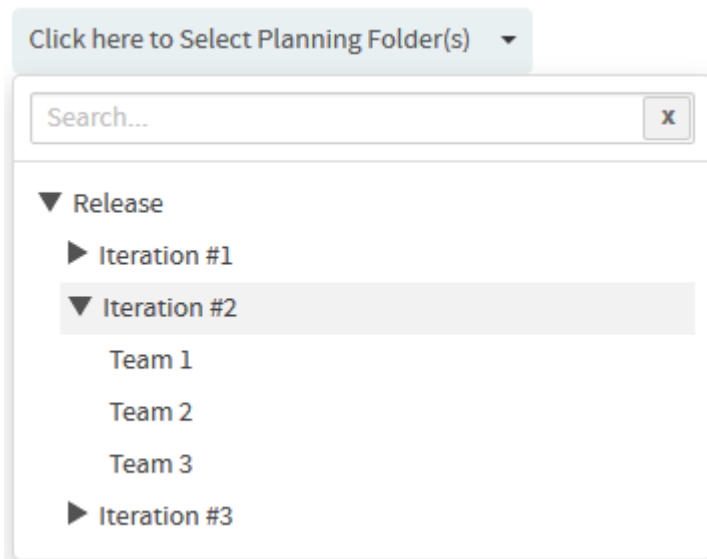
Committed vs Done vs Missed
The Committed vs Done vs Missed chart shows a comparison of the amount of work (in terms of story points) committed, completed and missed for a sprint.
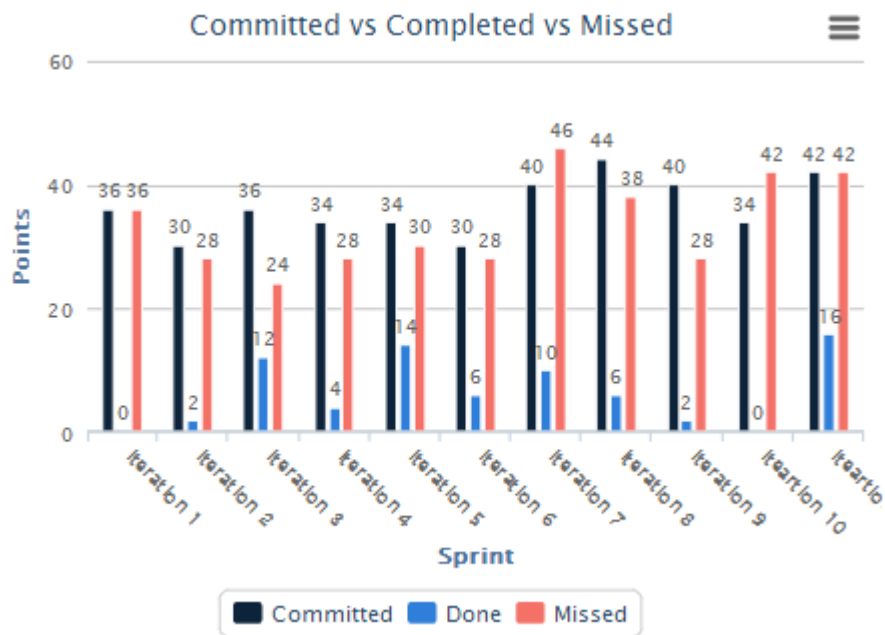
1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Committed vs Done vs Missed** from **Agile Reports**.
4. Type a report title and description.
5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.

6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.
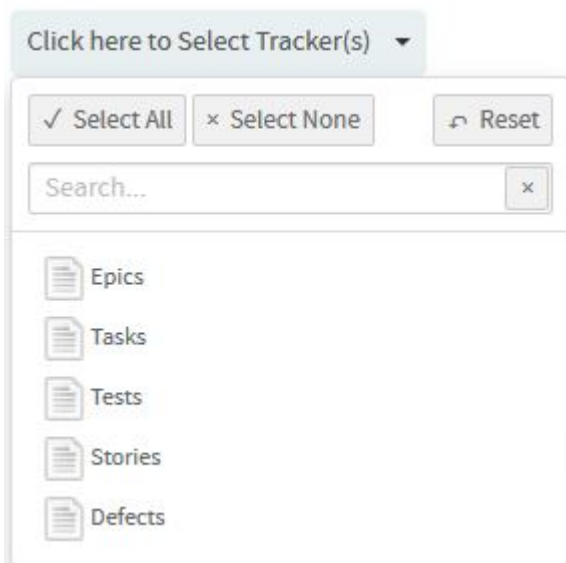
7. Leave the **CHART DISPLAY TYPE** as Columndone, which is the only available chart type for this report.
8. Select report visibility: Public or Private.
9. Click **Preview**.
10. Click **Create**.
   The report is created and the **View Report** page appears.

   Print or download charts

   You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

```
☰
Print chart
Download PNG image
Download JPEG image
Download PDF document
Download SVG vector image
```
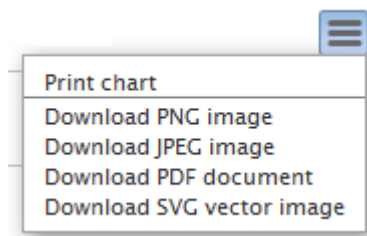
11. Click **Back to Reports List** to go back to the Reports dashboard.

Distribution reports

Here's a list of reports to see distribution of work items such as tracker artifacts by parameters such as status, size, effort and so on.
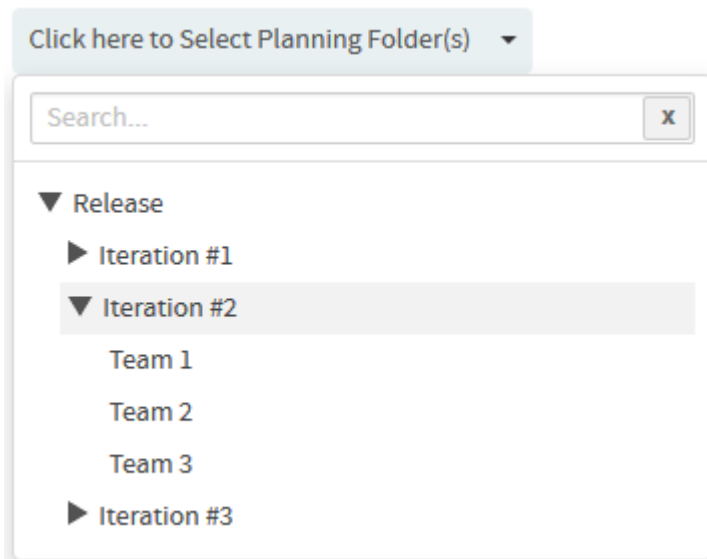
Status Distribution by Area/Group

You can generate this chart for a specific sprint or release planning folder (optionally you can include child planning folders, if any) and take stock of the number of artifacts in different statuses and have them grouped by Group, Category, Customer, Assigned To, Priority or Teams.
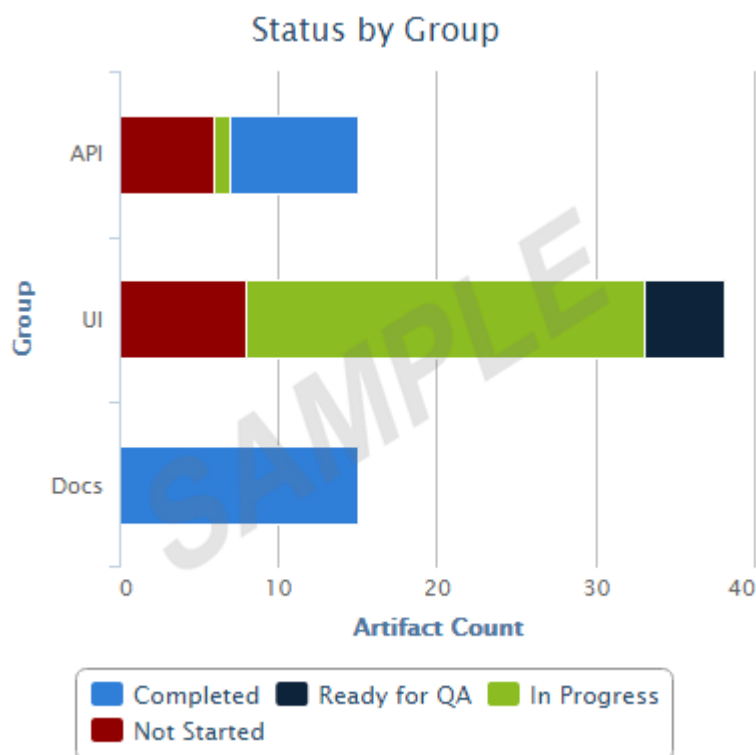
☞ **Note:** Unlike other reports (that use datamart), this report runs on your operational database.

1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page.
 The **Select Report Type** page appears.

3. Select **Status Distribution by Area/Group** from **Distribution Reports**.

4. Type a report title and description.

5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.

6. Optionally, select the **Include: Child planning folders** check box to have the child planning folders, if any, included.

7. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.
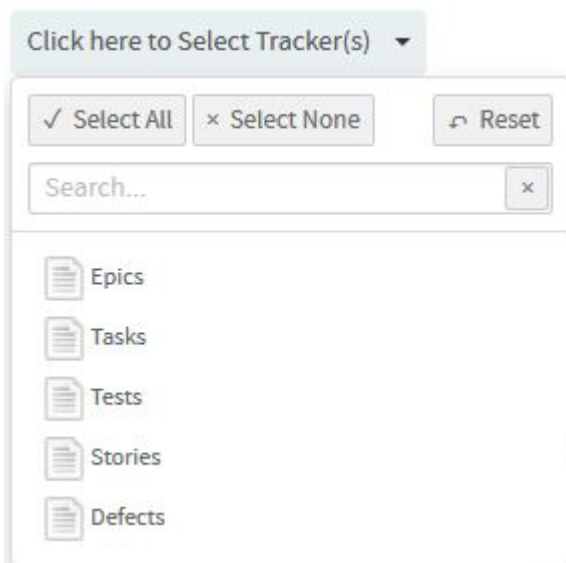
8. Select an option from the **GROUP BY** drop-down list to have the chart grouped by Group, Category, Customer, Assigned To, Priority or Teams.

9. Leave the **CHART DISPLAY TYPE** as Bar, which is the only available chart type for this report.

10. Select report visibility: Public or Private.

11. Click **Preview**.

12. Click **Create**.
    The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.
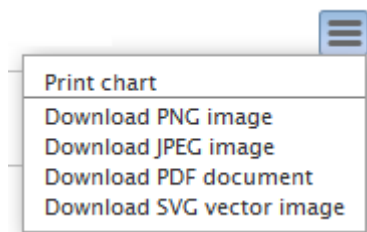
13. Click **Back to Reports List** to go back to the Reports dashboard.

Total Size by Area/Group

You can generate this chart for a specific sprint or release planning folder (optionally you can include child planning folders, if any) and take stock of the total size (as in total of story points or estimated, actual or remaining effort) of artifacts and have them grouped by Group, Category, Customer, Assigned To, Priority, Tracker or Teams. You can generate this report for all artifacts or just for open or closed artifacts.
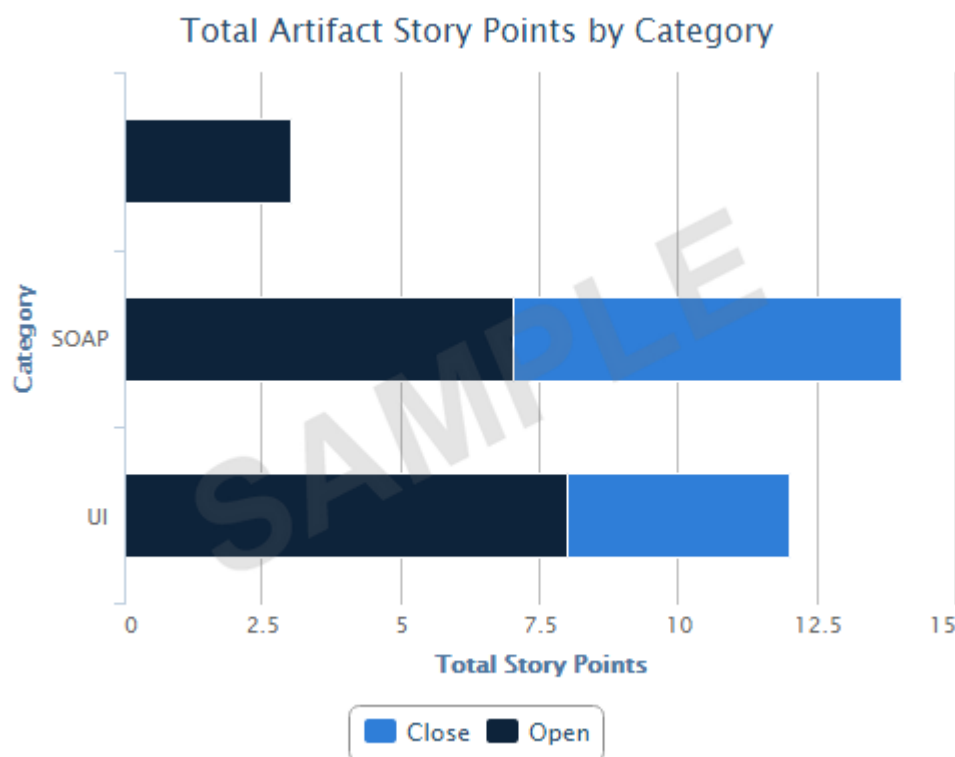
☛ **Note:** Unlike other reports (that use datamart), this report runs on your operational database.

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Total Size by Area/Group** from **Distribution Reports**.
4. Type a report title and description.
5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.
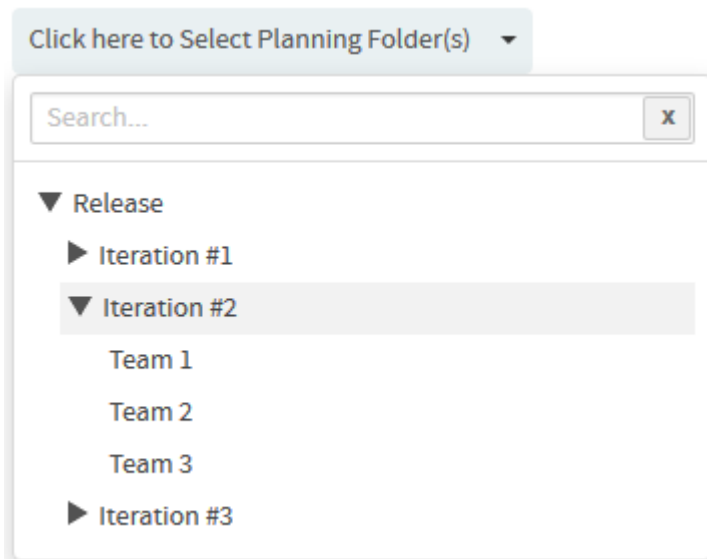
6. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.

7. Optionally, select the **Include: Child planning folders** check box to have the child planning folders, if any, included.

8. Select what you want to total from the **TOTAL BY** drop-down list. You can total by Story Points or Estimated Effort or Actual Effort or Remaining Effort.

9. Select an option from the **GROUP BY** drop-down list to have the chart grouped by Group, Category, Customer, Assigned To, Priority, Tracker or Teams.

10. Select one of the options, Include artifacts: **All** (default) or **Open** or **Closed**, to include all artifacts or just open or closed artifacts respectively.

11. Leave the **CHART DISPLAY TYPE** as Bar, which is the only available chart type for this report.

12. Select report visibility: Public or Private.

13. Click **Preview**.

14. Click **Create**.

The report is created and the **View Report** page appears.

Print or download charts

You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

```
☰

Print chart
Download PNG image
Download JPEG image
Download PDF document
Download SVG vector image
```

15. Click **Back to Reports List** to go back to the Reports dashboard.

Average Size by Area/Group

You can generate this chart for a specific sprint or release planning folder (optionally you can include child planning folders, if any) and take stock of the average size (as in average of story points or estimated, actual or remaining effort) of artifacts and have them grouped by Group, Category, Customer, Assigned To, Priority, Tracker or Teams. You can generate this report for all artifacts or just for open or closed artifacts.

Average Artifact Story Points by Category
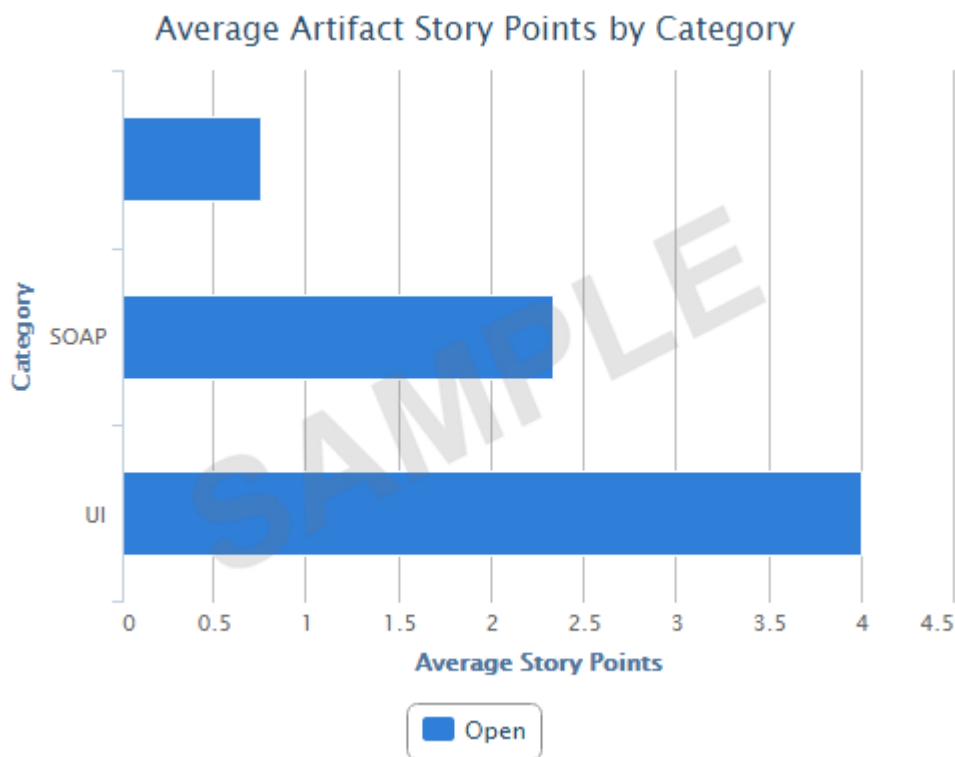
👉 **Note:** Unlike other reports (that use datamart), this report runs on your operational database.

1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page.
The **Select Report Type** page appears.

3. Select **Average Size by Area/Group** from **Distribution Reports**.

4. Type a report title and description.

5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.

6. Optionally, select the **Include: Child planning folders** check box to have the child planning folders, if any, included.

7. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.
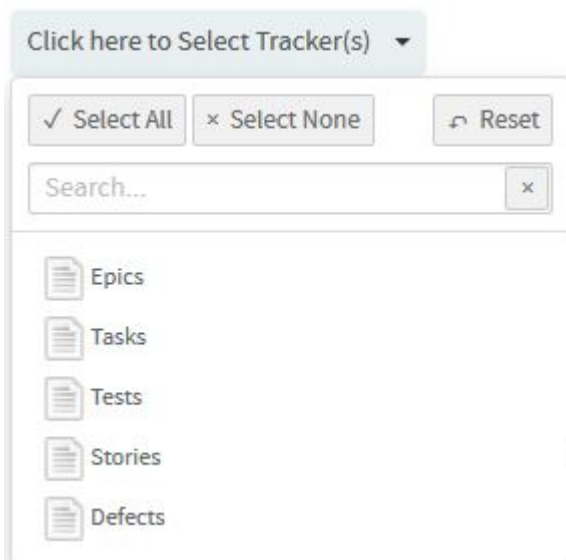
8. Select what you want to average from the **AVERAGE BY** drop-down list. You can average by Story Points or Estimated Effort or Actual Effort or Remaining Effort.

9. Select an option from the **GROUP BY** drop-down list to have the chart grouped by Group, Category, Customer, Assigned To, Priority, Tracker or Teams.

10. Select one of the options, Include artifacts: **All** (default) or **Open** or **Closed**, to include all artifacts or just open or closed artifacts respectively.

11. Leave the **CHART DISPLAY TYPE** as Bar, which is the only available chart type for this report.

12. Select report visibility: Public or Private.

13. Click **Preview**.

14. Click **Create**.

The report is created and the **View Report** page appears.

Print or download charts

You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

```
☰
Print chart
Download PNG image
Download JPEG image
Download PDF document
Download SVG vector image
```

15. Click **Back to Reports List** to go back to the Reports dashboard.

Total Size by Tracker Type

You can generate this chart for a specific planning folder (optionally you can include child planning folders, if any) and take stock of the total size of artifacts belonging to one or more tracker types and have them aggregated by Story Points or Actual Effort or Estimated Effort or Remaining Effort. This report is more useful if you want to know the total size by tracker types for a release and so is typically run against a specific release planning folder. You can generate this report for all artifacts or just for open or closed artifacts.
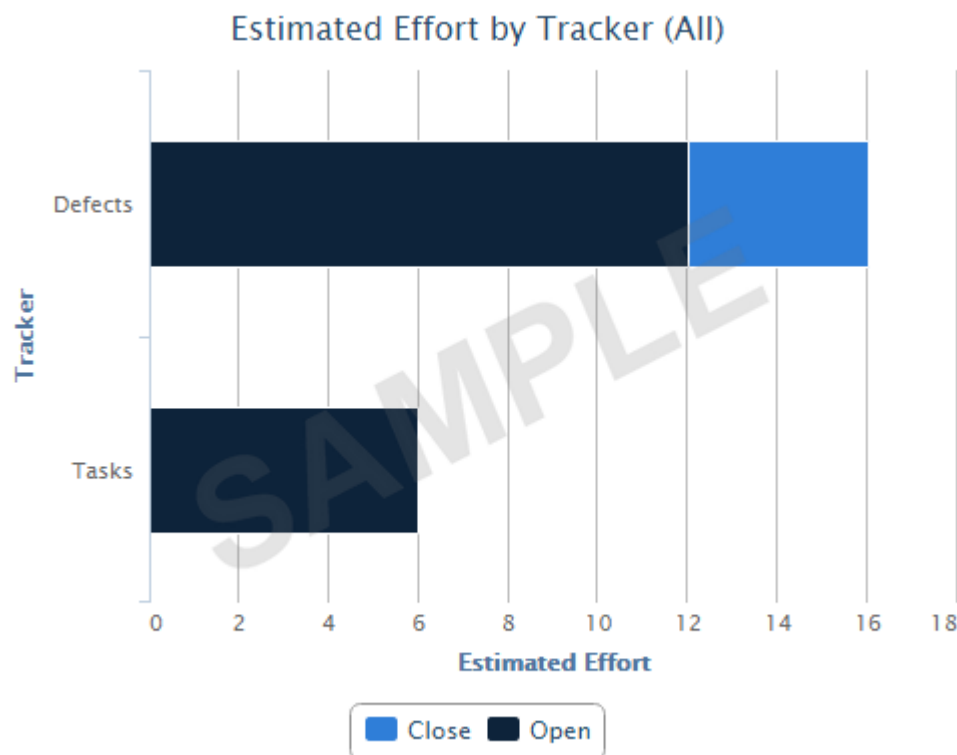
Note: Unlike other reports (that use datamart), this report runs on your operational database.

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Total Size by Tracker Type** from **Distribution Reports**.
4. Type a report title and description.
5. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.

6. Optionally, select the **Include: Child planning folders** check box to have the child planning folders, if any, included.

7. Select what you want to aggregate from the **AGGREGATE BY** drop-down list. You can aggregate by Story Points or Estimated Effort or Actual Effort or Remaining Effort.

8. Select one of the options, Include artifacts: **All** (default) or **Open** or **Closed**, to include all artifacts or just open or closed artifacts respectively.

9. Leave the **CHART DISPLAY TYPE** as Bar, which is the only available chart type for this report.

10. Select report visibility: Public or Private.

11. Click **Preview**.

12. Click **Create**.
   The report is created and the **View Report** page appears.

   Print or download charts

   You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.
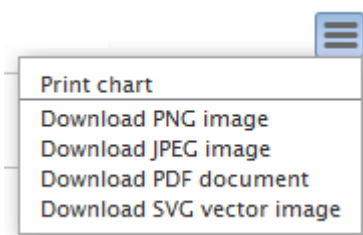
13. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Distribution Chart (Multiple Trackers)
   Using this chart, you can report the status of artifacts belonging to more than one tracker or planning folder. You can also create charts based on a combination of more than one tracker and planning folder.

Open 7.x Artifacts by Tracker Name
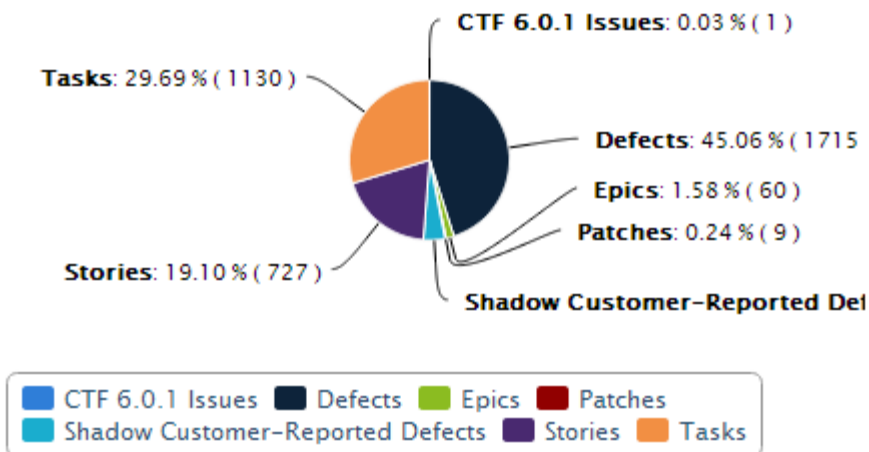
CTF 6.0.1 Issues: 0.03 % (1)

Tasks: 29.69 % (1130)

Defects: 45.06 % (1715

Epics: 1.58 % (60)

Patches: 0.24 % (9)

Stories: 19.10 % (727)

Shadow Customer-Reported Def

CTF 6.0.1 Issues — Defects — Epics — Patches — Shadow Customer-Reported Defects — Stories — Tasks

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Artifact Distribution Chart (Multiple Trackers)** from **Distribution Reports**.
4. Type a report title and description.
5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.

Click here to Select Tracker(s)

✓ Select All   × Select None   ↺ Reset

Search...   ×

Epics

Tasks

Tests

Stories

Defects

6. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.

7. Select one of the options from the **DISTRIBUTE BY** drop-down list.

8. Select one of the options from the **OPEN VS CLOSE** drop-down list.

9. Leave the **CHART DISPLAY TYPE** as Pie, which is the only available chart type for this report.

10. Select report visibility: Public or Private.

11. Click **Preview**.

12. Click **Create**.

    The report is created and the **View Report** page appears.
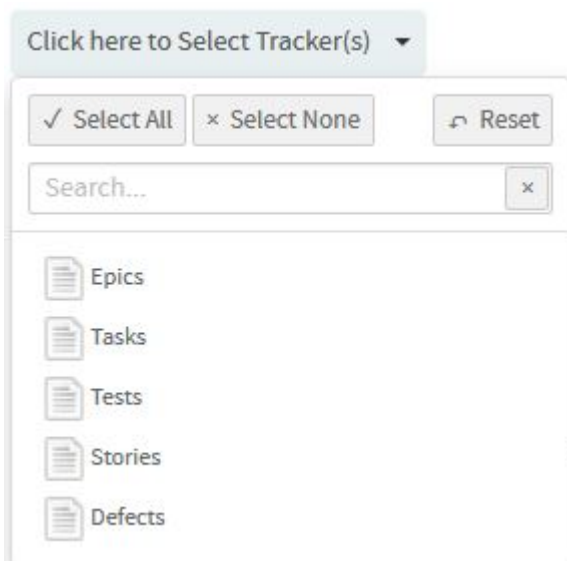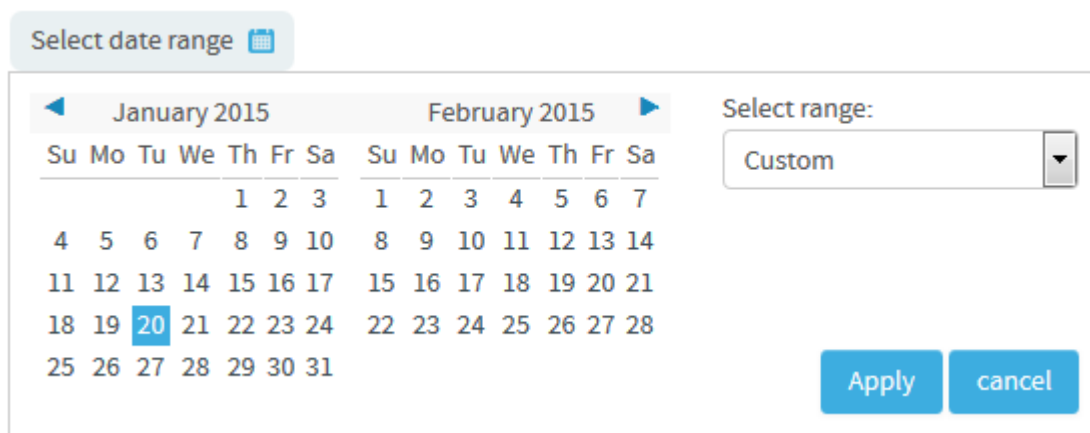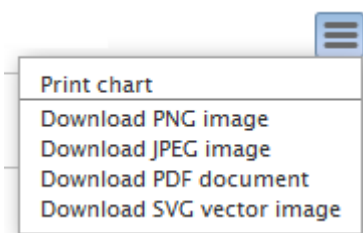
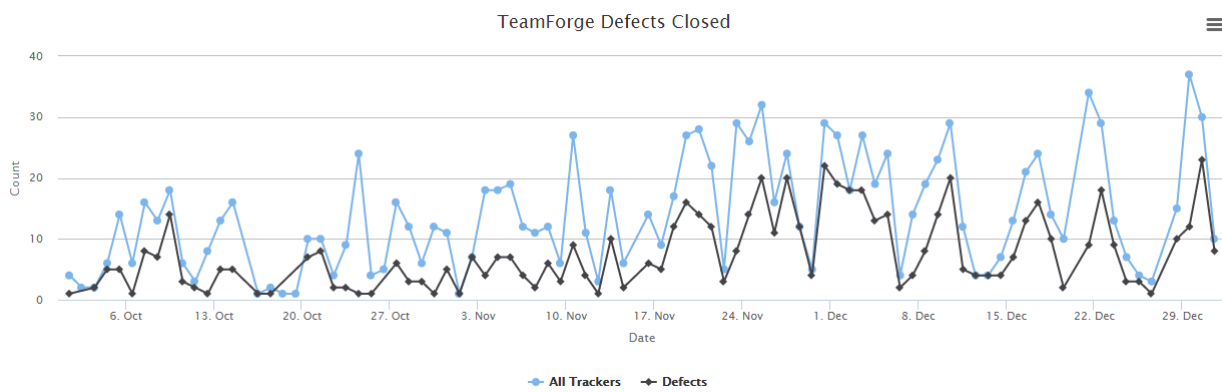   **Print or download charts**

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

13. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Distribution Chart (Single Tracker)

    This chart comes in handy if you want to report the status of artifacts belonging to a specific tracker.
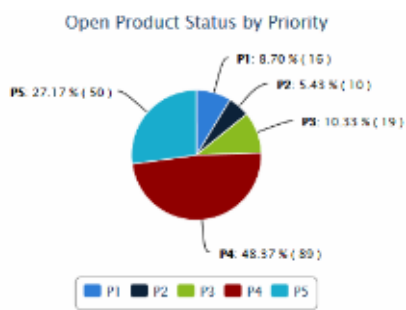
1. Click **REPORTS** from the **Project Home** menu.

2. Click **Create** in the **List Reports** page.
    The **Select Report Type** page appears.

3. Select **Artifact Distribution Chart (Single Tracker)** from **Distribution Reports**.

4. Type a report title and description.

5. Select a tracker from the **SELECT TRACKER(S)** drop-down list.

6. Select one or more planning folders (select the check boxes) from the **SELECT PLANNING FOLDER(S)** drop-down list.

7. Select one of the options from the **DISTRIBUTE BY** drop-down list.

8. Select one of the options from the **OPEN VS CLOSE** drop-down list.

9. Leave the **CHART DISPLAY TYPE** as Pie, which is the only available chart type for this report.

10. Select report visibility: Public or Private.

11. Click **Preview**.

12. Click **Create**.
    The report is created and the **View Report** page appears.

Print or download charts

You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.

13. Click **Back to Reports List** to go back to the Reports dashboard.

Trend reports
Here's a list of reports to know some artifact trend data.

Average Age Report
The average age chart lets you know the average age of artifacts in one or more trackers you select.
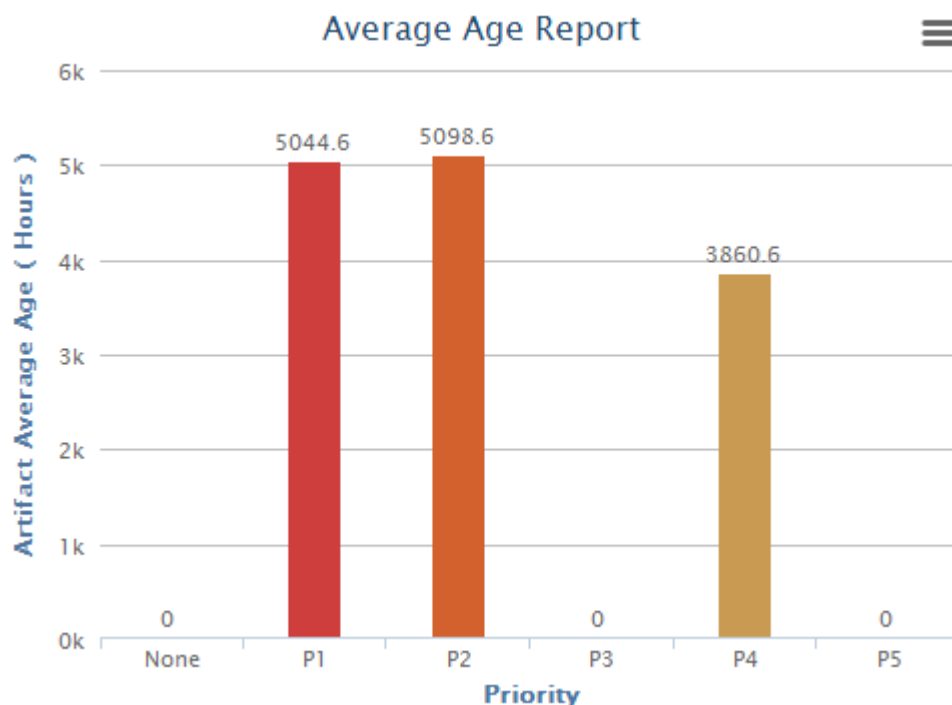
You can:

- Generate this report to know the average age of artifacts in various units of time such as number of hours, days or weeks.
- Group this report either by artifact priority such as P1, P2 and so on or by both artifact priority and category.
- Generate this report for either open or closed artifacts.
- Exclude weekends from being included in the average age calculation.

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Average Age Report** from **Trend Reports**.
4. Type a report title and description.
5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.

6. Select a planning folder from the **SELECT PLANNING FOLDER(S)** drop-down list.
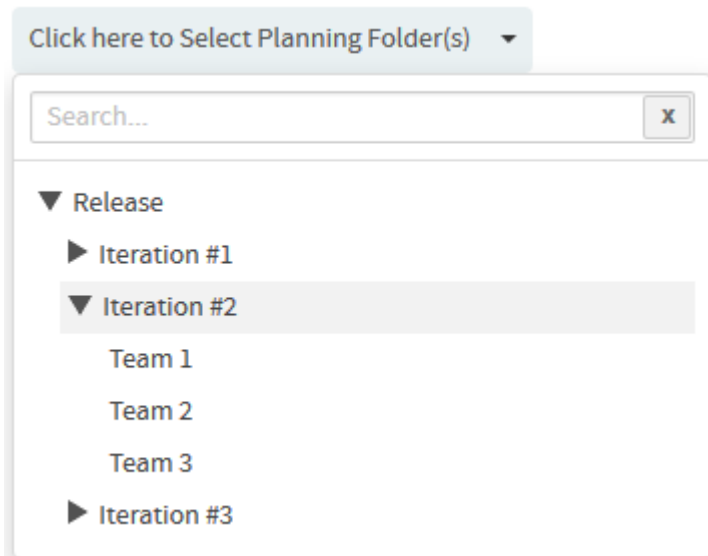
7. Select one of the time units such as hours, days or weeks from the **AVG. TIME IN** drop-down list.
8. Select either **Priority** or **Priority & Category** from the **GROUP BY** drop-down list.
9. If required, select **EXCLUDE Weekends** check box to exclude weekends from being included in the average age calculation.
10. Select either Open Only or Closed from the **OPEN VS. CLOSE** drop-down list.
11. Leave the **CHART DISPLAY TYPE** as Stackedcolumns, which is the only available chart type for this report.
12. Select report visibility: Public or Private.
13. Click **Preview**.
14. Click **Create**.

    The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.
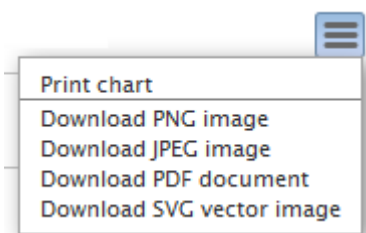
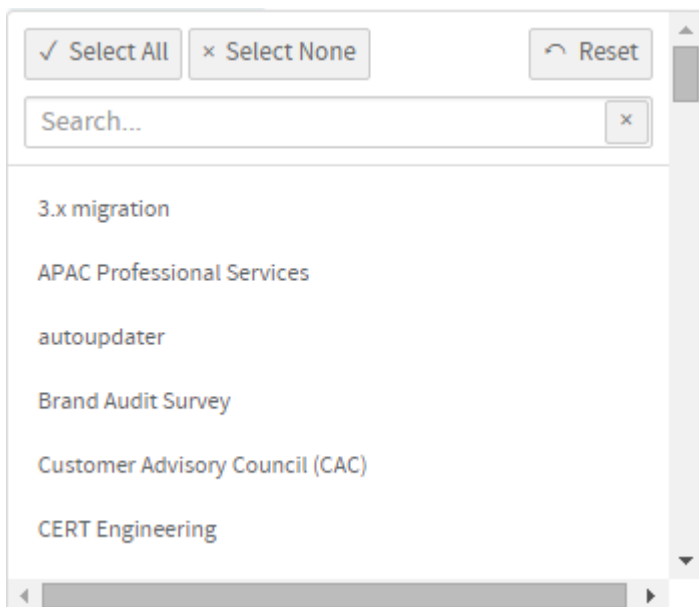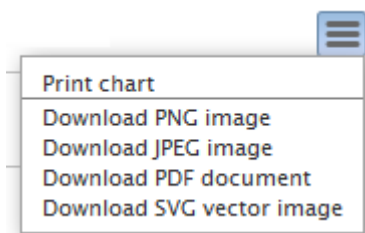15. Click **Back to Reports List** to go back to the Reports dashboard.

Artifact Open/Close Chart (Multiple Trackers)

Use this chart to know the number of opened and closed artifacts over a period of time across one or more trackers or planning folders. This chart also helps in knowing the number of associated commits in specific repositories over a period of time. You can also know the total number of open artifacts through this chart.
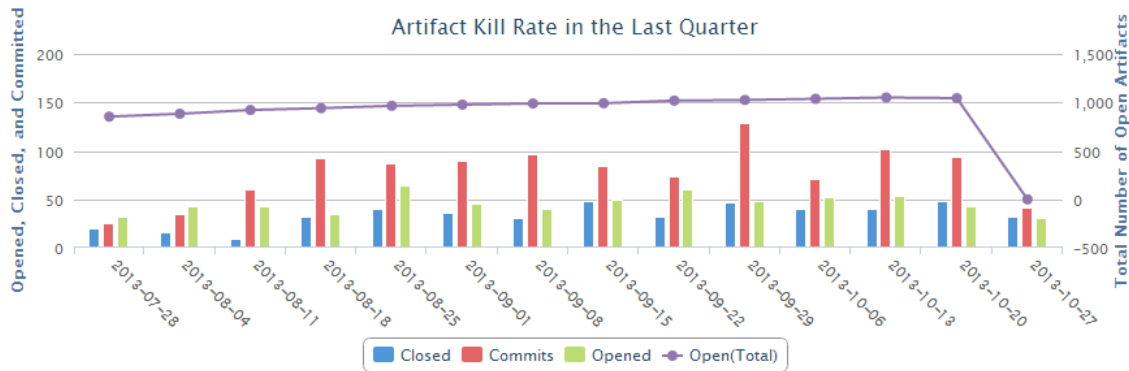
The number of opened and closed artifacts, and the number of associated commits are shown as color-coded bars (scale on left Y coordinate axis) and the total number of open artifacts is shown as a color-coded line (scale on right Y coordinate axis). The X coordinate axis shows the time scale of the chart.

At least one of the reporting parameters such as the **Tracker ID**, **Planning Folder ID** or the **Repository ID** is required to generate this chart. The following table lists the various reporting parameters for Artifact - Open/Close Chart (Multiple Trackers).

1. Click **REPORTS** from the **Project Home** menu.
2. Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.
3. Select **Artifact Open/Close Chart (Multiple Trackers)** from **Trend Reports**.
4. Type a report title and description.
5. Select one or more trackers from the **SELECT TRACKER(S)** drop-down list.

6. Select one or more planning folders (select the check boxes) from the **SELECT PLANNING FOLDER(S)** drop-down list.
7. Select one or more repositories from the **SELECT REPOSITORY(S)** drop-down list.

8. Select one or more artifact priorities.

   For example, select P1 and P2 to include P1 and P2 artifacts in your chart. Select the **None** to include artifacts that have no priority assigned to them.

9. Select one of the report durations: Last Week/Last Month/Last Quarter/Last Year.

10. Leave the **CHART DISPLAY TYPE** as Dualaxeslinecolumn, which is the only available chart type for this report.

11. Select report visibility: Public or Private.

12. Click **Preview**.

13. Click **Create**.

    The report is created and the **View Report** page appears.

    Print or download charts

    You can print charts or download them as .PNG, .JPG, .SVG or .PDF files using the print/download quick function icon.
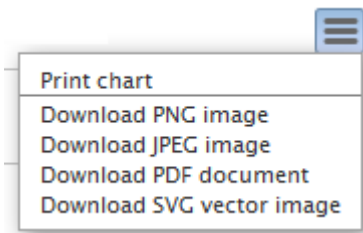
14. Click **Back to Reports List** to go back to the Reports dashboard.

Table reports: Task and Tracker reports

These are reports on trackers and tasks in a tabular format.

- **Task reports**: Task reports display selected summary data about project tasks. You can generate reports on the tasks in a selected project or across multiple projects.
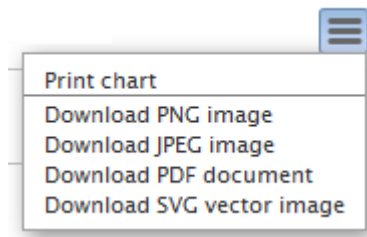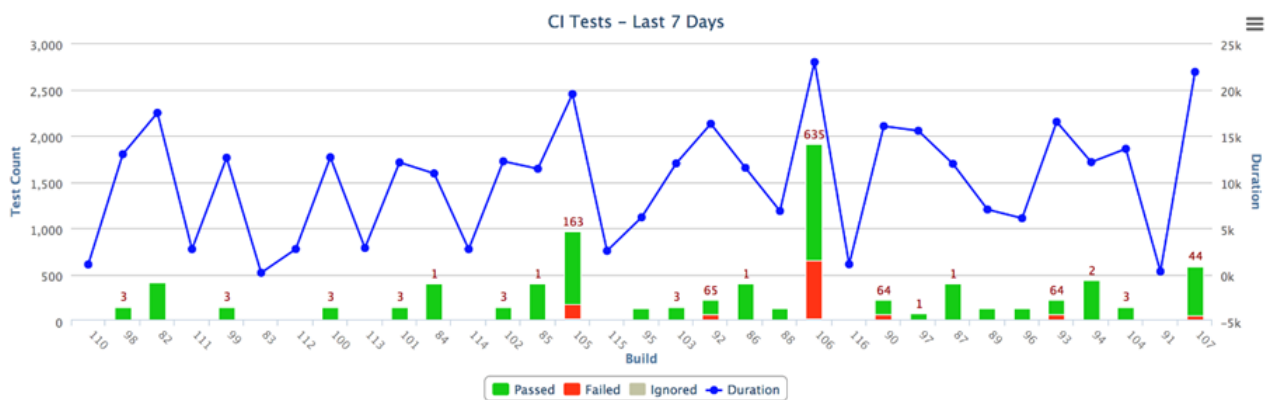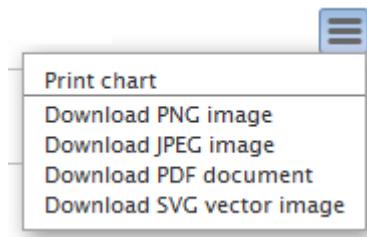
  a) Click **REPORTS** from the **Project Home** menu.

  b) Click **Create** in the **List Reports** page.

     The **Select Report Type** page appears.

  c) Select **Task** from **Table Reports**.

  d) **Title** and **Description**: Type a title and description for your report.

  e) **Project(s)**: Select the project or projects from which you want task data to be reported.

  f) **Status(s)**: Select the status of the tasks to be included in your report.

  g) **Priority(s)**: Select one or more priorities of tasks to be included in your report.

  h) Select one or more **Assigned To** or **Submitted By** project members.

     Click the '+' icon to add members.

ASSIGNED TO:

Select Assigned To

SUBMITTED BY:

Select Submitted By

i) **Start Date** and **End Date**: Select the start and end date ranges of tasks to be included in your report.

j) **Report Field(s)**: Select the fields you want displayed in your report.

k) Select one of the report visibility options: Public (default) or Private.

l) Click **Create**.

The **View Report** page appears. The task report is created.

• **Tracker reports**: Tracker reports give you a summary of the history of tracker artifacts. A tracker report can cover the tracker artifacts in a selected tracker or across all trackers in a project.

a) Click **REPORTS** in the project navigation bar.

b) Click **Create** in the **List Reports** page.
   The **Select Report Type** page appears.

c) Select **Tracker** from **Table Reports**.

   To create a tracker report across multiple projects, click **Show All Projects** and select the trackers from the required projects.

d) **Title** and **Description**: Type a title and description for your report.

e) **Select Tracker(s)**: Select the trackers from which you want tracker artifact data to be reported.

f) **Group By**: Have the report grouped by one of the following: Assigned To, Category, Customer, Group, Priority, Status or Team.

g) **Summary Statistics**: Select one of the following statistics to summarize the report: Count of Artifacts, Sum of Points or Sum of Effort.

h) **Priority(s)**: Select one or more priorities of artifacts to be included in your report

i) **Select Artifact Maturity(s)**: Select one or all of the following values to have all or new or modified artifacts included in the report respectively: Any or New or Edited

j) Select date ranges for create date (**Submitted On**), last edited date (**Last Modified**), or closed date (**Closed**).

k) Select one or more **Assigned To** or **Submitted By** project members.

   Click the '+' icon to add members.

ASSIGNED TO:

Select Assigned To ⊕

SUBMITTED BY:

Select Submitted By ⊕

l) Select a planning folder form the **Select Planning Folder(s)** drop-down list.

m) Select one of the report visibility options: Public (default) or Private.

n) Click **Next**.

A list of filters (such as category, status, reported in release, fixed in release and so on) for the selected trackers are shown. Select the filter values from the drop-down lists and click **Create**. Based on the criteria being selected for filtering, it returns the configured rows.

The **View Report** page appears. The default value of the token *MAX_REPORT_ROWS* is set to '300' in `site-options.conf` file. You can increase or decrease this value as desired. However, if you increase the *MAX_REPORT_ROWS* value, there would be a delay in the generation of the report as it is grouped by one of the values specified in the **Group By** field if selected, every time you run the report. The tracker report is generated based on the criteria:

• If you have selected a single tracker having more than 300 artifacts, the tracker report shows only the first 300 artifacts. To see all the artifacts in that tracker, click the **Export** button.

• If you have selected multiple trackers with more than 300 artifacts in all, then the tracker report shows the artifacts based on the following calculation:

No. of artifacts in report = *MAX_REPORT_ROWS* ÷ No. of selected trackers

For example, if you have selected 3 trackers with more than 300 artifacts in all, the first 100 artifacts is displayed in the tracker report for each tracker. You can export the report for each tracker to view all the artifacts in it.

👉 **Important:**

In a tracker report, when you select more than one value from a multi-select, user-defined field as filter and if an artifact is associated with all of the selected values, then that artifact's record is duplicated for each of the selected values.

For example, assume that you have a 'Select User' multi-select, user-defined field with values 'User 1', 'User 2' and 'User 3' in a tracker report. All these three values are associated with 'artifact 1001'. Select all three values as filter and generate the tracker report. You will see 'artifact 1001' record being duplicated, that is, you will see three individual 'artifact 1001' records created for each of the three users.

Run, edit, export or delete a report

Run a report to get fresh data on the status of your project.

The report data is regenerated each time the report is run. You can also refresh the data by clicking the 'Refresh' icon in the **List Reports** page.
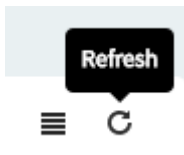
☞ **Tip:** You can modify the report criteria before generating the report data. For example, you might want to run the same report each week, using all of the same report criteria except the start and end dates.

☞ **Note:** Report results contain only those items that you have permission to view. Other project members with different permissions might get different results when running the same report.

1. Click **REPORTS** from the **Project Home** menu.

2. Click the title of the desired report. The report is generated and displayed.

3. To edit a report, click **REPORTS**, select the report you want to edit (select the check box), click **Edit**, make the changes, and click **Save**.

4. To export report data to a .csv file, click **REPORTS**, select the table report you want to export (click the report title), and from the **View Report** page, click **Export**.

   ☞ **Note:** You can export only Table reports.

   ☞ **Important:** For multi-select user-defined select user field, you will export individual record for each user.

5. To delete reports, click **REPORTS**, select one or more reports you want to delete (select the check boxes) and click **Delete**.

   A confirmation message is displayed. Click **OK** to delete the selected reports.

Customize reports

You can customize your reports by modifying the parameters in `config.ini` file. For example, some reports require a particular data source such as datamart and some require a particular license type such as ALM.

You may want to hide or display reports depending on the availability of their data source or the type of license so that users do not see reports which they cannot run.

For example:

- There are reports which require a data store (datamart or EventQ's events) to be enabled.
- There are many reports that do not apply to SCM users.

Further, you can also change a report's tilte, description or timeToLive duration of cached data and so on.

You can customize reports by modifying the parameters in `config.ini` file. For example, if you want to customize the "Average Size by Area/Group" report, modify the parameters in `/opt/collabnet/teamforge/var/cliserver/app/reports/pkg/averageSizeByArea/config.ini` file. The following illustration shows a sample `config.ini` file.

```
[main]
title = Average Size by Area/Group
description = Shows the average story points or effort per artifact by group, customer, category, assigned to, priorit
outputType = bar
fields = planId,trackerIds,recursive,averageBy,groupBy,includeArtifacts
timeToLive = 2
almRequired = true
category = DistributionReports

[planId]
label = Planning Folder ID
type = wizard
object = plan
max = 1
required = true

[trackerIds]
label = Tracker ID
type = wizard
object = tracker
max = 10
required = false

[recursive]
label = Include
type = checkbox
values = Child planning folders

[averageBy]
label = Average by
type = select
values = Story Points,Estimated Effort,Remaining Effort,Actual Effort
default = Story Points

[groupBy]
label = Group By
type = select
values = Group,Category,Customer,Assigned To,Priority,Tracker,Team
default = Group

[includeArtifacts]
label = Include artifacts
type = radio
values = All,Open,Closed
default = All

[bar]
title = Average Artifact [:var:box.field.averageBy:] by [:var:box.field.groupBy:] ([:var:box.field.includeArtifacts:])
width = 500
height = 400
xAxis = Average [:var:box.field.averageBy:]
yAxis = [:var:box.field.groupBy:]
```

Here's a sample list of parameters you can modify:

- *almRequired*

  - If *almRequired*=true: The site should be in the ALM mode and the users should have an ALM license to operate the ALM reports.
  - If *almRequired*=false: The site can be in either of the modes; both ALM and SCM licensed users can operate these reports.
- *devopsrequired*

  - If *devopsrequired*=true: Users should have DevOps license to operate DevOps reports.
- *scmrequired*

  - If *scmrequired*=true: Users should have Version Control license to operate SCM reports.
- *eventsRequired*

  - If *eventsRequired*=true: EventQ's events data store should be installed. Users can operate these reports irrespective of the site mode and license type.

- If *eventsRequired*=false: Users cannot operate these reports irrespective of the site mode and license type.
- *datamartRequired*

  - If *datamartRequired*=true: Datamart should be installed. Users can operate these reports irrespective of the site mode and license type.
  - If *datamartRequired*=false: Users cannot operate these reports irrespective of the site mode and license type.
- *category*

  - You can change the category of a report by specifying it in `config.ini`. For example, to change the category of a specific report from 'Agile' to 'Distribution', specify as shown below:

    category=DistributionReports
- *timeToLive*

  - If *timeToLive*=2: The cached report data invalidated and refreshed every 2 hours.
  - The *timeToLive* parameter sets the duration (in hours) after which the cached report data is invalidated and refreshed from the data source. The default *timeToLive* for reports that use operational database is 2 hours. It is 24 hours for reports that use datamart. You can change the *timeToLive* duration depending on your site's requirements. However, shorter *timeToLive* duration can impact performance, though.

👉 **Important:** Once you are done with your changes to `config.ini` file, run the `post-install.py` script to go live with the changes.

Advanced reporting and datamart access

Using external reporting and OLAP tools, query the datamart directly and generate reports. The database schema diagrams provide the means to create advanced query scripts to extract required information from the datamart.

Accessing the datamart for reporting provides more analytical data than is provided by the TeamForge user interface options. The external tool must connect directly to the datamart and is then granted read-only permission to all data in the datamart.

👉 **Note:** It is recommended that you limit the access to the datamart to a limited set of trusted users.

Datamart schemas

The `Users`, `SCM`, and `Tracker` schemas are documented here for use in queries to the datamart. The datamart uses a "Star Schema" design for tables.

User schema

Query the user schema to obtain useful information in the user database tables detailed here in a schema diagram.

User schema contains user login information; TeamForge captures one fact row for each user that is logged in during the day.

User schema diagram

*History (SCD-2) is maintained for the highlighted attributes
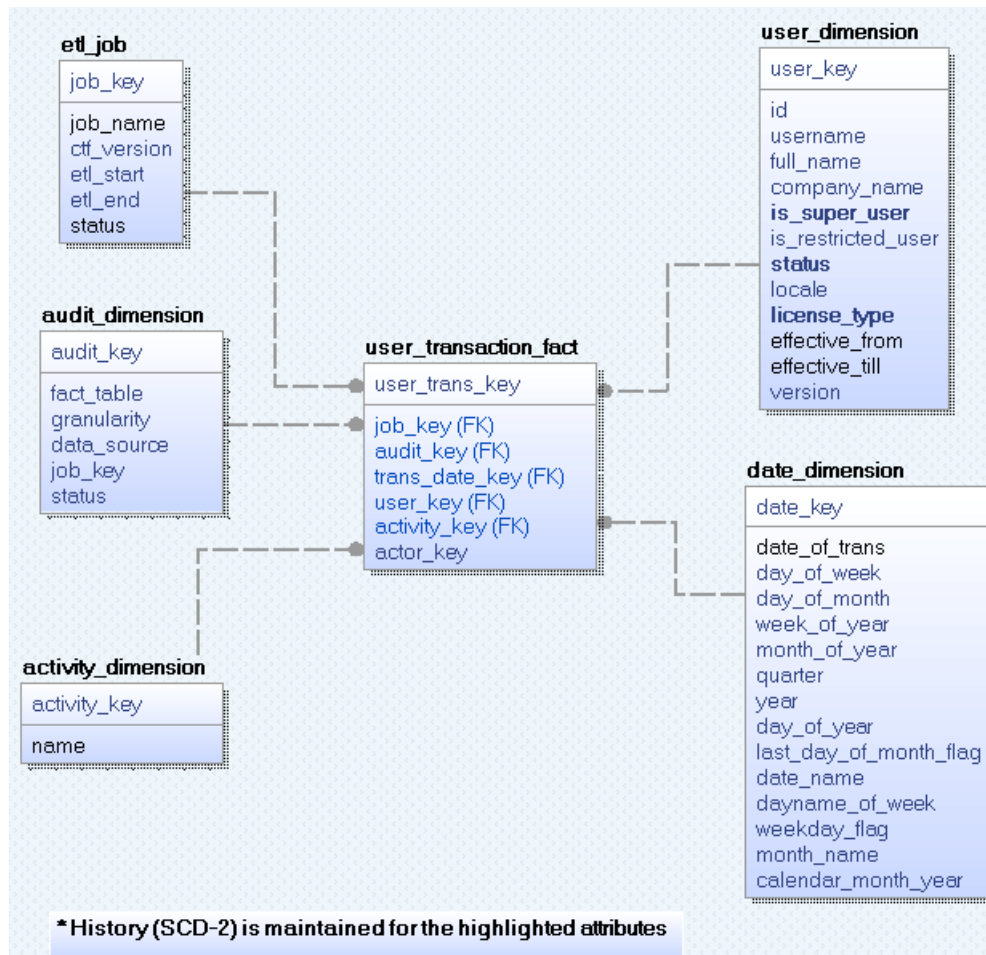
Description of user schema

etl_job
Used to track the ETL run status. There is one record per ETL run for a job, for example, Tracker ETL or User ETL. `etl_job` has a 1-to-many relationship with `audit_dimension` since a job may update more than one fact table. All report generation queries must "join" the `etl_job` table with the condition `etl_job.status=1`, thereby discarding data from incomplete ETL runs.

audit_dimension
Holds metadata about fact table records. There is one record per fact table for an ETL run.

date_dimension
Conformed dimension used for all transaction times.

user_dimension
Used for string user attributes and is a "slowly changing dimension of type 2 (SCD-2)." `is_super_user`, `status`, and `license_type` are the SCD-2 fields.

activity_dimension
Conformed dimension that stores the activity or transaction names for various activities being tracked.

user_transaction_fact
A fact-less fact table with user data of "daily" granularity.

Sample queries

You can obtain useful user information by querying the user database, and further refine the results by using filters on the "date", "user type" (admin or non), "status", and "license type" fields. For example:

- Number of users who are logged in, by day, over a period of time:

```
SELECT c.date_of_trans as Date, count(distinct(b.id)) as NumUsers

FROM user_transaction_fact a, user_dimension b, date_dimension c, etl_job
 d

WHERE a.user_key=b.user_key and a.trans_date_key=c.date_key and
 a.job_key=d.job_key

and d.status=1 and c.date_of_trans >= '2012-12-17' and c.date_of_trans <=
 '2012-12-21'

GROUP BY c.date_of_trans
```

- List of users who have logged in:

```
SELECT c.date_of_trans as Date, b.username as UserName
        FROM user_transaction_fact a, user_dimension b, date_dimension c,
 etl_job d
        WHERE a.user_key=b.user_key and a.trans_date_key=c.date_key and
 a.job_key=d.job_key
             and d.status=1 and c.date_of_trans >= '2012-12-17' and
 c.date_of_trans <= '2012-12-21'
        GROUP BY c.date_of_trans, b.username
```

SCM schema

Query the SCM schema to obtain useful commit information in the SCM tables detailed here in a schema diagram.

SCM schema diagram

*History (SCD-2) is maintained for the highlighted attributes
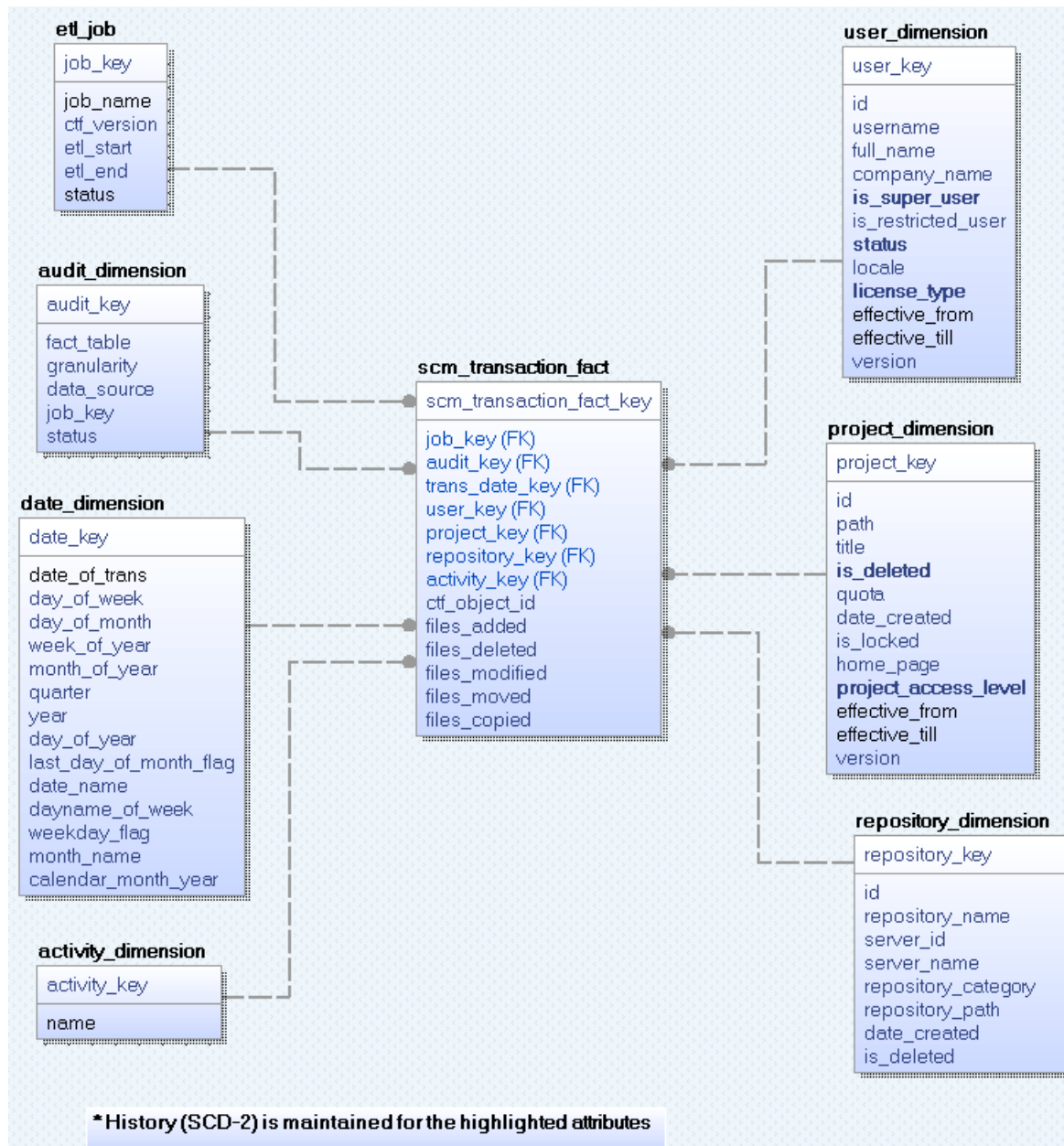
Description of SCM schema

etl_job	Used to track the ETL run status. There is one record per ETL run for a job, for example, Tracker ETL or User ETL. `etl_job` has a 1-to-many relationship with `audit_dimension` since a job may update more than one fact table. All report generation queries must "join" the `etl_job` table with the condition `etl_job.status=1,` thereby discarding data from incomplete ETL runs.
audit_dimension	Holds metadata about fact table records. There is one record per fact table for an ETL run.
date_dimension	Conformed dimension used for all transaction times.

activity_dimension	Conformed dimension that stores the activity or transaction names for various activities being tracked.
user_dimension	Used for string user attributes and is a "slowly changing dimension of type 2 (SCD-2)." `is_super_user`, `status`, and `license_type` are the SCD-2 fields.
project_dimension	Used for storing project attributes and is a "slowly changing dimension of type 2 (SCD-2)." `is_deleted` and `project_access_level` are the SCD-2 fields.
repository_dimension	Used for storing repository attributes and is a "slowly changing dimension of type 1."
scm_transaction_fact	The fact table for SCM activities with "transaction" granularity. TeamForge inserts a row in this table for every SCM activity that it processes in a transaction.

- TeamForge object id , if available.
- Number of files added, deleted, modified, moved, copied, if applicable.

Sample queries

You can obtain useful SCM information by querying the SCM database. For example:

- Number of SCM commits, sorted by date:

```
select b.date_of_trans as Date,
          count(a.scm_transaction_fact_key) as NumCommits
          from scm_transaction_fact a, date_dimension b
          where a.trans_date_key=b.date_key
          group by b.date_of_trans
```

- Number of SCM commits, with quarterly trend:

```
select 'Q'||b.quarter as Quarter,
          count(a.scm_transaction_fact_key) as NumCommits
          from scm_transaction_fact a, date_dimension b
          where a.trans_date_key=b.date_key
          group by b.quarter
```

- List of users who made commits.

```
select b.username as UserName,
          count(a.scm_transaction_fact_key) as NumCommits
          from scm_transaction_fact a, user_dimension b
          where a.user_key=b.user_key
          group by b.username
```

- Project-wise commit data:

```
select b.id as ProjectId, b.title as ProjectName,
          count(a.scm_transaction_fact_key) as NumCommits
          from scm_transaction_fact a, project_dimension b
          where a.project_key=b.project_key
          group by b.id, b.title
```

- Commits by date, in a specific project:

```
select c.date_of_trans as Date, b.id as ProjectId, b.title as
ProjectName,
          count(a.scm_transaction_fact_key) as NumCommits
```

```
               from scm_transaction_fact a, project_dimension b,
     date_dimension c
               where a.project_key=b.project_key and
     a.trans_date_key=c.date_key
                  and b.id='proj1008'
               group by c.date_of_trans, b.id, b.title
```

Tracker schema

Query the tracker schema to obtain useful information in the tracker database tables detailed here in a schema diagram.

TeamForge constructs the state or field values of the artifact during each update. The schema addresses both activity queries and total count queries.

TeamForge includes information only for activity in fixed and default artifact fields; flex-fields are not currently supported. This information includes artifact `create`, `move`, `update` for fixed-value changes, `delete`, `open_to_close`, and `close_to_open`.

Tracker schema diagram
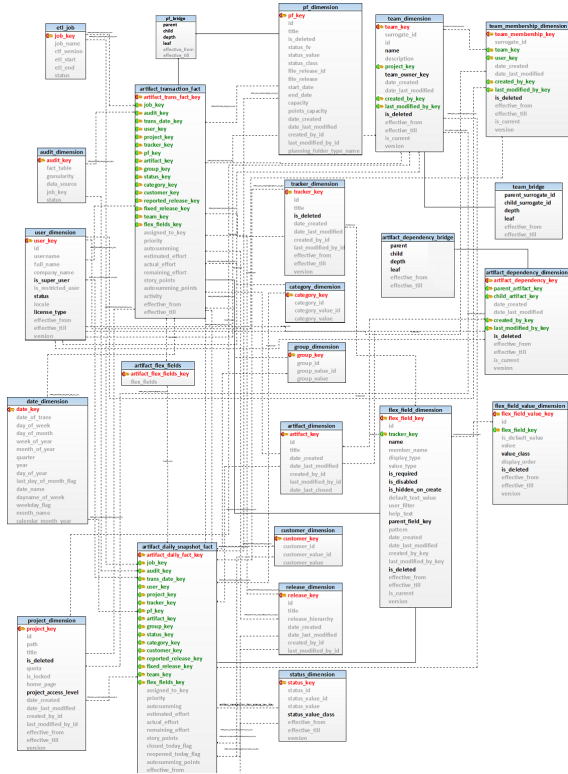
Description of tracker schema

etl_job	Used to track the ETL run status. There is one record per ETL run for a job, for example, Tracker ETL or User ETL. `etl_job` has a 1-to-many relationship with `audit_dimension` since a job may update more than one fact table. All report generation queries must "join" the `etl_job` table with the condition `etl_job.status=1`, thereby discarding data from incomplete ETL runs.
audit_dimension	Holds metadata about fact table records. There is one record per fact table for an ETL run.

date_dimension	Conformed dimension used for all transaction times.
user_dimension	Used for string user attributes and is a "slowly changing dimension of type 2 (SCD-2)." `is_super_user`, `status`, and `license_type` are the SCD-2 fields.
project_dimension	Used for storing project attributes and is a "slowly changing dimension of type 2 (SCD-2)." `is_deleted` and `project_access_level` are the SCD-2 fields.
pf_dimension	The planning folder dimension for storing planning folder attributes. Use this table to generate reports without planning folder hierarchy.
pf_bridge	A table for the planning folder hierarchy containing end-of-day status. `pf_bridge` is a "slowly changing dimension of type 2 (SCD-2)." You must limit queries to a given parent planning folder while joining with the `pf_bridge` table. Use `pf_bridge` to generate reports around planning folder hierarchy by joining with parent tables such as `artifact_daily_snapshot_fact` or `artifact_transaction_fact`.
	Joining `pf_bridge` with `artifact_transaction_fact` generates correct results only if end-of-day in queries is set to "12:00 a.m.". While formulating queries with `pf_bridge`, please note:
	`parent` and `child` fields — Contain values from `pf_dimension.pf_key`; a depth of 0 indicates self. `leaf` field — Is true if the child is a leaf node, otherwise false. Every planning folder will have an entry here with a depth of 0; every parent planning folder will have entries for all of its children, recursively, up to the leaf node of all branches. `effective_from` and `effective_till` fields — Indicate the period when the parent-child relationship is correct, and can be used in queries to get the hierarchy for a specified time period.
tracker_dimension	Used for storing tracker attributes and is a "slowly changing dimension of type 2 (SCD-2)." `is_deleted` is the changing field that act as a filter for reports .
artifact_dimension	Used for storing artifact data and is a "slowly changing dimension of type 1."
group_dimension	Used for holding values of the TeamForge tracker field "group". `group_dimension` has values for all artifacts from all Trackers.
status_dimension	Used for holding values of the TeamForge tracker field "status". The `status_value_class` field represents the meta-status of an artifact and has values "Open" and "Close". `status_dimension` has values for all artifacts from all Trackers.

category_dimension	Used for holding values of the TeamForge tracker field "category". `category_dimension` has values for all artifacts from all Trackers.
customer_dimension	Used for holding values of the TeamForge tracker field "customer". `customer_dimension` has values for all artifacts from all Trackers.
release_dimension	Used for holding values of the TeamForge tracker fields "Reported in Release" and "Fixed in Release". `release_dimension` has values for all artifacts from all Trackers.
artifact_transaction_fact	Every change in fixed or default artifact field values, or to project or tracker artifacts, results in a row inserted to `artifact_transaction_fact`. Changes to flex-field values, adding comments, attachments, and so on do not add a row to the table. The `artifact_dimension.date_last_modified` field has the time in the source tracker at the time of the ETL run. `artifact transaction fact` can be used to generate reports around activities such as create, update, delete and move, and for intra-day reports. `artifact_transaction_fact` has a "transaction" granularity.
artifact_daily_snapshot_fact	An aggregate table that holds the daily snapshot data or end-of-day status. `artifact_daily_snapshot_fact` can be used to generate reports for artifact close & re-open counts. It is recommended to use this table for end-of-day reports as it has fewer rows compared to `artifact_transaction_fact`. `artifact_daily_snapshot_fact` has a "daily" granularity.
team_dimension	A dimension for holding values of Teams. This is a "slowly changing dimension of type 2 (SCD-2)". Use this table to generate reports without team hierarchy.
team_membership_dimension	This is more or less a factless fact table that holds the changes in terms of Team memberships. This is a "slowly changing dimension of type 2 (SCD-2)".
team_bridge	A table for handling team hierarchy containing end-of-day status. `team_bridge` is a "slowly changing dimension of type 2 (SCD-2)." You must limit queries to a given parent team while joining with the `team_bridge` table. Use `team_bridge` to generate reports around team hierarchy (ex: a report on all artifacts assigned to Team 1 and its child teams) by joining with parent tables such as `artifact_daily_snapshot_fact` or `artifact_transaction_fact`. Joining `team_bridge` with `artifact_transaction_fact` generates correct results only if end-of-day in queries is set to "12:00 a.m.". While formulating queries with `team_bridge`, please note:

- parent_surrogate_id and child_surrogate_id fields — Contain values from team_dimension.surrogate_id; a depth of 0 indicates self.
- leaf field — Is true if the child is a leaf node, otherwise false. Every team will have an entry here with a depth of 0; every parent team will have entries for all of its children, recursively, up to the leaf node of all branches .
- effective_from and effective_till fields — Indicate the period when the parent-child relationship is correct, and can be used in queries to get the hierarchy for a specified time period.

> ☞ **Note:** team_dimension is a "slowly changing dimension of type 2". So, make sure that the JOIN clause in queries that join team_bridge and team_dimension includes the date and time as well.

flex_field_dimension	Used for storing information about flex fields and is a "slowly changing dimension of type 2 (SCD-2)".
flex_field_value_dimension	Used for storing information about flex field values and is a "slowly changing dimension of type-2 (SCD-2)".
artifact_dependency_dimension	Used for storing relationship between artifacts and is a "slowly changing dimension of type-2 (SCD-2)".
artifact_dependency_bridge	A table for the artifacts hierarchy containing end-of-day status. This is a "slowly changing dimension of type-2 (SCD-2). Use this table to generate reports around artifact hierarchy.
artifact_flex_fields	A table to bind the flex field updates on artifacts into a well formed XML, which stores the 'field key', 'value' and 'type' of the flex field.

Sample queries

You can obtain useful tracker information by querying the tracker database. For example:

- Number of artifacts created in a tracker, sorted by date:

```
SELECT b.date_of_trans, count(a.artifact_key)
         FROM artifact_transaction_fact a, date_dimension b,
tracker_dimension c
         WHERE a.trans_date_key=b.date_key and
a.tracker_key=c.tracker_key
         and a.activity='Create' and c.title='Tracker-1'
         and b.date_of_trans >= '2011-10-31' and b.date_of_trans <=
'2011-11-07'
         GROUP BY b.date_of_trans
         ORDER BY b.date_of_trans
```

- Number of artifacts created in a tracker, sorted by date and priority:

```
SELECT b.date_of_trans, d.title as Tracker, 'P'||a.priority as
Priority,c.id ArtifactId
         FROM artifact_transaction_fact a, date_dimension b,
artifact_dimension c, tracker_dimension d
```

```
        WHERE a.trans_date_key=b.date_key and
a.artifact_key=c.artifact_key and a. tracker_key=d.tracker_key
        and a.activity='Create' and d.title='Tracker-1'
        and b.date_of_trans >= '2011-10-31' and b.date_of_trans <=
'2011-11-07'
        ORDER BY b.date_of_trans, d.title, Priority
```

- Number of artifacts created on a particular day in a particular planning folder:

```
SELECT b.date_of_trans, c.id ArtifactId, c.title
        FROM artifact_transaction_fact a, date_dimension b,
artifact_dimension c, pf_dimension d, pf_bridge e
        WHERE a.trans_date_key=b.date_key and
a.artifact_key=c.artifact_key and a.pf_key=e.child
        and d.pf_key=e.parent and a.activity='Create' and
d.title='Pf-1'
        and '2011-10-31' >= e.effective_from and '2011-10-31' <
e.effective_till
        and b.date_of_trans = '2011-10-31'
        ORDER BY b.date_of_trans
```

- Number of closed tracker artifacts, sorted by day:

```
SELECT b.date_of_trans, count(a.artifact_key)
        FROM artifact_daily_snapshot_fact a, date_dimension b,
tracker_dimension c
        WHERE a.trans_date_key=b.date_key and
a.tracker_key=c.tracker_key and a.closed_today_flag='true'
        and c.title='Tracker-1' and b.date_of_trans >= '2011-10-31'
and b.date_of_trans <= '2011-11-07'
        GROUP BY b.date_of_trans
        ORDER BY b.date_of_trans
```

- List of artifacts in the "Open" state, sorted by day:

```
SELECT a.date_of_trans, c.id, c.title
        FROM date_dimension a inner join artifact_daily_snapshot_fact
b
        on a.date_of_trans >= date(b.effective_from) and
a.date_of_trans < date(b.effective_till)
        inner join artifact_dimension c on
b.artifact_key=c.artifact_key
        inner join status_dimension d on b.status_key=d.status_key
        WHERE d.status_value_class='Open' and a.date_of_trans >=
'2011-10-31'
        and a.date_of_trans <= '2011-11-02'
        ORDER BY a.date_of_trans
```

- List of artifacts assigned to a team (including child teams) on a given day:

```
select ad.id as artifact_id
        from artifact_transaction_fact fact
        inner join date_dimension dd on (dd.date_of_trans + 1) between
fact.effective_from and fact.effective_till
        inner join team_dimension child_td on child_td.team_key =
fact.team_key
        inner join team_bridge tb on tb.child_surrogate_id =
child_td.surrogate_id and
            (dd.date_of_trans + 1) between tb.effective_from and
tb.effective_till
        inner join team_dimension parent_td on tb.parent_surrogate_id =
parent_td.surrogate_id and
            (dd.date_of_trans + 1) between parent_td.effective_from
and parent_td.effective_till
```

```
        inner join artifact_dimension ad on ad.artifact_key =
fact.artifact_key
        inner join etl_job ej on ej.job_key = fact.job_key and
ej.status = 1
        where dd.date_of_trans = '2015-01-01' and parent_td.id =
'team1002'
        order by artifact_id
```

Datamart access using external tools

Use external tools to directly query the PostgreSQL or Oracle datamarts to access more TeamForge data than is available through options on the TeamForge user interface.

Accessing the datamart

You can use OLAP or GUI tools to query your datamart directly to procure all the TeamForge data that is relevant to your analysis.

The external tool must be in the same network as the datamart to ensure fast access with a direct connection to the database. On the TeamForge host machine, you must enable `REPORTS_DATABASE_HOST` and `REPORTS_DATABASE_PORT` for remote access.

☞ **Note:** You only have read access; the datamart does not allow writes.

Enabling the datamart for access

You must enable the datamart for direct queries and re-start the site. Once enabled, all data in the datamart can be queried using external tools.

Enable TeamForge (PostgreSQL datamart) for queries from external tools

You must enable, then re-start TeamForge so that you can use an external tool to query the datamart directly.

You cannot use external reporting tools to connect directly to a datamart if you have a single-box installation using `HOST_localhost`.

1. Edit the `site-options.conf` file. Set `REPORTS_DB_ACCESS_HOSTS` to the IP address or IP address range (CIDR address) from which the tool will establish connection to the datamart. Specify multiple IPs as a comma-separated list.

   For example: `REPORTS_DB_ACCESS_HOSTS` = 10.0.0.2,10.2.1.0/24

   ☞ **Note:**
   - If the TeamForge site is not within your network and a Network Address Translation (NAT) is configured, then specify the NAT's external facing IP in the token.
   - If you have used an "advanced" installation method to install TeamForge, you must also manually add the IP addresses to the `pg_hba.conf` file.

2. In `site-options.conf`, set `REPORTS_DATABASE_READ_ONLY_USER` and `REPORTS_DATABASE_READ_ONLY_PASSWORD` with your user and password.

3. On the TeamForge host machine, enable `REPORTS_DATABASE_HOST` and `REPORTS_DATABASE_PORT` for remote access.

4. Re-create the runtime environment.

   ```
   sudo ./install.sh -r -n -V -d <site_dir>
   ```

5. Start the site.

   ```
   sudo /etc/init.d/collabnet start
   ```

   You can now enable trusted users to establish connections by providing them with values of options `REPORTS_DATABASE_HOST`, `REPORTS_DATABASE_PORT`, `REPORTS_DATABASE_READ_ONLY_USER`, and `REPORTS_DATABASE_READ_ONLY_PASSWORD`.

Enable TeamForge (Oracle datamart) for queries from external tools

You must enable, then re-startTeamForge so that you can use an external tool to query the datamart directly.

You can use external reporting tools to connect directly to a datamart only if you do not have a single-box installation using `HOST_localhost`.

1. In `site-options.conf`, set `REPORTS_DATABASE_READ_ONLY_USER` and `REPORTS_DATABASE_READ_ONLY_PASSWORD` with your user and password.

2. On the TeamForge host machine, enable `REPORTS_DATABASE_HOST` and `REPORTS_DATABASE_PORT` for remote access.

3. Re-create the runtime environment.

   ```
   sudo ./install.sh -r -n -V
   ```

4. Start the site.

   ```
   /etc/init.d/httpd start
   sudo /etc/init.d/collabnet start
   ```

   You can now enable trusted users to establish connections by providing them with values of options `REPORTS_DATABASE_HOST`, `REPORTS_DATABASE_PORT`, `REPORTS_DATABASE_READ_ONLY_USER`, and `REPORTS_DATABASE_READ_ONLY_PASSWORD`.

Common errors while connecting to PostgreSQL or Oracle datamarts

You may encounter these errors while configuring your datamart to allow queries from external tools.

PostgreSQL database access error

Error: `psql: FATAL: no pg_hba.conf entry for host "111.111.111.111", user "test", database "datamart"`

This error in a PostgreSQL datamart is because TeamForge is not granting access to your IP address.

Solution: Have the TeamForge administrator grant access to your IP address.

PostgreSQL authentication error

Error: `psql: FATAL: password authentication failed for user "test"`

Solution: Specify the correct user name and password.

PostgreSQL connection error

Error: `psql: could not connect to server: Connection refused. Is the server running on host "<datamart-host>" and accepting TCP/IP connections on port 5632?`

Solution:

- Check the host and port numbers; the database must be running on host "datamart-host" and accept TCP/IP connections on port 5632.
- Check if remote access is enabled on TeamForge; the Teamforge administrator can enable the required access.

Oracle connection error

Error: `ORA-12505, TNS:listener does not currently know of SID given in connect descriptor.`

This error in an Oracle datamart occurs when some connection parameters are set incorrectly.

Solution: Check and set your connection parameters appropriately.

Query PostgreSQL datamart based on flex fields

Create query scripts to query and extract the required information from the PostgreSQL datamart. Make sure these scripts are available to any user (including users with read-only permission) who wants to execute these scripts. This topic lists the functions and sample queries for a few specific use cases.

> **Important:**
>
> - Log into TeamForge as a Reporting user and run the queries. Note that this is a one-time process.
> - Use the following flex field functions by joining the `flex_field_dimension`, `artifact_flex_field` and `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` tables as it depends on the *XML data* and *flex field key* generated by the platform.

- Common script used across all flex field functions:

```
CREATE OR REPLACE FUNCTION array_search(needle anyelement, haystack
 anyarray)
  RETURNS integer AS
$BODY$
    SELECT i
      FROM generate_subscripts($2, 1) AS i
      WHERE $2[i] = $1
  ORDER BY i
$BODY$
  LANGUAGE sql STABLE
  COST 100;
```

- **Function for Date flex fields**

  Input arguments
  - Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables)
  - Flex field key (derived automatically from `flex_field_dimension` table and other tables)

  Output arguments Date flex field values

```
CREATE OR REPLACE FUNCTION get_artifact_date_value(artifact_xml xml,
 field_key integer)
  RETURNS character varying AS
$BODY$
DECLARE
field_value_key varchar(9055) default '';
 BEGIN
 SELECT cast((xpath('/fields/field/@val',artifact_xml))
 [(array_search(field_key::text,(xpath('/fields/field/
@key',artifact_xml))::text[])))]
  as varchar) into field_value_key;
 Return field_value_key;
 END;
$BODY$
  LANGUAGE plpgsql VOLATILE
  COST 100;
```

- **Function for Text flex fields**

  Input arguments
  - Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or)

artifact_daily_snapshot_fact and
artifact_flex_field tables)

- Flex field key (derived automatically from
  flex_field_dimension table and other
  tables)

Output arguments   Text flex field values

```
CREATE OR REPLACE FUNCTION replacen(artifact_xml1 xml)
  RETURNS text AS
$BODY$
 select replace ((select
 replace(artifact_xml1::varchar(3000),'<!','&lt;!')),']>',']&gt;')

$BODY$
  LANGUAGE sql STABLE
  COST 100;


CREATE OR REPLACE FUNCTION get_artifact_text_value(artifact_xml xml,
 field_key integer)
  RETURNS character varying AS
$BODY$
 DECLARE
  field_value_key varchar(9055) default '';
 BEGIN
 SELECT cast((xpath('/fields/field/@val',replacen(artifact_xml)::xml))
 [(array_search(field_key::text,(xpath('/fields/field/
@key',replacen(artifact_xml)::xml))::text[])))]
  as varchar) into field_value_key;
 Return field_value_key;
 END;
$BODY$
  LANGUAGE plpgsql VOLATILE
  COST 100;
```

- **Function for Single-select flex fields**

  Input arguments
  - Flex field XML (derived automatically by joining
    the artifact_transaction_fact (or)
    artifact_daily_snapshot_fact and
    artifact_flex_field tables)
  - Flex field key (derived automatically from
    flex_field_dimension table and other
    tables)

  Output arguments   Values selected or stored in single-select flex field

```
CREATE OR REPLACE FUNCTION get_artifact_select_value(artifact_xml xml,
 field_key integer)
  RETURNS character varying AS
$BODY$
 DECLARE
  field_value_key varchar(9055) default '';
 singleSelectValues varchar(9055) default '';
 BEGIN
  SELECT cast((xpath('/fields/field/@val',artifact_xml))
 [(array_search(field_key::text,(xpath('/fields/field/
@key',artifact_xml))::text[])))]
  as varchar) into field_value_key;
```

```
  Select value into singleSelectValues  from flex_field_value_dimension
 where flex_field_value_key::text = (field_value_key::text);

 Return singleSelectValues;
 END;
$BODY$
  LANGUAGE plpgsql VOLATILE
  COST 100;
```

- **Function for Multi-select flex fields**

  Input arguments

  - Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
  - Flex field key (derived automatically from `flex_field_dimension` table and other tables).
  - Specific value(s) stored in the field or the keyword 'ALL' for retrieving all values stored in multi-select flex field.
  - Logical conditional operator. Possible values are 'ALL' or 'ANY'. The argument value of 'ALL' performs a search equivalent to the 'AND' condition and the value of 'ANY' performs a search equivalent to the 'OR' condition.

  Output arguments   Values selected or stored in multi-select flex field

```
     -----First Execute Inner Function------
-------Inner function start------
CREATE OR REPLACE FUNCTION
 get_artifact_multiselect_values_any(artifact_xml xml, field_key integer,
 selectedfields text[])
RETURNS text AS
$BODY$
DECLARE
fieldValues text;
multiFieldValues text default '';
splitfield text[];
splitfield2 text[];
loop1 integer DEFAULT 1;
 loop2 integer DEFAULT 1;
arrayLength integer DEFAULT 1;
 arrayLength2 integer DEFAULT 1;
multifield text DEFAULT '';
filteredField text Default '';
BEGIN
 Select cast((xpath('/fields/field/@val',artifact_xml))
[array_search(field_key::text,(xpath('/fields/field/
@key',artifact_xml)::text[]))]
 as varchar(9055)) into fieldValues;
Select array_length(regexp_split_to_array(fieldValues, ','),1) into
 arrayLength;
Select regexp_split_to_array(fieldValues, ',') into splitfield;
IF arrayLength IS NOT NULL THEN
        FOR loop1 IN 1..arrayLength LOOP
        Select value into multiFieldValues from flex_field_value_dimension
 where flex_field_value_key = splitfield[loop1]::integer;
        multifield := concat(multifield,',',multiFieldValues);
        EXIT WHEN loop1 > arrayLength;
```

```
        END LOOP;


        multifield := rtrim(multifield,'" ');
        multifield := ltrim(multifield,',');

Select
 array_length(regexp_split_to_array(array_to_string(selectedFields,','),
 ','),1) into arrayLength2;
Select regexp_split_to_array(array_to_string(selectedFields,','), ',')
 into splitfield2;
IF selectedFields = Array['ALL'] THEN
filteredField:=multifield;
Return filteredField;
ELSE
FOR loop2 IN 1..arrayLength2 LOOP
IF   multifield ~ splitfield2[loop2] THEN
      filteredField := concat(splitfield2[loop2],',',filteredField);
END IF;
EXIT WHEN loop2 > arrayLength2;
END LOOP;
END IF;
END IF;
filteredField := rtrim(filteredField,'" ');
filteredField := rtrim(filteredField,',');
Return filteredField;
 END;
$BODY$
LANGUAGE plpgsql VOLATILE
COST 100;
-------Inner function end------


CREATE OR REPLACE FUNCTION get_artifact_multiselect_value(artifact_xml
 xml, field_key integer,selectedfields text[],condition character varying)
  RETURNS text as
  $BODY$
  DECLARE
   fieldValues text;
   loop1 integer default 1;

   loop2 integer default 1;
  arrayLength1 integer default 1;
  arrayLength2 integer default 1;
  arrayLength3 integer default 1;
  multiFieldValues text default '';
  multifield text DEFAULT '';
  returnvalues text DEFAULT '';
    i integer default 1;
    j integer default 1;
     v_array1 text[];
     v_array2 text[];
     v_array3 text[];
     results text Default '' ;
     filteredField text Default '';
     count integer default 0;
   BEGIN
       Select cast((xpath('/fields/field/@val',artifact_xml))
[array_search(field_key::text,(xpath('/fields/field/
@key',artifact_xml)::text[])))]
       as varchar(9055)) into fieldValues;
       Select array_length(regexp_split_to_array(fieldValues, ','),1) into
 arrayLength1;
       Select regexp_split_to_array(fieldValues, ',') into v_array1;
```

```
  IF arrayLength1 IS NOT NULL THEN
      FOR  loop1 in 1..arrayLength1 LOOP
      select value into multiFieldValues from  flex_field_value_dimension
 where flex_field_value_key=v_array1[loop1]::integer;
      multifield := concat(multifield,',',multiFieldValues);
      EXIT WHEN loop1 > arrayLength1;
      multifield := rtrim(multifield,'" ');
      multifield := ltrim(multifield,',');
      END LOOP;
      Select array_length(regexp_split_to_array(multifield, ','),1) into
 arrayLength2;
      Select regexp_split_to_array(multifield, ',') into v_array2;
      Select
 array_length(regexp_split_to_array(array_to_string(selectedfields,','),
 ','),1) into arrayLength3;
      Select regexp_split_to_array(array_to_string(selectedfields,','),
 ',') into v_array3;

              IF (UPPER(condition)=UPPER('ALL')) THEN
                  IF selectedFields = Array['ALL'] THEN
                       results:=multifield;
                  ELSE
    IF (v_array2 is not  null AND v_array3 is not null) THEN
        FOR i in 1..arrayLength3 LOOP
        FOR j in 1..arrayLength2 LOOP
            IF (v_array3[i]= v_array2[j]) THEN
          filteredField := concat(v_array3[i],',',filteredField);

          count=count+1;
          EXIT;
              END IF;

        END LOOP;
          END LOOP;
    ELSE
      results = false;

          END IF;
          IF (count=arrayLength3) THEN
          results=filteredField;
          ELSE
          results='';
          END IF;
          END IF;
         ELSE

          select
 get_artifact_multiselect_values_any(artifact_xml,field_key,selectedfields)
 into returnvalues;
              IF returnvalues IS NOT NULL THEN
                  results=returnvalues;
              ELSE
                   results='';
              END IF;

          END IF;
  END IF;
  results := rtrim(results,'" ');
  results := rtrim(results,',');
 RETURN(results);
 END;
$BODY$
LANGUAGE plpgsql VOLATILE
```

```
COST 100;
```

- **Function for Multi-user flex fields**

Input arguments	
	• Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
	• Flex field key (derived automatically from `flex_field_dimension` table and other tables).
	• Specific value(s) stored in the field or the keyword 'ALL' for retrieving all values stored in multi-select flex field.
	• Logical conditional operator. Possible values are 'ALL' or 'ANY'. The argument value of 'ALL' performs a search equivalent to the 'AND' condition and the value of 'ANY' performs a search equivalent to the 'OR' condition.

Output arguments	Values selected or stored in multi-user flex field

```
      -----First Execute Inner Function------

 ------Inner function start---
CREATE OR REPLACE FUNCTION get_artifact_user_values_any(artifact_xml xml,
 field_key integer, selectedfields text[])
  RETURNS text AS
$BODY$
DECLARE
fieldValues text;
multiFieldValues text default '';
splitfield text[];
splitfield2 text[];
loop1 integer DEFAULT 1;
 loop2 integer DEFAULT 1;
arrayLength integer DEFAULT 1;
 arrayLength2 integer DEFAULT 1;
multifield text DEFAULT '';
filteredField text Default '';
BEGIN
 Select cast((xpath('/fields/field/@val',artifact_xml))
[array_search(field_key::text,(xpath('/fields/field/
@key',artifact_xml)::text[])))]
 as varchar(9055)) into fieldValues;
Select array_length(regexp_split_to_array(fieldValues, ','),1) into
 arrayLength;
Select regexp_split_to_array(fieldValues, ',') into splitfield;
IF arrayLength IS NOT NULL THEN
   FOR loop1 IN 1..arrayLength LOOP
     Select full_name into multiFieldValues from user_dimension where
 user_key = splitfield[loop1]::integer;
     multifield := concat(multifield,',',multiFieldValues);
     EXIT WHEN loop1 > arrayLength;
   END LOOP;
     multifield := rtrim(multifield,'" ');
     multifield := ltrim(multifield,',');

    Select
 array_length(regexp_split_to_array(array_to_string(selectedFields,','),
 ','),1) into arrayLength2;
```

```
      Select regexp_split_to_array(array_to_string(selectedFields,','),
 ',') into splitfield2;
       IF selectedFields = Array['ALL'] THEN
       filteredField:=multifield;
       Return filteredField;
       ELSE
           FOR loop2 IN 1..arrayLength2 LOOP
           IF   multifield ~ splitfield2[loop2] THEN
        filteredField := concat(splitfield2[loop2],',',filteredField);
           END IF;
           EXIT WHEN loop2 > arrayLength2;
           END LOOP;
       END IF;
END IF;
filteredField := rtrim(filteredField,'" ');
filteredField := rtrim(filteredField,',');
Return filteredField;
 END;
$BODY$
  LANGUAGE plpgsql VOLATILE
  COST 100;

 ------Inner Function End-----

create or replace FUNCTION get_artifact_user_value(artifact_xml xml,
 field_key integer,selectedfields text[],condition character varying)
   RETURNS text as
  $BODY$
  DECLARE
   fieldValues text;
   loop1 integer default 1;

   loop2 integer default 1;
  arrayLength1 integer default 1;
  arrayLength2 integer default 1;
  arrayLength3 integer default 1;
  multiFieldValues text default '';
  multifield text DEFAULT '';
  returnvalues text DEFAULT '';
    i integer default 1;
    j integer default 1;
     v_array1 text[];
     v_array2 text[];
     v_array3 text[];
     results text Default '' ;
     filteredField text Default '';
     count integer default 0;
   BEGIN
       Select cast((xpath('/fields/field/@val',artifact_xml))
[array_search(field_key::text,(xpath('/fields/field/
@key',artifact_xml)::text[]))]
       as varchar(9055)) into fieldValues;
       Select array_length(regexp_split_to_array(fieldValues, ','),1) into
 arrayLength1;
       Select regexp_split_to_array(fieldValues, ',') into v_array1;
  IF arrayLength1 IS NOT NULL THEN
       FOR  loop1 in 1..arrayLength1 LOOP
       select full_name into multiFieldValues from  user_dimension where
 user_key=v_array1[loop1]::integer;
       multifield := concat(multifield,',',multiFieldValues);
       EXIT WHEN loop1 > arrayLength1;
       multifield := rtrim(multifield,'" ');
       multifield := ltrim(multifield,',');
       END LOOP;
```

```
        Select array_length(regexp_split_to_array(multifield, ','),1) into
arrayLength2;
        Select regexp_split_to_array(multifield, ',') into v_array2;
        Select
array_length(regexp_split_to_array(array_to_string(selectedfields,','),
','),1) into arrayLength3;
        Select regexp_split_to_array(array_to_string(selectedfields,','),
',') into v_array3;

             IF (UPPER(condition)=UPPER('ALL')) THEN
                 IF selectedFields = Array['ALL'] THEN
                     results:=multifield;
                 ELSE
     IF (v_array2 is not  null AND v_array3 is not null) THEN
         FOR i in 1..arrayLength3 LOOP
         FOR j in 1..arrayLength2 LOOP
             IF (v_array3[i]= v_array2[j]) THEN
         filteredField := concat(v_array3[i],',',filteredField);

         count=count+1;
         EXIT;
             END IF;

         END LOOP;
         END LOOP;
     ELSE
       results = false;

           END IF;
           IF (count=arrayLength3) THEN
           results=filteredField;
           ELSE
           results='';
           END IF;
           END IF;
          ELSE

          select
 get_artifact_user_values_any(artifact_xml,field_key,selectedfields) into
 returnvalues;
                 IF returnvalues IS NOT NULL THEN
                     results=returnvalues;
                 ELSE
                     results='';
                 END IF;

          END IF;
  END IF;
  results := rtrim(results,'" ');
  results := rtrim(results,',');
 RETURN(results);
 END;
$BODY$
LANGUAGE plpgsql VOLATILE
COST 100;
```

Sample use cases and queries

☞ **Note:**

- Flex field Name, Value, Tracker title and so on that are used in the filtering conditions are case-sensitive.
- See *Case 7* below for building a case-insensitive search using upper() or lower() functions.

Use case 1: Filter Date and Text flex fields

Suppose you want to retrieve data based on the following assumptions:

- Date flex field named 'CreatedDate' that has a value of '2015-09-02 00:00:00'
- Text flex field named 'Text102' with part of its value containing the pattern 'soft'
- Tracker title is 'Tracker101'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system
- The date passed as input should be of the format "YYYY-MM-DD HH24:MI:SS"

Sample query

```
select
 t.Date as Date,
 t.Project as Project,
 t.Tracker as Tracker,
 t.Planing_folder as Planing_folder,
 'P'||t.Priority as Priority,
 count(distinct t.artifact_key)
from

(          select
 b.date_of_trans as Date,
    e.title as Project,c.title as
 Tracker,
    d.title as Planing_folder,
    'P'||a.priority as Priority,

 get_artifact_text_value(ff.flex_fields,ffd.flex
 as TextFlexField,
    a.artifact_key
from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields ff,
    tracker_dimension c,
    pf_dimension d,
    project_dimension e ,
    flex_field_dimension ffd
where
    a.trans_date_key=b.date_key
    and ff.artifact_flex_fields_key
 = a.flex_fields_key
    and a.tracker_key=c.tracker_key
    and a.pf_key=d.pf_key
    and a.project_key=e.project_key
    and
 a.tracker_key=ffd.tracker_key
    and e.title='Project SCD-2 Test'
    and ffd.name in ('Text102')
    and c.title in ('Tracker101')
    and
 get_artifact_text_value(ff.flex_fields,ffd.flex
 like 'soft%'

union

    select   b.date_of_trans as Date,
```

```
    e.title as Project,c.title as
Tracker,
    d.title as Planing_folder,
    'P'||a.priority as Priority,

 regexp_split_to_table(get_artifact_date_value(f
 ffd.flex_field_key),E',') as
 DateFlexField,
    a.artifact_key

from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields ff,
    tracker_dimension c,
    pf_dimension d,
    project_dimension e ,
    flex_field_dimension ffd
where
    a.trans_date_key=b.date_key
    and ff.artifact_flex_fields_key
 = a.flex_fields_key
    and a.tracker_key=c.tracker_key
    and a.pf_key=d.pf_key
    and a.project_key=e.project_key
    and
 a.tracker_key=ffd.tracker_key
    and e.title='Project SCD-2 Test'
    and ffd.name in ('CreatedDate')
    and c.title in ('Tracker101')
    and
 get_artifact_date_value(ff.flex_fields,
 ffd.flex_field_key)='2015-09-02
 00:00:00') as t
group by 1,2,3,4,5
order by 1,2,3,4,5;
```

Use case 2: Filter Single-select flex fields

Suppose you want to retrieve data based on the following
assumptions:

- Flex field name: 'FSS1'
- Value selected or stored: 'S1'
- Tracker title: 'Tracker2'

☞ **Note:**

- You can change the field name, tracker title
  and values based on the data in your system

Sample query

```
select
 b.date_of_trans as Date,
 pd.title as Project,
 td.title as Tracker,
 d.title as Planning_folder,
 'P'||a.priority as Priority ,

 get_artifact_select_value(aff.flex_fields,ffd.f
 as SingleSelectFlexField,
 count(distinct a.artifact_key)
 TotalArtifacts
from
 artifact_transaction_fact a,
```

```
   date_dimension b,
   artifact_flex_fields aff,
   project_dimension pd ,
   tracker_dimension td,
   pf_dimension d,
   flex_field_dimension ffd
 where
   a.trans_date_key=b.date_key
   and
  a.flex_fields_key=aff.artifact_flex_fields_key
   and a.tracker_key=td.tracker_key
   and a.project_key=pd.project_key
   and td.tracker_key=ffd.tracker_key
   and a.pf_key=d.pf_key
   and
  pd.title='ProjectTestFunctionSCD-2'
   and td.title in ('Tracker2')
   and ffd.name='FSS1'
   and
  get_artifact_select_value(aff.flex_fields,ffd.f
   --and
  date(a.effective_from)<=date(now())
   and
  date(a.effective_till)>'2012-04-01'
 group by 1,2,3,4,5,6
 order by 1,2,3,4,5,6;
```

Use case 3: Filter Text flex fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: 'Text101'
- Value contains: 'hello'
- Tracker name: 'Tracker101'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system

Sample query

```
select
     b.date_of_trans as Date,
   e.title as Project,
   c.title as Tracker,
   d.title as Planing_folder,
   'P'||a.priority as Priority,

  get_artifact_text_value(ff.flex_fields,ffd.flex
  as TextFlexField,
   count(distinct a.artifact_key)
  Artifacts
 from
   artifact_transaction_fact a,
   date_dimension b,
   artifact_flex_fields ff,
   tracker_dimension c,
   pf_dimension d,
   project_dimension e ,
   flex_field_dimension ffd
 where
   a.trans_date_key=b.date_key
   and ff.artifact_flex_fields_key =
  a.flex_fields_key
```

```
  and a.tracker_key=c.tracker_key
  and a.pf_key=d.pf_key
  and a.project_key=e.project_key
  and a.tracker_key=ffd.tracker_key
  and e.title='Project SCD-2 Test'
  and ffd.name = 'Text101'
  and c.title in ('Tracker101')
  and
get_artifact_text_value(ff.flex_fields,ffd.flex
like 'hello%'
  --and
date(a.effective_from)<=date(now())
  and
date(a.effective_till)>'2012-04-01'
group by 1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Use case 4: Filter Date flex fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: 'CreatedDate'
- Value: '2015-09-02 00:00:00'
- Tracker name: 'Tracker101'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system

Sample query

```
select
  b.date_of_trans as Date,
  e.title as Project,
  c.title as Tracker,
  d.title as Planing_folder,
  'P'||a.priority as Priority,

 regexp_split_to_table(get_artifact_date_value(f
 ffd.flex_field_key),E',') as
 DateFlexField,
  count(distinct a.artifact_key)

from
  artifact_transaction_fact a,
  date_dimension b,
  artifact_flex_fields ff,
  tracker_dimension c,
  pf_dimension d,
  project_dimension e ,
  flex_field_dimension ffd
where
      a.trans_date_key=b.date_key
      and
ff.artifact_flex_fields_key =
a.flex_fields_key
      and
a.tracker_key=c.tracker_key
      and a.pf_key=d.pf_key
      and
a.project_key=e.project_key
      and
a.tracker_key=ffd.tracker_key
```

```
      and e.title='Project SCD-2
Test'
      and ffd.name in
('CreatedDate')
      and c.title in ('Tracker101')
      and
get_artifact_date_value(ff.flex_fields,
ffd.flex_field_key)='2015-09-02
00:00:00'

group by  1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Use case 5: Multi-select flex fields

Suppose you want to retrieve data based on the following assumptions:

- Multi-select flex field name: 'Country'
- Value: 'Russia,India'
- Tracker name: 'TrackerN'
- Conditional parameter: 'ALL'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system
- Flex field name, value and tracker title that are used in the SQL filter conditions are case sensitive
- If you want to select all the values in User flex field, then pass 'ALL' as the conditional parameter
- If you want to select any value, then pass 'ANY' as the conditional parameter

Sample query

```
select b.date_of_trans as Date,
 e.title as Project,c.title as
 Tracker,
d.title as Planing_folder,'P'||
a.priority as Priority,
 get_artifact_multiselect_value(ff.flex_fields,f
 as MultiselectFlexField,
count(distinct a.artifact_key)
 Artifacts
--,a.artifact_key
from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields ff,
    tracker_dimension c,
    pf_dimension d,
    project_dimension e ,
    flex_field_dimension ffd
where
    a.trans_date_key=b.date_key
    and ff.artifact_flex_fields_key
 = a.flex_fields_key
    and a.tracker_key=c.tracker_key
    and a.pf_key=d.pf_key
    and a.project_key=e.project_key
```

```
        and
 a.tracker_key=ffd.tracker_key
    --and
 e.title='TestMultiFunction'
    and ffd.name = 'Country'
    and c.title in ('TrackerN')
    and
 get_artifact_multiselect_value(ff.flex_fields,f
='''
    --and
 date(a.effective_from)<=date(now())
 and
 date(a.effective_till)>'2012-04-01'
group by 1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Use case 6: Filter User flex fields

Suppose you want to retrieve data based on the following assumptions:

- User flex field name: 'Select User'
- Value: 'user1,user2'
- Tracker name: 'TrackerN'
- Conditional parameter: 'ALL'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system
- Flex field name, value and tracker title that are used in the SQL filter conditions are case sensitive
- If you want to select all the values in User flex field, then pass 'ALL' as the conditional parameter
- If you want to select any value, then pass 'ANY' as the conditional parameter

Sample query

```
select b.date_of_trans as Date,
 e.title as Project,c.title as
 Tracker,
d.title as Planing_folder,'P'||
a.priority as Priority,
 get_artifact_user_value(ff.flex_fields,ffd.flex
 as UserFlexField,
count(distinct a.artifact_key)
 Artifacts
--,a.artifact_key
from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields ff,
    tracker_dimension c,
    pf_dimension d,
    project_dimension e ,
    flex_field_dimension ffd
where
    a.trans_date_key=b.date_key
    and ff.artifact_flex_fields_key
 = a.flex_fields_key
```

```
    and a.tracker_key=c.tracker_key
    and a.pf_key=d.pf_key
    and a.project_key=e.project_key
    and
a.tracker_key=ffd.tracker_key
    --and
e.title='TestMultiFunction'
    and ffd.name = 'Select User'
    and c.title in ('TrackerN')
    and
get_artifact_user_value(ff.flex_fields,ffd.flex
=''
    --and
date(a.effective_from)<=date(now())
and
date(a.effective_till)>'2012-04-01'
group by 1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Use case 7: Filter Text flex fields, case-insensitive search

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: 'Text102'
- Value contains: 'soft'
- Tracker name: 'Tracker101'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system
- The upper() function can be replaced with the lower() function appropriately for case-insensitive search

Sample query

```
select
        b.date_of_trans as Date,
    e.title as Project,
    c.title as Tracker,
    d.title as Planing_folder,
    'P'||a.priority as Priority,

get_artifact_text_value(ff.flex_fields,ffd.flex
as TextFlexField,
    count(distinct a.artifact_key)
as TotalCounts
from
    artifact_transaction_fact a,
    date_dimension b,
    artifact_flex_fields ff,
    tracker_dimension c,
    pf_dimension d,
    project_dimension e ,
    flex_field_dimension ffd
where
    a.trans_date_key=b.date_key
    and ff.artifact_flex_fields_key
= a.flex_fields_key
    and a.tracker_key=c.tracker_key
    and a.pf_key=d.pf_key
    and a.project_key=e.project_key
```

```
      and
a.tracker_key=ffd.tracker_key
   and
UPPER(e.title)=UPPER('Project SCD-2
Test')
   and
UPPER(ffd.name)=UPPER('Text102')
   and UPPER(c.title) =
UPPER('Tracker101')
   and
UPPER(get_artifact_text_value(ff.flex_fields,ff
like UPPER('soft%')

group by 1,2,3,4,5,6
order by 1,2,3,4,5,6;
```

Query Oracle datamart based on flex fields

Create query scripts to query and extract the required information from the Oracle datamart. Make sure these scripts are available to any user (including users with read-only permission) who wants to execute these scripts. This topic lists the functions and sample queries for a few specific use cases.

☞ **Important:**

- Log into TeamForge as a Reporting user and run these queries. Note that this is a one-time process.
- Use the following flex field functions by joining the `flex_field_dimension`, `artifact_flex_field` and `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` tables as it depends on the *XML data* and *flex field key* generated by the platform.

- **Function for Date flex fields**

Input arguments	• Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables) • Flex field key (derived automatically from `flex_field_dimension` table and other tables)
Output arguments	Date flex field values

```
create or replace FUNCTION get_artifact_date_value( artifact_xml IN
 XMLTYPE, field_key IN INTEGER )
   RETURN varchar2
   IS
 flex varchar2(4000);
   BEGIN
     SELECT x.val INTO flex FROM dual ,
     XMLTABLE ('/fields/field[@key=$keyalias]'
         PASSING artifact_xml,field_key as "keyalias"
         COLUMNS val VARCHAR2(4000) PATH '@val') x;
   RETURN(flex);
   END;
```

- **Function for Text flex fields**

Input arguments	• Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables)

- Flex field key (derived automatically from `flex_field_dimension` table and other tables)

Output arguments   Text flex field values

```
create or replace FUNCTION replacen( artifact_xml1 IN XMLTYPE)
   RETURN varchar2
   IS
   replaceval  varchar2(3000);
   BEGIN
   SELECT REPLACE((REPLACE(cast(artifact_xml1 as
 varchar2(3000)),'',''))'','') into replaceval  from dual;
   RETURN(replaceval);
   END;
create or replace FUNCTION get_artifact_text_value( artifact_xml IN
 XMLTYPE, field_key IN integer )
   RETURN varchar2
   IS flex varchar2(3200);
   fields_xml XMLTYPE;
   BEGIN
   SELECT XMLTYPE(replacen(artifact_xml)) INTO fields_xml FROM dual;
      SELECT x.val INTO flex FROM dual ,
      XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING fields_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(400) PATH '@val' ) x;
   RETURN(flex);
   END;
```

- **Function for Single-select flex fields**

  Input arguments
  - Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables)
  - Flex field key (derived automatically from `flex_field_dimension` table and other tables)

  Output arguments   Values selected or stored in single-select flex field

```
create or replace FUNCTION get_artifact_select_value( artifact_xml IN
 XMLTYPE, field_key IN INTEGER)
   RETURN VARCHAR2
   IS
    singleSelectValue VARCHAR2(100);
    flex INTEGER;
   BEGIN
      SELECT x.val INTO flex FROM dual ,
      XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(30) PATH '@val') x;
   SELECT  z.value INTO singleSelectValue from  FLEX_FIELD_VALUE_DIMENSION
 z  where z.FLEX_FIELD_VALUE_KEY=flex;
   RETURN(singleSelectValue);
   END;
```

- **Function for Multi-select flex fields**

  Input arguments
  - Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or)

artifact_daily_snapshot_fact and artifact_flex_field tables).

- Flex field key (derived automatically from flex_field_dimension table and other tables).
- Specific value(s) stored in the field or the keyword 'ALL' for retrieving all values stored in multi-select flex field.
- Logical conditional operator. Possible values are 'ALL' or 'ANY'. The argument value of 'ALL' performs a search equivalent to the 'AND' condition and the value of 'ANY' performs a search equivalent to the 'OR' condition.

Output arguments   Values selected or stored in multi-select flex field

```
-----Inner Function----
  create or replace FUNCTION get_artifact_multiselect_any(artifact_xml IN
 XMLTYPE, field_key IN INTEGER,selectedfields IN varchar2)
   RETURN varchar2
   IS
   flexs varchar2(300);
   arraylength integer default 1;
   loop1 integer default 1;
   loop2 integer default 1;
    i integer default 1;
    j integer default 1;
     v_array1 apex_application_global.vc_arr2;
     v_array2 apex_application_global.vc_arr2;
     v_array3 apex_application_global.vc_arr2;
     results varchar2(200) ;

   BEGIN
       SELECT x.val  INTO flexs FROM dual ,
        XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(4000) PATH '@val') x;
   v_array1 := apex_util.string_to_table(flexs,',');

   FOR  loop1 in 1..v_array1.count LOOP
     select value into v_array2(loop1) from  flex_field_value_dimension
 where flex_field_value_key=v_array1(loop1);
   END LOOP;

 v_array3 := apex_util.string_to_table(selectedfields,',');


 FOR   loop2 in 1..1 LOOP
               IF (v_array3(loop2)='ALL') THEN
                              FOR i in 1..v_array2.count LOOP
                                 IF LENGTH(results)!=0 THEN
                                               results:=results
 ||','||v_array2(i);
                                             ELSE

 results:=v_array2(i);
                                             END IF;

                         END LOOP;
           EXIT;
```

```
        ELSE

                                        FOR  i in 1..v_array2.count LOOP
                                          FOR  j in 1..v_array3.count LOOP
                                            IF
 (v_array3(j)=v_array2(i)) THEN

                                              IF LENGTH(results)!=0
 THEN

                                                results:=results
 ||','||v_array3(j);

                                              ELSE

 results:=v_array3(j);

                                              END IF;
                                              EXIT;
                                            END IF;

                                          END LOOP;
                                        END LOOP;
                  END IF;
  END LOOP;

 RETURN(results);
          END;
/

-----End Inner function-----

   create or replace FUNCTION get_artifact_multiselect_value(artifact_xml
 IN XMLTYPE, field_key IN INTEGER,selectedfields IN varchar2,condition IN
 varchar2)
   RETURN varchar2
   IS
   flexs varchar2(300);
   loop1 integer default 1;
   loop2 integer default 1;
    i integer default 1;
    j integer default 1;
     v_array1 apex_application_global.vc_arr2;
     v_array2 apex_application_global.vc_arr2;
     v_array3 apex_application_global.vc_arr2;
     results varchar2(200) ;
     filteredField varchar2(200) ;
  valcount integer default 0;

   BEGIN
       SELECT x.val  INTO flexs FROM dual ,
        XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(4000) PATH '@val') x;
   v_array1 := apex_util.string_to_table(flexs,',');

   FOR  loop1 in 1..v_array1.count LOOP
     select value into v_array2(loop1) from  flex_field_value_dimension
 where flex_field_value_key=v_array1(loop1);
   END LOOP;

 v_array3 := apex_util.string_to_table(selectedfields,',');

 IF (UPPER(condition)=UPPER('ALL')) THEN
           FOR   loop2 in 1..1 LOOP
             IF (v_array3(loop2)='ALL') THEN
                 FOR i in 1..v_array2.count LOOP
```

```
                               IF LENGTH(results)!=0 THEN
                                results:=results ||','||v_array2(i);
                               ELSE
                                 results:=v_array2(i);
                               END IF;

                              END LOOP;
                              filteredField:=results;
                              EXIT;

                     ELSE

                         FOR  i in 1..v_array3.count LOOP
                           FOR  j in 1..v_array2.count LOOP
                             IF (v_array3(i)=v_array2(j)) THEN
                             IF LENGTH(results)!=0 THEN
                              results:=results ||','||v_array3(j);
                              valcount:=valcount+1;
                              ELSE
                              results:=v_array3(j);
                              valcount:=valcount+1;
                             END IF;
                             EXIT;
                            END IF;

                            END LOOP;
                         END LOOP;
                         IF (valcount=v_array3.count) THEN
                         filteredField:=results;
                         ELSE
                         filteredField:='';
                         END IF;
                     END IF;
                 END LOOP;



   ELSE
     select
 get_artifact_multiselect_any(artifact_xml,field_key,selectedfields) into
 results from dual;
     IF results IS NOT NULL THEN
      filteredField:=results;
     ELSE
      filteredField:='';
     END IF;
    END IF;
 filteredField:=TRIM(TRAILING ',' FROM filteredField);

 RETURN(filteredField);
            END;
```

- **Function for Multi-user flex fields**

  Input arguments

  - Flex field XML (derived automatically by joining the `artifact_transaction_fact` (or) `artifact_daily_snapshot_fact` and `artifact_flex_field` tables).
  - Flex field key (derived automatically from `flex_field_dimension` table and other tables).

- Specific value(s) stored in the field or the keyword 'ALL' for retrieving all values stored in multi-select flex field.
- Logical conditional operator. Possible values are 'ALL' or 'ANY'. The argument value of 'ALL' performs a search equivalent to the 'AND' condition and the value of 'ANY' performs a search equivalent to the 'OR' condition.

Output arguments   Values selected or stored in multi-user flex field

```
------Inner Function-----
  create or replace FUNCTION get_artifact_user_any(artifact_xml IN
 XMLTYPE, field_key IN integer,selectedfields IN varchar2)
   RETURN varchar2
   IS
   flexs varchar2(300);
   arraylength integer default 1;
   loop1 integer default 1;
   loop2 integer default 1;
    i integer default 1;
    j integer default 1;
     v_array1 apex_application_global.vc_arr2;
     v_array2 apex_application_global.vc_arr2;
     v_array3 apex_application_global.vc_arr2;
     results varchar2(200) ;

   BEGIN
       SELECT x.val  INTO flexs FROM dual ,
        XMLTABLE ('/fields/field[@key=$keyalias]'
          PASSING artifact_xml,field_key as "keyalias"
          COLUMNS val VARCHAR2(4000) PATH '@val') x;
   v_array1 := apex_util.string_to_table(flexs,',');

   FOR  loop1 in 1..v_array1.count LOOP
     select full_name into v_array2(loop1) from  user_dimension where
user_key=v_array1(loop1);
   END LOOP;

 v_array3 := apex_util.string_to_table(selectedfields,',');


 FOR   loop2 in 1..v_array3.count LOOP
             IF (v_array3(loop2)='ALL') THEN
                            FOR i in 1..v_array2.count LOOP
                               IF LENGTH(results)!=0 THEN
                                              results:=results
||','||v_array2(i);
                                       ELSE

results:=v_array2(i);
                                        END IF;


                        END LOOP;
          EXIT;

        ELSE


                             FOR  i in 1..v_array2.count LOOP
                                FOR  j in 1..v_array3.count LOOP
```

```
                                                        IF
  (v_array3(j)=v_array2(i)) THEN
                                                           IF LENGTH(results)!=0
  THEN
                                                              results:=results
  ||','||v_array3(j);
                                                           ELSE

  results:=v_array3(j);
                                                           END IF;
                                                           EXIT;
                                                        END IF;

                                                  END LOOP;
                                            END LOOP;
                  END IF;
  END LOOP;

 RETURN(results);
            END;

------End Inner Function----

create or replace FUNCTION get_artifact_user_value(artifact_xml IN
 XMLTYPE, field_key IN INTEGER,selectedfields IN varchar2,condition IN
 varchar2)
   RETURN varchar2
   IS
   flexs varchar2(300);
   loop1 integer default 1;
   loop2 integer default 1;
    i integer default 1;
    j integer default 1;
     v_array1 apex_application_global.vc_arr2;
     v_array2 apex_application_global.vc_arr2;
     v_array3 apex_application_global.vc_arr2;
     results varchar2(200) ;
     filteredField varchar2(200) ;
  valcount integer default 0;

   BEGIN
       SELECT x.val  INTO flexs FROM dual ,
        XMLTABLE ('/fields/field[@key=$keyalias]'
           PASSING artifact_xml,field_key as "keyalias"
           COLUMNS val VARCHAR2(4000) PATH '@val') x;
   v_array1 := apex_util.string_to_table(flexs,',');

   FOR  loop1 in 1..v_array1.count LOOP
     select full_name into v_array2(loop1) from  user_dimension where
 user_key=v_array1(loop1);
   END LOOP;

 v_array3 := apex_util.string_to_table(selectedfields,',');

 IF (UPPER(condition)=UPPER('ALL')) THEN
            FOR   loop2 in 1..1 LOOP
             IF (v_array3(loop2)='ALL') THEN
                 FOR i in 1..v_array2.count LOOP
                    IF LENGTH(results)!=0 THEN
                     results:=results ||','||v_array2(i);
                    ELSE
                      results:=v_array2(i);
                    END IF;
```

```
                    END LOOP;
                    filteredField:=results;
                    EXIT;

          ELSE

              FOR  i in 1..v_array3.count LOOP
                FOR  j in 1..v_array2.count LOOP
                  IF (v_array3(i)=v_array2(j)) THEN
                 IF LENGTH(results)!=0 THEN
                  results:=results ||','||v_array3(j);
                  valcount:=valcount+1;
                  ELSE
                  results:=v_array3(j);
                  valcount:=valcount+1;
                 END IF;
                  EXIT;
                END IF;

                 END LOOP;
              END LOOP;
              IF (valcount=v_array3.count) THEN
              filteredField:=results;
              ELSE
              filteredField:='';
              END IF;
          END IF;
      END LOOP;



  ELSE
    select get_artifact_user_any(artifact_xml,field_key,selectedfields)
 into results from dual;
    IF results IS NOT NULL THEN
     filteredField:=results;
    ELSE
     filteredField:='';
    END IF;
  END IF;
filteredField:=TRIM(TRAILING ',' FROM filteredField);

RETURN(filteredField);
          END;
```

Sample use cases and queries

> ☞ **Note:**
> - Flex field Name, Value, Tracker title and so on that are used in the filtering conditions are case-sensitive.
> - See *Case 7* below for building a case-insensitive search using upper() or lower() functions.

Use case 1: Filter Date and Text flex fields	Suppose you want to retrieve data based on the following assumptions:
	- Date flex field named 'CreatedDate' that has a value of '2015-07-07 00:00:00'
	- Text flex field named 'TEXT' with part of its value containing the pattern 'cre'
	- Tracker title is 'TestDateTracker'

> ☞ **Note:**
>
> - You can change the field name, tracker title and values based on the data in your system
> - The date passed as input should be of the format "YYYY-MM-DD HH24:MI:SS"

Sample query

```
SELECT t.date_of_trans  "Date",
       t.Project  ,
       t.Tracker  ,
       t.PlaningFolder  ,
       t.Priority,
       count(distinct
 t.artifact_key) "TotalArtifacts"
  FROM (SELECT
 b.date_of_trans,
                       e.title  AS
 Project,
    c.title AS Tracker,
                       d.title AS
 PlaningFolder,
                       'P'||
a.priority AS Priority,

 a.artifact_key
 FROM  artifact_transaction_fact
 a inner join date_dimension b on
 (a.trans_date_key=b.date_key)
                      inner join
 artifact_flex_fields ff   on
 (ff.artifact_flex_fields_key =
 a.flex_fields_key)
                      inner
 join   tracker_dimension c on
 (a.tracker_key=c.tracker_key)
            inner join  pf_dimension
 d on   (a.pf_key=d.pf_key)
            inner join
 project_dimension e on
 (a.project_key=e.project_key)
                 inner join
 flex_field_dimension ffd on
 (a.tracker_key=ffd.tracker_key)
                 and
 ffd.name='Date'
                 and
 c.title='TestDateTracker'
            and
 get_artifact_date_value(ff.flex_fields,ffd.flex
 00:00:00'

UNION


 SELECT              b.date_of_trans,
                        e.title
 AS Project,
      c.title AS Tracker,
                        d.title
 AS PlaningFolder,
                        'P'||
a.priority,
```

```
a.artifact_key
FROM  artifact_transaction_fact
a inner join date_dimension b on
(a.trans_date_key=b.date_key)
                    inner join
 artifact_flex_fields ff  on
(ff.artifact_flex_fields_key =
a.flex_fields_key)
                    inner
join  tracker_dimension c on
(a.tracker_key=c.tracker_key)
          inner join  pf_dimension
d on  (a.pf_key=d.pf_key)
                 inner join
 project_dimension e on
(a.project_key=e.project_key)
                 inner join
 flex_field_dimension ffd on
(a.tracker_key=ffd.tracker_key)
                  and
ffd.name='TEXT'
and c.title='TrackerTest'
          and
get_artifact_text_value(ff.flex_fields,ffd.flex
LIKE 'cre%'
          and d.id='plan1004' ) t


GROUP BY
 t.date_of_trans,t.Project,t.Tracker,t.PlaningFo
ORDER BY
 t.date_of_trans,t.Project,t.Tracker,t.PlaningFo
```

Use case 2: Filter Single-select flex fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: 'FSS1'
- Value selected or stored: 'S1'
- Tracker title: 'FTracker'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system

Sample query

```
SELECT    b.date_of_trans  "Date",
                   e.title
 "Project",
                   c.title
 "Tracker",
                   d.title
 "Planingfolder",
                   'P'||a.priority
 "Priority",
                   count( distinct
a.artifact_key) "TotalArtifacts"
FROM  artifact_transaction_fact
a inner join date_dimension b on
(a.trans_date_key=b.date_key)
                            inner
join  artifact_flex_fields ff  on
```

```
                      (ff.artifact_flex_fields_key =
 a.flex_fields_key)
                                          inner
 join   tracker_dimension c on
 (a.tracker_key=c.tracker_key)
          inner join  pf_dimension d on
   (a.pf_key=d.pf_key)
         inner join  project_dimension
 e on  (a.project_key=e.project_key)

                       inner join
 flex_field_dimension ffd on
 (a.tracker_key=ffd.tracker_key)
                        and
 ffd.name='FSS1'
                        and
 c.title='FTracker' and
 get_artifact_select_value(ff.flex_fields,ffd.fl
 --where a.effective_from <=
 sysdate and a.effective_till >
 to_date('2012-04-01', 'YYYY-MM-DD')
 GROUP BY
 b.date_of_trans,e.title,c.title,d.title,a.prior
 ORDER BY
 b.date_of_trans,e.title,c.title,d.title,a.prior
```

Use case 3: Filter Text flex fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: 'TEXT'
- Value contains: 'cre
- Tracker name: 'TrackerTest'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system

Sample query

```
SELECT            b.date_of_trans
 "Date",
                         e.title
 "Project",
     c.title  "Tracker",
                         d.title
 "Planingfolder",
                         'P'||
 a.priority  "Priority",

 count( distinct a.artifact_key)
 "TotalArtifacts"
 FROM  artifact_transaction_fact
 a inner join date_dimension b on
 (a.trans_date_key=b.date_key)
                      inner join
  artifact_flex_fields ff  on
 (ff.artifact_flex_fields_key =
 a.flex_fields_key)
                      inner
 join   tracker_dimension c on
 (a.tracker_key=c.tracker_key)
          inner join  pf_dimension
 d on  (a.pf_key=d.pf_key)
```

```
                  inner join
 project_dimension e on
(a.project_key=e.project_key)
                  inner join
 flex_field_dimension ffd on
(a.tracker_key=ffd.tracker_key)
                  and
ffd.name='TEXT'
and c.title='TrackerTest'
          and
get_artifact_text_value(ff.flex_fields,ffd.flex
LIKE 'cre%'
          and d.id='plan1004'
--where a.effective_from <=
 sysdate and a.effective_till >
 to_date('2012-04-01', 'YYYY-MM-DD')
GROUP BY
 b.date_of_trans,e.title,c.title,d.title,a.prior
ORDER BY
 b.date_of_trans,e.title,c.title,d.title,a.prior
```

Use case 4: Filter Date flex fields

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: 'Date'
- Value: '2015-08-12 00:00:00'
- Tracker name: 'TestDateTracker'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system

Sample query

```
SELECT           b.date_of_trans
 "Date",
                    e.title
 "Project",
   c.title  "Tracker",
                    d.title
 "Planingfolder",
                    'P'||
a.priority  "Priority",

 count( distinct a.artifact_key)
 "TotalArtifacts"
FROM   artifact_transaction_fact
a inner join date_dimension b on
(a.trans_date_key=b.date_key)
                  inner join
 artifact_flex_fields ff  on
(ff.artifact_flex_fields_key =
a.flex_fields_key)
                  inner
join   tracker_dimension c on
(a.tracker_key=c.tracker_key)
         inner join  pf_dimension
d on  (a.pf_key=d.pf_key)
         inner join
project_dimension e on
(a.project_key=e.project_key)
```

```
                    inner join
flex_field_dimension ffd on
(a.tracker_key=ffd.tracker_key)
                    and
ffd.name='Date'
                    and
c.title='TestDateTracker'
          and
get_artifact_date_value(ff.flex_fields,ffd.flex
00:00:00'
--where a.effective_from <=
sysdate and a.effective_till >
to_date('2012-04-01', 'YYYY-MM-DD')
GROUP BY
b.date_of_trans,e.title,c.title,d.title,a.prior
ORDER BY
b.date_of_trans,e.title,c.title,d.title,a.prior
```

Use case 5: Multi-select flex fields

Suppose you want to retrieve data based on the following assumptions:

- Multi-select flex field name: 'Multiselect'
- Value: 'M12,M11'
- Tracker name: 'TrackerTest'
- Conditional parameter: 'ALL'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system
- Flex field name, value and tracker title that are used in the SQL filter conditions are case sensitive
- If you want to select all the values in User flex field, then pass 'ALL' as the conditional parameter
- If you want to select any value, then pass 'ANY' as the conditional parameter

Sample query

```
SELECT          b.date_of_trans
"Date",

e.title "Project",
                    c.title
"Tracker",

d.title  "Planingfolder",
                        'P'||
a.priority  "Priority",
            count(distinct
a.artifact_key)
FROM
artifact_transaction_fact a
inner join date_dimension b on
(a.trans_date_key=b.date_key)
            inner join
artifact_flex_fields ff  on
(ff.artifact_flex_fields_key =
a.flex_fields_key)
```

```
             inner join
 tracker_dimension c on
(a.tracker_key=c.tracker_key)
        inner join  pf_dimension d
on  (a.pf_key=d.pf_key)
        inner join
project_dimension e on
(a.project_key=e.project_key)
             inner join
 flex_field_dimension ffd on
(a.tracker_key=ffd.tracker_key)
             and
 ffd.name='Multiselect'
             and
 c.title='TrackerTest'
             and
get_artifact_multiselect_value(ff.flex_fields,ff
 IS NOT NULL
GROUP BY
 b.date_of_trans,e.title,c.title,d.title,a.prior
ORDER BY
 b.date_of_trans,e.title,c.title,d.title,a.prior
```

Use case 6: Filter User flex fields

Suppose you want to retrieve data based on the following assumptions:

- User flex field name: 'Select User'
- Value: 'user1,user2'
- Tracker name: 'TrackerTestUser'
- Conditional parameter: 'ALL'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system
- Flex field name, value and tracker title that are used in the SQL filter conditions are case sensitive
- If you want to select all the values in User flex field, then pass 'ALL' as the conditional parameter
- If you want to select any value, then pass 'ANY' as the conditional parameter

Sample query

```
 SELECT
        b.date_of_trans  "Date",
        e.title "Project",
  c.title  "Tracker",
        d.title  "Planingfolder",
        'P'||a.priority
"Priority",
        count( distinct
a.artifact_key) "TotalArtifacts"
FROM  artifact_transaction_fact
a inner join date_dimension b on
(a.trans_date_key=b.date_key)
      inner join
artifact_flex_fields ff  on
(ff.artifact_flex_fields_key =
a.flex_fields_key)
```

```
        inner join
tracker_dimension c on
(a.tracker_key=c.tracker_key)
 inner join  pf_dimension d on
(a.pf_key=d.pf_key)
 inner join  project_dimension e on
 (a.project_key=e.project_key)
        inner join
flex_field_dimension ffd on
(a.tracker_key=ffd.tracker_key)
        and ffd.name='Select User'
        and
c.title='TrackerTestUser'
        and
get_artifact_user_value(ff.flex_fields,ffd.flex
IS NOT NULL
--where a.effective_from <=
sysdate and a.effective_till >
to_date('2012-04-01', 'YYYY-MM-DD')
GROUP BY
b.date_of_trans,e.title,c.title,d.title,a.prior
ORDER BY
b.date_of_trans,e.title,c.title,d.title,a.prior
```

Use case 7: Filter Text flex fields, case-insensitive search

Suppose you want to retrieve data based on the following assumptions:

- Flex field name: 'TEXT'
- Value contains: 'cre'
- Tracker name: 'TestDateTracker'

☞ **Note:**

- You can change the field name, tracker title and values based on the data in your system
- The upper() function can be replaced with the lower() function appropriately for case-insensitive search

Sample query

```
SELECT
        b.date_of_trans  "Date",
        e.title "Project",
    c.title  "Tracker",
        d.title
"Planingfolder",
        'P'||a.priority
"Priority",
        count( distinct
a.artifact_key) "TotalArtifacts"
FROM  artifact_transaction_fact
a inner join date_dimension b on
(a.trans_date_key=b.date_key)
        inner join
artifact_flex_fields ff  on
(ff.artifact_flex_fields_key =
a.flex_fields_key)
        inner join
tracker_dimension c on
(a.tracker_key=c.tracker_key)
    inner join  pf_dimension d on
(a.pf_key=d.pf_key)
```

```
      inner join  project_dimension
e on  (a.project_key=e.project_key)

            inner join
flex_field_dimension ffd on
(a.tracker_key=ffd.tracker_key)
            and
UPPER(ffd.name)=UPPER('TEXT')
            and
upper(c.title)=upper('TestDateTracker')
      and
upper(get_artifact_text_value(ff.flex_fields,ff
like upper('cre%')
--where a.effective_from <=
sysdate and a.effective_till >
to_date('2012-04-01', 'YYYY-MM-DD')
GROUP BY
b.date_of_trans,e.title,c.title,d.title,a.prior
ORDER BY
b.date_of_trans,e.title,c.title,d.title,a.prior
```

TrackerInitialJob - parallel processing

A parallel processing feature has been introduced in TeamForge 8.1 Patch 1 to improve the performance of the tracker initial load job.

A performance observation was conducted on the tracker initial load job when fetching data from TeamForge and loading them to datamart. For more information on the performance observation, see *TrackerInitialJob performance observation*.

Based on the result, a parallel processing feature has been introduced to reduce the data processing time. For the observation results on parallel processing, see *here*.

The parallel processing includes the following steps:

1. In the *ETL_JAVA_OPTS* site option token available in the installer configuration directory (`/opt/collabnet/teamforge-installer/<Product Version>/conf`), change the JVM heap size by setting the `Xmx` value to `-Xmx2048m`.

    ☞ **Important:** To make this a permanent change, you must modify the above setting before installing the product.

If you do not want to make this a permanent change due to hardware resource constraint or for any other reason, then change the value in the `set-env.sh` file located in `runtime/conf` directory as below:

2. Open `set-env.sh` in `<teamforge-installed-base-dir>/runtime/conf` directory.

3. In *ETL_JVM_OPTS*, change the `Xmx` setting to `-Xmx2048m`.

4. Restart the CollabNet ETL service.

    ```
    sudo /etc/init.d/collabnet restart etl
    ```

    ⚠ **Warning:** If the heap size (Xmx) is not set to 2048 MB for the ETL service, there is every possibility of the ETL service throwing an exception pertaining to insufficient memory while running the tracker initial load job.

Upgrade CLI reports to the latest version on Red Hat or CentOS

Use these instructions to upgrade the CLI reports to the latest version.

Though life cycle metric reports are released as part of regular TeamForge product releases (including patch releases), new reports are, at times, made available to customers as and when they are ready. When a new report is available, you can upgrade your CLI reports to the latest version.

1. Log on to TeamForge application server as root.

2. Install the latest `ctf_clireports`.

```
yum install ctf_clireports
```

3. Run the `post-install.py` script.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

4. Run the following command and verify the CLI reports installation. Observe the version and release information.

```
rpm -qi ctf_clireports
```

Control project access

To control who can access your project, consider the purpose of your project and the appropriate type of user.

Control access by user role

Project administrators use existing global project roles or create and assign roles to project members to define what those project members can do in the project.

If you choose to use existing global project roles, you can quickly assign relevant roles to your project members. It is a good practice to create a new role in your project only when a suitable global project role is not available.

☞ **Note:** If your project is a subproject of any other project, your project may have inherited some roles from the parent project. To work with the properties and permissions associated with inherited roles, you must go to the parent project where those roles are specified.

Create a role

A role defines the applications that project members with that role can use, and the specific things project members can do in each application.

Any project administrator can create and assign a role.

☞ **Tip:** It is a good idea to check the existing global project roles via **Project Admin** > **Permissions** > **Roles** > **View: global project roles** before creating any new role in a project.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Permissions**.
3. On the **ROLES** tab, select **View: Roles Created For a Project**.
4. Click **Create**.
5. On the **Create Role** page, write a name and description for the role.
6. To let this role be automatically added to any private subprojects of this project, clear the **PREVENT INHERITANCE** option.

   By default, roles are inherited by public and gated subprojects, but not private subprojects.

   ☞ **Tip:** If this project is a subproject of any other project, you may already have inherited some roles.

7. To allow project members to be able to request this role, select **PROJECT MEMBERS CAN REQUEST THIS ROLE**.

   Project members can submit requests for requestable roles, which the project administrator can approve or reject.

   ☞ **Tip:** Select **GRANT AUTOMATICALLY ON REQUEST** to skip the need to approve or reject a request.

8. Click **Create**.
   The role is created. The **Edit Role** page appears.
9. For each application listed on the **Role Permissions** page, select the permissions and resources you want to make available to users with this role.

   ☞ **Tip:** You can specify the permissions for Binaries and integrated applications here too.

   ☞ **Note:** You can specify access to individual top-level folders, but not to specific subfolders. However, in the case of documents, you can specify access to individual subfolders as well.

10. Click **Save**.

The role is created. You can assign it to project members at any time.

☞ **Tip:** Based on the earlier settings, a project member may be able to submit a request for the role.

Create a project administrator role
The project administrator is responsible for managing the project's users and roles.

The project creator is assigned the Founder Project Admin role, a special role granting all project and application administration permissions for the project. You can transfer the Founder Project Admin role to another user.

If the project creator is a CollabNet TeamForge administrator, no Founder Project Admin is created.

☞ **Note:** By default, project adminstrators do not have application administration permissions, such as Tracker Admin or Task Admin. Application administration permissions can be included in a project administrator role, but must be assigned separately.

1. On the **Role Permissions** page, select **Project Admin Permissions**.

   ☞ **Note:** If this is an inherited role, you can not edit the permissions associated with it. You can edit the project members and user groups to whom this inherited role is assigned.

   • If you want the role to contain only the project administrator permissions to manage users and roles, **Project Admin Permissions** is all you need to select.
   • If you want the role to contain additional application administration or other permissions, check the additional permissions.

2. Click **Save**.

All project members assigned this role have project administrator permissions to manage the project's users and roles.
Change a role
If users need to do things that are not allowed by a role you have assigned to them, you may need to change the permissions associated with that role.

When you edit a role, all project members with that role get the updated permissions automatically.

☞ **Note:** If your project is a subproject of any other project, your project may have inherited some roles from the parent project. To work with the properties and permissions associated with inherited roles, you must go to the parent project where those roles are specified.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin** menu, click **Permissions**.
3. From the list of roles created for this project, click the role you want to edit.

   ☞ **Note:** You can assign a role to direct project members and user groups of the project regardless of whether the role belongs directly to this project or is inherited from a parent project.

4. On the **Role** page, make the changes you need.

   • To edit the title or other role details, click **Edit**. Make the required changes and click **Update**.
   • To edit the role's permissions, choose an application from the left side of the page and select or deselect permissions and resources.
   • To edit the project members to whom the role is assigned, click **Assigned Project Members**.
   • To edit the user groups to whom the role is assigned, click **Assigned Groups**.

5. Click **Save**.

The role is modified.
Give roles to a project member
A project member can have any number of roles. As project administrator, you must assign each project member's roles.

Permissions are cumulative. The project member has all of the access permissions allowed by all of the assigned roles, plus any permissions that may have been assigned globally using application permissions.

☞ **Note:** If your project is a subproject of any other project, you may have inherited some roles, project members or user groups from the parent project. You can assign any role to any project member, regardless of whether it's a global project role, or the role belongs directly to the project, or is inherited from a parent project.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin** menu, click **Permissions**, then click the **User-Role Matrix** tab.

   Observe the users listed on the left and all the available roles (global, direct and inherited) on the right.

   ☞ **Note:** Users can be assigned global, direct and inherited roles. Inherited users have all the permissions that they have in the parent project.

   • If you assign another inherited role to an inherited user, that user gets both the roles.
   • If you assign a role to a user in a project, that user becomes a current member in that project.

3. Select roles for each project member.

   ☞ **Remember:** A user's license type also influences what the user can see and do on your site. A user's license type supersedes any role assignments.

   • An ALM license enables the user who holds it to use the full range of TeamForge features: both the core source-code management tools and the extended application life cycle management functionality.
   • An SCM license enables the user who holds it to use the core TeamForge source-code management tools.

   Ask your site administrator how many licenses of each kind are available for your users.

4. Click **Save**.

The roles are now assigned to each project member.

Give a role to multiple project members

A role can be assigned to many users at once.

The user-role matrix provides a convenient way to add project members to a role, but it can become unwieldy if the project has a large number of users or roles. When that is the case, try assigning roles to multiple users at one time.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin** menu, click **Permissions**.

3. Click the name of the role that you want to assign to project members.

4. On the **Role Permissions** page, click the **Assigned Project Members** tab.
   The **Assigned Project Members** page shows all users who currently have the role.

   ☞ **Tip:** You can assign the role only to the direct project members of the project.

   ☞ **Remember:** A user's license type also influences what the user can see and do on your site. A user's license type supersedes any role assignments.

   • An ALM license enables the user who holds it to use the full range of TeamForge features: both the core source-code management tools and the extended application life cycle management functionality.
   • An SCM license enables the user who holds it to use the core TeamForge source-code management tools.

   Ask your site administrator how many licenses of each kind are available for your users.

5. Click **Add** and use the **Find a User** window to move your desired users into the **Selected Users** list.

   ☞ **Tip:** You can search by full or partial user name or full name to find the right project members.

6. Click **OK**.

The project members are now assigned the role.

Handle a role request from a project member

A project member in CollabNet TeamForge can ask to be granted a role. As the project administrator, it's up to you to approve or reject such requests.

When a CollabNet TeamForge project member submits a request for a role in a project, the request is placed in the **User Membership** section of the **Project Administration** page, pending approval by a project administrator. The request is also displayed in the **Items Pending My Approval** section of each project administrator's **My Page**.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin** menu, click **User Membership**, then click the **Pending Requests** tab.

3. Under **Role Requests**, select the user whose role request you want to approve or reject.

   • Click **Approve** to approve the request and assign the role to the user.
   • Click **Reject** to deny the request.

   ☞ **Tip:** To view the permissions granted via the role, click the role name, if required.

   ☞ **Note:** Before approving or rejecting a role request, you can check the roles already assigned to the user by clicking the user name and selecting the **Roles** tab.

The user receives an email notification when the request is approved or rejected.

Assign a global project role on request
As a project administrator you can edit a requestable global project role for your project. You can update the settings to immediately assign the role to a project member on request.

☞ **Note:** Only site administrators or restricted site administrators with "Role-Edit" permission can edit global project role details. Project administrators can only edit requestable global project role's grant settings for their projects.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin** menu, click **Permissions**.

3. From the list of global project roles, click the requestable global project role you want to edit.

4. On the **Edit Role** page, click **Edit**.

5. Select the **Grant Automatically on Request** option and click **Save**.
   Selecting this option enables the project member requesting the role to be assigned the same immediately, without waiting for an approval.

The role is modified for your project.

Assign roles to a user group
Enable multiple users to do something all at once by giving their group a role.

A user group can have any number of roles. Each member of the user group has all the access permissions allowed by all of the assigned roles, plus any permissions that may have been assigned by other methods, such as application permissions or individually assigned roles.

Roles and the associated permissions can be inherited. If your project is a subproject of any other project, you may have inherited some roles or user groups. You can assign any inherited role to any user group.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin** menu, click **Permissions**, then click the **Group-Role Matrix** tab.

   Observe the user groups listed on the left and all the available roles (global, direct and inherited) on the right.

   ☞ **Note:** User groups can be assigned global, direct and inherited roles. The inherited user groups have all the permissions that they have in the parent project.

      • If you assign another inherited role to an inherited group, the members of that group get both the roles.
      • If you assign a role to a user group in a project, that user group becomes a direct user group in that project.

3. Select the roles you want for each user group and click **Save**.

   ☞ **Restriction:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

The roles are now assigned to each user group.

Assign user groups to a role

To manage permissions for a lot of groups or roles at once, try assigning user groups to roles.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin** menu, click **Permissions**.

3. To display existing global, direct or inherited roles, click the **Global project roles**, **Roles Created For a Project** or **Roles Inherited From Parent Project** option of **View**.

   ☞ **Note:** If your project is a subproject of another project, you may have inherited some roles or user groups. The inherited role details like role name, description and the source project name are listed.

   You can assign direct user groups to a global/direct/inherited role.

4. Click the name of the role that you want to assign to user groups.

5. On the **Role Permissions** page, click the **Assigned Groups** tab.
   The **Assigned Groups** page shows all user groups that are currently assigned the role.

   ☞ **Tip:** You can assign the role only to the direct user groups of the project.

6. Click **Add**.

7. Type some of the group's name in the **Name (search)** box and click **Find**.

8. In the **Find a user group** window, select the user groups that you want to add, and click **Finish**.

   ☞ **Restriction:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

The user groups are now assigned the role.

View users and user groups assigned to a role

Before adding a role to a project member or user groups, see all the users and user groups who are assigned that role through inheritance.

A user or user group can have any number of roles. Roles and the associated permissions can be inherited via project hierarchy or project groups.

1. Click **Project Admin** from the **Project Home** menu.

2. On the **Project Admin** menu, click **Permissions**.

3. In the **View** drop-down, select **Global project roles** and from the results, a specific role.

4. In the **Assigned Project Members** tab, click the **View** drop-down and make a selection.

   • **Direct Members** displays the users who are directly assigned the role in the project.
   • **Inherited Members** displays the users who inherit the role from parent projects as well as project groups.

5. In the **Assigned Groups** tab, click the **View** drop-down and make a selection.

   • **Direct User Groups** displays the user groups that are directly assigned the role in the project.
   • **Inherited User Groups** displays the user groups that inherit the role from parent projects as well as project groups.

The roles are now assigned to each user group.

Control access by user class

To avoid having to create and assign a lot of similar roles for individual users, give access to applications to whole classes of users whenever possible.

For each application (tasks, documents, file releases, trackers, and discussion forums), you can assign permissions globally based on user type.

For example, if you know that you want all project members to be able to view and submit to all project trackers, set the application's permissions to reflect this. You need to configure these settings only once.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin** page, configure your project access settings and click **Next**.

3. On the **Edit Default Access Application Permissions** page, click the + symbol to expand the section for which you want to assign permissions.

4. For each application and resource, choose a user type from the drop-down menus.

   • All users of the selected type will have the specified permissions: view, submit/view, or post/view.

   • For discussion forums and trackers, you can also specify submit or post permissions.

   ☞ **Note:** You can specify access to top-level folders, but not to specific subfolders.

   ☞ **Note:** If you want to control access to an application or resource that is not displayed on the **Edit Default Access Application Permissions** page, you can do so using role-based access control.

5. Click **Finish**.

Control access by project type

Projects can be open only to project members, open to everyone in the world, or something in between.

By default, all new projects are created as private projects, accessible only to project members. Your system administrator can change the default access level for new projects.

☞ **Important:** Users who do not have access to a project cannot see it on the **Home** page, in the **All Projects** list, or in search or reporting results.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **Permissions**, then click the **DEFAULT ACCESS PERMISSIONS** tab to see the project's current access setting.

3. Click **Edit**.

4. On the **Edit Default Access Permissions** page, select the kind of access you want to allow to your project.

   • **Private** - Project members only.
   • **Gated community** - Project members and unrestricted users.
   • **Public** - All users.

   ☞ **Note:** If the option is not available, your system administrator has prohibited changing the access levels of projects on the site.

Let users see other users' roles

You can enable some project members to view the roles assigned to other project members.

For example, if your project includes both core team members and consultants, you may want to restrict full visibility of user details to the core team members.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **Permissions**.

3. On the **ROLES** tab, click the role you want to edit.

   For example, if you have divided project members into a "Core Team" role and a "Consultant" role, click the "Core Team" role.

4. Under **Project Admin Permissions**, select **View User Membership**.

Now only project members who have the role you edited can see the roles held by other project members.

☞ **Note:** They still can't see the actual permissions included in those roles, unless they have Project Admin status.

Lock or unlock a project

To ensure that no changes occur in a project while you are collating or migrating project data, lock the project. You must have project administration permissions or be a site administrator to lock or unlock a project.

To lock or unlock a project in TeamForge, go to **Project Settings** and lock/unlock the project.

☞ **Note:** A locked project does not allow any member (including project administrators and site administrators) to make any changes to the project. Besides that, a locked project can not be set as the parent project for any other project and tasks like adding, editing or deleting integrated applications are also not allowed.

Click **PROJECT ADMIN** from the **Project Home** menu.

The project is locked or unlocked as desired. The lock icon 🔒 **Locked** appears on all the project pages while the project is locked.

☞ **Note:** If a locked project has an integrated application, for example, project tracker, all the project tracker pages are also non-editable while the project is locked. The user who has access permissions for the integrated application can only view the pages.

Control access to source code

It's a good idea to make sure your source code can only be used by people who have business with it.

You can control which users can view or commit source code. You can make these distinctions at the repository level or at the path level within a repository.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** page, click **Permissions**.
3. Click the role whose access to source code you want to control.

   ☞ **Tip:** If the appropriate role does not exist, you must create it first. See *Create a role* on page 240.

4. On the **Edit Role** page, click **Source Code**.
5. Under **Permissions for Specific Repositories**, select a repository in the project and specify this role's access to the repository as a whole.

   - To block users with this role from seeing this repository at all, select **No Access**.
   - To allow users with this role to see everything in the repository but not commit to it, select **View Only**.
   - To allow users with this role to commit to any path in the repository, select **View and Commit**.

6. If you want to control access to a specific path within the repository for users with this role, select **Path-based Permissions**.

   See *Who can access source code?* and *Wildcard-based access control and path-based permissions in TeamForge* for more information.

   a) Click **Add**.
   b) Specify a path in the repository.

      ☞ **Tip:** The path does not have to exist when you specify it. You can set a permission for a path that may be created later.

   c) Select **No Access**, **View Only**, or **View and Commit** for this path.

      ☞ **Note:**

      - If none of the available permissions (**View Only**, **View and Commit**, or **Path-based Permissions**) is selected for any repository, and none of the options under **Source Code Permissions** is selected, users with this role do not see the **Source Code** toolbar button.
      - If two paths have different permissions, the permissions on the lower-level path take effect. For example, consider a role that has "No Access" set for the path `/branches/version3/users`, but has "View and Commit" access to `/branches/version3/users/vijay`. Users with this role can:

        - Check code in and out of the `vijay` directory.
        - Click down through all the directories in that path, including `users`.

        Users with this role cannot check files in and out of the `users` directory or monitor commits to `users`.

7. Click **Save**.

☞ **Note:** You can also restrict the information that goes out with commit notification emails. See *Who can access source code?* on page 946.

Control HTML headers in hand-coded project pages

When a HTML page that you created in Microsoft Word or Frontpage looks strange, you may be able to fix it by suppressing HTML head content.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** section, select **Show custom web page** and then select **Preserve HTML head content**.
3. Click **Save**.

Pages created in Microsoft Word or Frontpage should render correctly.

Allow anonymous Subversion checkouts

To grant anonymous checkout access while restricting write access to a Subversion repository, set the project's default access permissions to public, provide Source Code View permission to all users, and limit other permissions to specific user classes.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Permissions**, then the **DEFAULT ACCESS PERMISSIONS** tab.
3. Under **Application Permissions**, choose **All users** from the drop-down for Source Code View permission.

   ☞ **Note:** You can give all users checkout access to all repositories or a specific repository.

4. For other application permissions, choose a user class based on your access requirement.

Users can now check out from a repository without entering a password.

Show or hide an application

To help users focus on the relevant parts of your project, choose which applications they can see in the **Project Home** menu.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu** page, click **Tools**.
3. Select the applications that you want to be visible to users in your project, and click **Save**.

Only the application buttons you have set to **VISIBLE** appear in the Project Home menu of the project page.

- Removing buttons from this menu does not remove the application from the site. Users with the appropriate permissions can still get to the hidden applications by other means, such as clicking a link in an email. The hidden application button will not appear.
- When an application button is set to **Visible**, but a given user does not have permission to use that application, that user still cannot see the application button. For example, if a user has a role that does not permit access to any source code repositories, that user does not see the **Source Code** toolbar button, even if the button is enabled for the project.
- If the Tasks tool is hidden for a project, the option to run reports on the Task tool is also hidden.
- If you create a project template from a project that has hidden applications, any project you create from that template will have the same applications hidden.

Who can post to discussions by email?

To help reduce the risk of spam or other mischief, you may need to limit the users who can post to your project's discussion forums by email.

To leverage the advantages of community collaboration, you should keep your forums as open as you can. However, some projects require tighter control over who can participate in discussions. TeamForge enables you to balance openness against privacy along a spectrum of choices.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Discussion Settings**.
3. Set the value of **EMAIL POSTING** to one of these choices (listed from most restrictive to least restrictive).

This option...	has this effect:
Allow only forum admins	Only users with the "discussion admin" permission can post by email.
Users with roles & permissions	Only registered users with the "topic post" permission can post by email. This is the default. (For discussions marked as private, this is the least restrictive setting possible.)
All logged in users	All users who are registered on the site can post by email.
Allow known email addresses only	Users can post by email only if they have explicitly been added to the monitoring list. (This can include people who are not registered site users.)
All site users and guests	Anyone can post by email.

If you select a value that's more liberal than the value your site administrator has set for the site as a whole, the site administrator's setting rules. For example, suppose the site administrator has provided that only users with the appropriate role ("Users with roles & permissions") can post by email. If your project requires extra security, you can choose to accept email only from forum administrators. However, you cannot accept email posts from a less restrictive category of users, such as "All logged in users."

Manage project membership

You can add or remove project members on their request or on your own initiative.

☞ **Note:** If your project is a subproject of any other project, your project may have inherited some roles from the parent project. To work with the properties and permissions associated with inherited roles, you must go to the parent project where those roles are specified.

Add users to a project

Before a person can work on a project, you have to make them a member of the project.

You can make any registered user on your TeamForge site a project member. You can assign roles to the user at the same time.

☞ **Note:** Project members, roles and the associated permissions can be inherited and reused in subprojects. If your project is a subproject of any other project, you may have inherited some roles, project members, or user groups.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **User Membership**.
3. Click **Add**.
4. On the **Add Users** page, find the users you want by one of these methods:

    • Under **FIND USERS**, filter the list of site users eligible to join this project. You can filter by full or partial name or user name.

    ☞ **Note:** Search text is not case-sensitive.

    • Browse the list of registered users on the site. Sort them by name, user name, email address or membership status.

    The inherited project members continue to hold the inherited roles with corresponding permissions, as specified in the source projects.

    ☞ **Note:** If a site has a great many users, you must filter them first to narrow down the list. This helps avoid slowing down the system.

5. Select the users you want to add.
6. Under **ASSIGN ROLES**, select the roles you want the users to have.

You can select any global project role, role created just for this project, or inherited role that is available.

☞ **Note:** If your project is a child project of another project, the members of the parent project become inherited members of your project. The user roles specified in the parent project are available in your project provided the role inheritance is not prevented. If you assign a role in your project to a user, that user becomes a direct member of your project.

☞ **Tip:** If you prefer, you can skip this step and assign roles later on the **Project Admin** > **Permissions** page. Note that using the **Assigned Project Members** page, you can assign roles only to the direct project members.

7. Save your changes.

- Click **Save** to return to the **User Membership** page.
- Click **Save and Add More** to keep adding users.

Remove a user from a project

When you remove a user from a project, all items such as tasks and tracker artifacts that were assigned to the user are re-assigned to None.

☞ **Note:** If your project is a subproject of any other project, your project may have inherited some users from the parent project. To remove an inherited user, you must go to the parent project where that user is a direct member.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **User Membership**.

3. From the list of current project members, select the user you want to remove and click **Remove**.

The user is now removed from the project.

Handle a request for project membership

A registered CollabNet TeamForge user can ask to be a member of a project. As the project administrator, it's up to you to approve or reject such requests.

When a CollabNet TeamForge user submits a request for project membership, the request is placed in the **User Membership** section of the **Project Administration** page, pending approval by a project administrator. The request is also displayed in the **Items Pending My Approval** section of each project administrator's **My Page**.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **User Membership**, then click the **PENDING REQUESTS** tab.

3. Under **Users Requesting to Join Project**, select the user you want to approve or reject.

- Click **Approve** to approve the request and add the user to the project.
- Click **Reject** to deny the request.

4. To view details about the user or add a comment before approving or rejecting the request, click the user name.

This is optional.

The user receives an email notification when the request is approved or rejected.

Handle a request to leave a project

A TeamForge user who wants to leave a project must submit a request. The project administrator can approve or reject the request.

A request to leave is placed in the **User Membership** section of the **Project Administration** page, pending approval by a project administrator. The request is also displayed in the **Items Pending My Approval** section of each project administrator's **My Page**.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **User Membership**, then click the **PENDING REQUESTS** tab.

3. Under **Users Requesting to Leave Project**, select the user whose request you want to approve or reject.

- Click **Approve** to approve the request and remove the user from the project.
- Click **Reject** to deny the request.

4. To view the user details or add a comment before approving or rejecting the request, click the user name.

    This is optional.

The user receives an email notification when the request is approved or rejected.

Create a project template

To make it easier to start projects, provide project templates based on existing projects.

A project template is used to populate new projects with the structure and configuration of the source project.It can also include the actual contents of the project it is based on, such as tasks, tracker items and documents.

When you create a project template, it is available only to projects on your own site. To make it available more widely, you can export the project template and share it with others. Ask your CollabNet representative for more information about doing this.

> 👉 **Note:** The access settings for a project template are the same as the access settings for the project on which the template is based. For example:
>
> - If you create a template from a project that has hidden applications, any project you create from that template will have the same applications hidden.
> - If you create a template from a project that is private, any project created from that template will also be private.

1. Set up the project that will serve as the basis for the new project template.

    > 👉 **Tip:** If you need a clean project template, you may want to create a new project specifically for this purpose.

2. Click **PROJECT ADMIN** from the **Project Home** menu.
3. On the **Project Settings** page, click **Create Project Template**.
4. Write a name and description for the template.

    > 👉 **Tip:** Consider the uses that you or other project managers might have for this project template, and include keywords in your description that are related to those uses.

5. Choose the items that you want to make available in new projects that are created from this template.

    - Every new project includes a **Trackers** component into which you can add new trackers; a **Documents** folder into which you can add more document folders and documents; and so on.
    - If you wish, you can include the actual trackers from the current project's **Trackers** component; the actual documents from the current project's **Documents** component, and so on.

6. Click **OK**.

The project template is created. The template is available to all users who have the required permissions to create new projects.

The template name and description appear on the **Template** tab of the **Projects** list, accessible from your personal navigation bar.

Categorize a project

Organizing projects by categories can help users find what they need on a site quickly and easily.

Project categories express the relationships among projects. If your project is in a category, it is visible to users browsing projects from the main **Project Categories** page.

Your project can be in one or more project categories.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click the **PROJECT CATEGORIZATION** tab.
    A list of all the categories that the project belongs to is displayed.
3. Click **Add**.
    The **Select Category** window shows all the available project categories.
4. If you find a project category that your project should be in, select that category and click **Add**.

Your project is now a member of the selected project category. You can repeat this process to add your project to any number of project categories.

Make other applications available to project members

You can make it easy for project members to use a wide variety of applications and sites from inside TeamForge.

Link an external application

To make an application or site outside of CollabNet TeamForge available in your project, create a linked application.

You can create as many linked applications per project as you need.

☞ **Tip:** Some sites employ page code that disables this feature. For example, some Google apps automatically generate user login data and append it to their URLs, which prevents them from opening in an iFrame.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **Tools**.
   A list of all the applications in the project is displayed on the **Project Tools** section.

3. Click **Add Tool**.

4. On the **Add Tool** page, enter a name for the linked application.
   This name will appear, with an icon you choose, in the **Project Home** menu.

5. Enter the server location or URL for the linked application.

6. If you are a TeamForge administrator, select whether you want to use single sign on for the linked application.

   • By default, only site administrators can edit (turn on/off) single sign on (SSO) for linked applications. However, you can set the site options token *ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON* to false to have both site and project administrators turn SSO on and off. For more information, see *ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON* on page 801.

   • If you use single sign on, TeamForge users can automatically log into the linked application.

   • If you do not use single sign on, users must log into the linked application using its native authentication system.

7. Click **Choose File** and select a gif, .jpg, or .png file to serve as the new icon for the linked application.

   Make the image 25 pixels wide and 20 pixels high.

   This icon appears with the application name in the **Project Home** menu.

8. Click **Save**.

A button for the linked application is added to your **Project Home** menu. Clicking it launches the application in the main TeamForge project window.

Change a linked application

When the use patterns of a linked application change, you may need to change the way the application integrates with CollabNet TeamForge .

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. From the list of applications, click the application you want to edit.
   The **Edit Tool** page is displayed.

3. Edit the name of the linked application, if required.

4. Edit the linked application URL, if required.

5. Enable or disable single sign on for the linked application.

   By default, only site administrators can edit (turn on/off) single sign on (SSO) for linked applications. However, you can set the site options token *ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON* to false to have both site and project administrators turn SSO on and off. For more information, see *ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON* on page 801.

6. Click **Choose File** and select a gif, .jpg, or .png file to serve as the new icon for the linked application.

   Make the image 25 pixels wide and 20 pixels high.

   This icon appears with the application name in the **Project Home** menu.

7. Click **Update**.

Integrate an external application into your project

To make an application or site outside of TeamForge available to your users seamlessly from inside your TeamForge project, bring it in as an integrated application.

If the application you want to use is not yet available for your project, ask your site administrator to set it up.

You can use as many integrated applications as you wish, after your site administrator has made them available.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Tools**.
   The **Project Tools** section displays the list of all the current applications in the project.
3. Click **Add Tool**.
4. On the **Add New Tool** page, select the desired application. and all the relevant paramters to configure the tool are displayed.

   a) If your site administrator has made it possible, specify a prefix for the resources created by the application you are setting up.
      A prefix enables TeamForge to provide handy links between objects managed by this application and other TeamForge objects. For example, if you are bringing in a blogging application:

      • You can connect a blog post with a TeamForge artifact using a special link called an "association."
      • For each blog post, you can give readers a simplified address, known as a "go URL."

      ☞ **Note:** For project-level associations and go URLs the application's prefix is permanent after you save it.

   b) Set any other configuration parameters you need and save your changes.

      • Click **Save** to return to the **Project Tools** page.

An icon for the integrated application is added to your **Project Home** menu. Clicking it launches the application in the main TeamForge project window.

Export an integrated application

TeamForge site administrators can export an integrated application's variables in XML format and have them available for editing.

1. Go to **My Workspace** > **Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. To export the integrated application, select the integrated application name.
4. From the **View Integrated Application** page, click **Export**.

The integrated application will be exported.

Remove an integrated application from your project

To stop making an application or site outside of CollabNet TeamForge available to your users from inside your CollabNet TeamForge projects seamlessly, disintegrate an application.

☞ **Note:** You may have integrated several external applications per project to maximize the integrated applications feature's utility. However, because each integrated application adds an icon to the project navigation bar, a large number of integrated applications can cause horizontal scrolling. Consider removing the integrated applications that are not in use.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Admin Menu**, click **Tools**.
3. To remove an integrated application from the list of tools displayed , click the relevant Delete icon..
   The selected integrated application is removed from the project.

The icon of the removed integrated application disappears from your **Project Home** menu.

☞ **Note:** Removing an integrated application, removes all associations to the integrated application too.

Add a parent project to your project

You can coordinate work among multiple projects; enable the user, user group and role inheritance by adding a common parent project to several projects.

☞ **Note:** A subproject can have only one parent. You can change or remove that parent at any time.

☞ **Note:** You must be a project administrator or site administrator to add/edit or remove the parent projects.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** page, click **Add Parent**.
3. Choose a parent project that it makes sense for your project to belong to, and click **Update**.

   ☞ **Note:** You cannot add a parent project for the 'Look' project as it is a special project in itself.

Separate a subproject from its parent project

When a subproject grows beyond its original scope, you may want to make it stand-alone project or move it to a different project hierarchy.

By removing the association with a parent project, you can manage the subproject as a separate project.

☞ **Note:** Only one parent project can be selected for a subproject. However, the parent project can be changed or removed, as required.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. On the **Project Settings** page, click **Edit Parent**.
3. Change the parent project as required.

   You can be a project administrator or a site administrator to change or remove a parent project. As a project administrator, you can remove or change a parent project only if you have administrator permissions for both the projects that are being linked.

   ☞ **Note:** A parent project can be removed or changed only when its members, user groups and roles are not in use in any other project. In other words, you can not remove/change a parent project while its members, user groups or roles are in use in any other project.

   • • On the **Choose a Parent Project** page, select the desired parent project and click **Update**.

      ☞ **Note:** If the project hierarchy exists, the project and its subprojects are moved only under the project from which members, user groups or roles are inherited. If project hierarchy does not exist and no inheritance is in use, the project is made a 'Root' project.

   • • Click **Remove Parent** in the **Project Settings** page to remove the association with parent project.
   • On the pop-up message box, click **OK**, if you wish to make the project a 'Root' project.

   The project hierarchy is changed or removed in accordance with role based access control and inheritance rules.

Require a wiki version comment

To help keep track of changes to a Wiki page, require users to record a comment about their changes when they save the page.

1. On any project page, click **PROJECT ADMIN** from the **My Page** menu.
2. On the **Project Settings** page, click **Wiki Settings**.
3. Select **REQUIRES VERSION COMMENT ON EDIT**.

Now users who edit a page in your project's Wiki must leave a note describing their changes.

Provision hardware for project members

You can use TeamForge Lab Management to provision physical or virtual hardware on which project members can remotely build and test their code.

☞ **Tip:** For more information about the new features in TeamForge Lab Management 2.2, see *this introductory video*. (You'll need a free open.collab.net membership to watch it.)

Set up hardware for your team to use

When you set up Lab Management, your team members can use TeamForge to access their own virtual machines for developing and testing.

1. Go to your Lab Management project.
2. Select **Project Admin** from the **Project Home** menu.
3. On the **Project Admin** menu, click **Tools**.
4. Click **Add Tool**.
5. Select **Other** from **Select Tool Type** drop-down.
6. Enter the display name as **Lab Mgmt**.
7. Select the **Show in Toolbar** checkbox.
8. Enter the server location or URL of your Lab Management server.
9. If you are a TeamForge administrator, select whether you want to use single sign-on for the linked application.

   • If you use single sign-on, TeamForge manages authentication for Lab Management, and users don't have to log into Lab Management after they have logged into TeamForge.
   • If you do not use single sign-on, users must log into Lab Management using its native authentication system.

10. Click **Browse** and select this icon for the linked application:

    This icon will appear with the name **Lab Mgmt** in the project navigation bar.

11. Click **Save**.

A **Lab Mgmt** button is added to your **Project Home** menu.. Clicking it launches the application in the main TeamForge project window.

Reallocate a system

To enable a project member to use a system, you must reallocate the system to that user.

You must be a Domain Administrator or Project Administrator to free a host that does not already belong to you.

1. Notify the user you're taking the system from through email or some other means, so that they don't lose any important work which they may not have saved on that system.

   The user whose system was deallocated will get an email after the deallocation, but it is a courtesy to notify the user beforehand.

2. On your project's **Project Home** page, click the host you want to reallocate.
3. Click **Free Host**.
   The host is now free and can be allocated by anyone in the project. TeamForge Lab Management emails the user that their system has been deallocated by an administrator.
4. To discourage other users from allocating it, put a note in the host's **Description** stating something like "This host reserved for Jane."
5. Notify the user who is going to allocate the new system that their system is now ready.

Edit a project

As a project administrator in TeamForge Lab Management, you can edit certain properties for your project.

1. On the project home page for the project that you wish to edit, click **Edit Project**.
2. Edit the following parameters:

Project Summary	A brief, one-line summary of the project.
Project Description	A more detailed explanation of the project.

Project MOTD (Message of the Day)	The Project MOTD is displayed to all users in your projects. This message also may display to users when they log in to client nodes in your project.
Project-specific host allocation time limits	In this section, project administrators can control how long users in their project can allocate hosts for. If you set this value to 0 (zero), there is no limit on the time a host can be allocated.
	If you reduce this value, users' hosts in your project may be deallocated. If deallocations will occur as a result of your lowering of the maximum allocation time, you will be warned which systems will be affected, and given a chance to change your mind.
Delete this Project/Undelete this Project	If your project is not deleted and has no hosts, you will be given the option to delete this project from TeamForge Lab Management. If your project is deleted, you will be given the option to undelete it.
	Note: only Domain Admins will be able to delete and undelete projects

Dictate which profiles can be used in your project

To control which operating system profiles the users in your project can build hosts with, adjust the settings in your profile library.

1. On the **Profile Library** page, click **Add/Remove Profiles**.
2. In the list of all potential profiles that you can add to your project, select the profiles you wish to be allowed in your project and click **Save Changes**.

   Your changes take effect immediately.

   ☞ **Note:** To revert any changes you have made, click **Cancel** or **Reset**.

Deliver a build

To get your software into users' hands, upload your build to the Project Build Library.

Get the PBL upload client

You must download the pbl.py script to transfer files into the Project Build Library.

Th PBL upload client is free, open-source, and freely modifiable and distributable.

Download the PBL upload client from *http://cubit.open.collab.net/pbl/*.

☞ **Note:** The API operations are fully documented, for users who might want to develop their own PBL upload client.

Upload a file

To upload a file, run the `pbl.py upload` command.

In this example, we upload a file named `Release.zip` from our local machine into the public area of the project `myproject`, in the directory `/foo/bar/baz/`.

☞ **Note:** If any part of the requested path does not already exist, `pbl.py` creates the intermediate directories.

Run the `pbl.py upload` command like this, substituting the correct values for your situation.

```
pbl.py upload --api-user=username --api-
key=713cdf90-2549-1350-80c3-2d0bcf9a1697 --api-url http://$external_host/
cubit_api/1 --project=myproject -t pub -r /foo/bar/baz -d "This is the
 description." /home/Release.zip
```

☞ **Tip:** Wildcards are accepted in the filename argument. If the file argument is a directory, or a wildcard which includes one or more directories, `pbl.py` recursively uploads all the subdirectories underneath the parent. All the files in the recursive upload get the same description.

Once this operation has completed, you can download this file from `https://$external_host/myproject/pub/foo/bar/baz/Release.zip`.

For other options to `pbl.py`, run `pbl.py help upload`.

Change the description of a file

You can change the description associated with a file or directory without changing the file itself or the md5 checksum of the file.

In this example, you have already authenticated and saved your user name and key credentials in your home directory.

Run the `pbl.py changedesc` command like this:

```
pbl.py changedesc -l http://$external_host/cubit_api/1 --project=myproject
 -t pub -r /foo/bar/baz/Release.zip -d "This is the new description"
```

Move a file

With the `pbl.py move` command, you can move files or directories within a project, or even between projects.

The syntax for this command is a bit different than the rest of the commands, because the other commands only operate on one project or file or directory at a time, and that is not the case the the move operation.

To move a file, run the `pbl.py move` command with these options.

In the simplest case, we move a file, or a directory and all its contents, from one name to another.

--srcproj projname	The name of the project the source file is located in.
--destproj projname	The name of the project to move the file to. If left blank, defaults to value of `--srcproj`.
--srcpath path	The path to the file or directory to move.
--destpath path	The destination path for the file or directory specified in `--srcpath`. Two important things to note about this option: • If you specify a path which does not exist, that path will be automatically created for you as part of the move. • If the `--destpath` parameter ends with a slash ("/"), the destination will be assumed to be a directory. If it does not end with a slash, the destination will be assumed to be a file. An example of this behavior is below. This is approximately how the UNIX "mv" command behaves.
--srctype {pub|priv}	The visibility type of the source file, either `pub` or `priv`.
--desttype {pub|priv}	The visibility type of the destination file, either `pub` or `priv`.
--force	If the destination file exists, the --force option must be used to replace it.

> ☞ **Note:** Because destpath does not end with a slash -- /foo/bar/baz/Release_old.zip -- the
> last component of the path is interpreted as "a file named 'Release_old.zip'".

```
pbl.py move -l http://$external_host/TeamForge Lab Management_api/1 --
srcprj=myproject --srctype=pub --srcpath=/foo/bar/baz/Release.zip --
destpath=/foo/bar/baz/Release_old.zip
```

To move a file from one project to another, and also change it from public to private, run the command like this.

> ☞ **Note:** Because destpath ends with a slash -- "/foo/bar/baz/archive/" -- the last component of the
> path is interpreted as "a directory named 'archive'".

```
pbl.py move -l http://$external_host/TeamForge Lab Management_api/1 --
srcprj=myproject --destproj=myproject_archive --srctype=pub --desttype=priv
 --srcpath=/foo/bar/baz/Release.zip --destpath=/foo/bar/baz/archive/
```

Allow clouds in a project

To enable project members to allocate hosts from a public cloud in your Lab Management site , you must turn on a
setting to allow the inclusion of public clouds in your project.

1. In **Administration** > **Projects**, click **Edit** for your project.

2. Turn on the **Allow Public Clouds** setting.

   If you want to be able to control the hosts your project members can select from, turn this setting OFF. In this
   case, your project members can only select systems from a cloud your project explicitly owns. Your project's
   Allowed Clouds page lists all the available clouds from which project members can allocate hosts.

> ☞ **Note:** If you think that your project needs more systems than those available from a cloud, you need to ask
> the cloud administrator for your site to increase this number.

Provision a Continuous Integration server from a Lab Management cloud

The Lab Management Cloud plugin enables Hudson and Jenkins to automatically create slaves from Lab Management
clouds.

Install the Lab Management Cloud plugin for Hudson and Jenkins

Get the plugin from openCollabNet and upload it using the Plugin Manager.

1. Download the labmanagement.hpi file from openCollabNet. (Please check back later for availability.)

2. In the Hudson or Jenkins Plugin Manager page, click the **Advanced** tab.

3. In the Upload Plugin section, browse to the location where you saved the .hpi file and click **Upload**.

In the Plugin Manager's **Installed** tab, you should see the Lab Management Cloud plugin enabled.

Configure the Lab Management Cloud plug in for Hudson and Jenkins

When you add a new cloud, you set up templates specifying details such as the host type and profile for the nodes that
will be provisioned from this cloud.

1. In the Cloud section of the Hudson or Jenkins configuration page, click **Add a new cloud** and select **CollabNet
   Lab Management**.

   Lab Management options are displayed. Here's an example:

2. Enter the URL of the Lab Management Manager node. For example, https://mgr.cloud.sp.collab.net/.

3. Provide the user name and password.

   This user must have access to the Lab Management project where new hosts will be allocated.

4. Enter the user's API key for the Lab Management web service.

   ☞ **Tip:** Copy and paste the key from the user's Lab Management home page.

5. Click **Test Connection** to make sure that the options you provided are valid.

6. Specify the Lab Management project where new hosts will be added.

7. Select a **Cloud** from which new hosts will be allocated.

   You can find the list of valid cloud names in the Lab Management Manager interface. For example:

8. Click **Add Template** to create a template that Hudson will start.

   a) Specify a name that characterizes the template.

   The **name** is used to identify the template and is displayed in various parts of the Hudson interface.

   b) For **Host Type**, select the type of hardware you want for the node.

   c) For **Size**, select the amount of resources -- memory, CPU, and disk -- assigned to this template.

   You can find the exact amount of resources for each size in the Lab Management Manager's **Clouds** > **Allocate From Cloud** page. For example:

   d) For **Profile**, select the Lab Management profile to be used for the template.

   e) Provide a list of **label**s separated by white spaces.

   👉 **Note:** The names and labels you provide for the templates are used to specify which jobs will use this cloud.

   f) Enter a value for **# of Executors**.

   This controls the number of concurrent builds that Hudson or Jenkins can perform. So the value affects the overall system load that might be incurred. A good value to start with is the number of processors on your system.

   When using Hudson or Jenkins in the master/slave mode, setting this value to 0 would prevent the master from doing any building on its own. Slaves may not have zero executors, but may be temporarily disabled using the button on the slave's status page.

9. Click **Save**.

In the **Manage Nodes** page, you'll see an option to provision a node from Lab Management.

Run a Hudson or Jenkins job on a host provisioned from Lab Management cloud

When you run a job on a host provisioned from a Lab Management cloud, use a template you defined earlier.

1. In the Hudson or Jenkins configuration page for the job, provide a name and description.
2. Select the **Restrict where this project can be run** option to tie the job to the Lab Management cloud.
3. Enter a label expression.

   To always run this job on a specific node, just specify its name. However, when several nodes could be available and you don't want to tie the job to a specific one, enter an expression based on the name and label values of a Lab Management cloud template you configured earlier.

4. Specify any other options you want and click **Save**.

   When a node is no longer used, it will be brought down and released.

Support project teams with CollabNet TeamForge

As a site administrator, you can help project managers and members of the site work together effectively throughout the application life cycle.

Move a site from CollabNet Enterprise Edition to CollabNet TeamForge 8.2

As a domain administrator, you will find some aspects of TeamForge similar to CollabNet Enterprise Edition and others quite different. Here are some steps you can take to ease the transition for yourself and your site's users.

👉 **Tip:** Training is available to help you manage the transition. See the *CollabNet training portal* for details.

1. Get a new password for TeamForge 8.2 site

   Your login name is migrated from CollabNet Enterprise Edition to TeamForge 8.2, but your password is not.

To create your new password, click **Forgot Your Password?** on the TeamForge 8.2 home page.

Your TeamForge site emails you a link you can use to set a new password.

2. Be aware of password policy differences.

TeamForge 8.2 uses a slightly different password policy from your CollabNet Enterprise Edition site. To review the site's password policy settings, click **View Password Policy** on the **Reset Password** page.

☞ **Note:** Users' password histories are not migrated. This means that if "Size of user's password history" was set in CollabNet Enterprise Edition, the history will be cleared on migration.

3. Site-wide broadcasts are a good way to keep your site's users up to date during the transition.

See *Communicate with all site users at once* on page 271.

4. Each of the CollabNet Enterprise Edition tools has its counterpart in CTF. Project information and resources work a little differently in each one.

For a detailed feature-to-feature comparison, see *Migrating tools from Enterprise Edition to TeamForge 8.2* on page 343.

5. For the end user, permissions mostly work the same way on the new site as on the CEE site. However, it may be useful to know about the different ways permissions are handled behind the scenes.

For a detailed comparison of permission handling between the two platforms, see *How do user permissions in CollabNet Enterprise Edition map to permissions in TeamForge 8.2?* on page 939

6. Project templates are not migrated from CollabNet Enterprise Edition to CTF.

CollabNet will provide support to manually recreate templates after the site is upgraded.

For information about how project templates work in CTF, see *Create a project template* on page 250.

7. TeamForge 8.2 URLs are different from CollabNet Enterprise Edition URLs. Backward compatibility is provided for certain important URLs.

For detailed information about how URLs are handled, see *How do URLs in CollabNet TeamForge 8.2 compare with URLs in CollabNet Enterprise Edition?* on page 976

8. Remote Publishing

Remote publishing gets its own repository now. It is listed as a publishing repository in the SCM listing page. Please check out the publishing repository and use it for all remote publishing activities.

9. Customize Project Home Page

Post migration, please check the **Project Admin** > **Project Settings** `Home Page` settings to ensure that the desired content is displayed as the Home page.

If the **Show default project pages and components** option is selected, the project home page displays project pages. You will see the content of index.html (if you had one in CollabNet Enterprise Edition) in the project home page as an html component. Moving forward, you will have to edit this html component through the PCE only and you can not edit `index.html` directly in the repository to modify the project home page.

If the **Show custom web page** option is selected, the project home page displays content from the `www/index.html` of the 'Publishing' repository of the project. In this case, any changes to the `www/index.html` file will be visible in the project home page.

You can refer this help page for details: *Create a custom project home page* on page 111

10. Change of CVS URL

CVS URLs used in CollabNet Enterprise Edition will work in TeamForge 8.2, but the repository path is different. See the **Source Code** tab of the TeamForge 8.2 project for the new checkout command for your CVS repository. The CVS working copies must be recreated (checked-out afresh).

11. Subversion

Existing Subversion working copies can be used as available and do not require a fresh checkout, except for the remote publishing content under the "www" folder.

12. Re-do project requests/posts

Unapproved project requests or discussion posts are not migrated to TeamForge 8.2. If you have project requests in CollabNet Enterprise Edition that were not approved until migration, it will have to be done afresh in TeamForge 8.2.

13. CollabNet Enterprise Edition's audit history

Audit history from CollabNet Enterprise Edition is not migrated into TeamForge 8.2, but the data is archived. Please get in touch with your CollabNet representative for accessing the data.

Delegate site administration tasks with site-wide roles

To spread out the site administration burden, consider creating additional site administrators with permissions to do almost everything a full site administrator can do.

To allow some CollabNet TeamForge users to use one or more TeamForge tools across projects, give them a role that allows them to use project tools in any project on the site.

Create additional site administrators

To assist in the administration of the TeamForge site, a person must have a site administrator user account with a corresponding role on that site. TeamForge administrators can create suitable site-wide roles and delegate site administration responsibilities.

☞ **Note:** You can choose site administration permissions through site-wide roles.

1. Go to **My Workspace > Admin**.
2. Click **Roles** from the **Projects** menu.
3. Click **Create**.
4. On the **SITE-WIDE ROLE** tab, write a name and description for the role.

   The role name is case-sensitive.
5. To prevent inheritance of the role into private projects, select the **PREVENT ACCESS** option.

   ☞ **Note:** Selecting the option to prevent role inheritance does not affect access to public and gated projects. On selecting **Prevent access**, the user may not be allowed to do project-permissions related tasks in private projects.
6. Click **Create**.
   The restricted site administrator role is created. The **Edit Site-wide Role Permissions** page appears.

   ☞ **Note:** You can select the permissions for site administration tools as well as for applications available across all projects.
7. Select the apt site administration and/or project permissions listed on the **ROLE PERMISSIONS** tab, to match the responsibility assigned to a user with that role.

   ☞ **Tip:** You may not want to risk delegating the task of deleting projects, users, groups, roles or categories.

   ☞ **Note:** If you are creating a site-wide role that has Project Tracker's "Configure - Site" permission, you must also assign the "Role - View" permission.

   ☞ **Note:** To manage artifact types globally, users must have project administrator permissions in a site-wide role.

The role is created. You can assign it to site members at any time.

Modify the additional site administrators privileges

If restricted site administrators need to do things that are not allowed by a role you have assigned to them, you may need to change the permissions associated with that role.

When you edit a role, all restricted site administrators with that role get the updated permissions automatically.

☞ **Tip:** You may have prevented the access permission into private projects earlier. Now you can modify the restriction or change other relevant permissions.

1. Go to **My Workspace > Admin**.
2. Click **Roles** from the **Projects** menu.

3. From the list of roles, click the role you want to edit or select the check box and click **Edit**.

4. On the **Edit Site-wide Role Permissions** page, make the changes you need.

   - To edit the title or description of the role, click **Edit**.
   - To edit the site administration and/or project permissions, choose an application from the left side of the page and select or deselect permissions and resources.
   - To edit the site members to whom the role is assigned, click **Assigned Users** tab.

5. Click **Save**.

Make selected users additional site administrators

You can empower site users to assist in site administration by giving them a suitable role.

Depending upon the permissions you grant via site-wide roles, you can select site users who could be granted the privilege.

1. Go to **My Workspace > Admin**.

2. Click **ROLES** from the **Projects** menu.
   The existing site-wide roles are listed.

3. Click the role that you want to assign to the site users.

4. On the **Edit Site-wide Role Permissions** page, click the **ASSIGNED USERS** tab.
   All users who currently have the role are displayed.

5. Click **Add**.

6. In the **Find a User** window, select the site users you want to add, and move them from the **Found Users** list to the **Selected Users** list. Click **Add**.

   ☞ **Note:** You can search by full or partial user name or full name to find the desired site members.

7. Click **OK**.

The additional site administrators are now ready to act! Their names are added to the **Assigned Users** list.

Give project-independent access for project tools

To allow some CollabNet TeamForge users to use one or more CollabNet TeamForge tools across several projects, create site-wide roles with specific project permissions, minus site administrative permissions. Assign these site-wide roles to those who may need to access the project tools in any project.

☞ **Note:** For example, you may want a user to be able to use Tracker across several projects. You don't need to create and assign a role supporting the task individually across all projects. Just do it one time as a site-wide role and assign it to the user.

1. Go to **My Workspace > Admin**.

2. Click **Roles** from the **Projects** menu.

3. Click **Create**.

4. On the **Create Site-wide Role** page, write a name and description for the role.
   The role name is case-sensitive.

5. To prevent inheritance of the role into private projects, select the **PREVENT ACCESS** option.

   ☞ **Note:** Selecting the option to prevent role inheritance does not affect access to public and gated projects. On selecting **Prevent access**, the user may not be allowed to do project-permissions related tasks in private projects.

6. Click **Create**.
   The site-wide role is created. The **Edit Site-wide Role Permissions** page appears.

   ☞ **Note:** You can select the permissions for applications and resources available across all projects.

7. Select the required project permissions listed on the **ROLE PERMISSIONS** tab, to match the tasks you want the user with that role to perform.

   ☞ **Tip:** Select "Tracker-Create" permission if you want the user to be able to create new trackers.

The role is created. You can assign it to site members at any time.

Provide ready-made roles

You might want to create roles that projects across the site can use with minimum effort and maintenance. Using global project roles is an easy way of enforcing role-based similarities and removing role duplication across projects.

You can suggest to the project administrators to use common global project roles while assigning project tasks in CollabNet TeamForge, instead of creating and managing several similar roles for their individual projects.

Create a global project role

To help project managers get their project members set up quickly, provide ready-made project roles that any project on your site can use.

☞ **Note:** You need the "Role-Create" permission to create global project roles. All site administrators and some restricted site administrators have this permission.

1. Go to **My Workspace > Admin**.

2. Click **Roles** from the **Projects** menu.

3. Click the **GLOBAL PROJECT ROLES** tab.

   All the existing Global Project Roles are listed here. It is a good idea to check this list before you create another role.

   ☞ **Note:** You can suggest that the project administrators check the **Project Admin** > **Permissions** > **Roles** > **View: Global Project Roles** list before creating any new roles in their projects.

4. Click **Create**.

5. On the **Create Global Project Role** page, write a name and description for the role.

   The role name is case-sensitive.

   ☞ **Tip:** Remember that the role name can not be the same as a site-wide role name.

6. To allow inheritance of the role's permissions into private sub-projects, clear the **PREVENT INHERITANCE** option.

7. To allow the project members to be able to request this role, select **PROJECT MEMBERS CAN REQUEST THIS ROLE**.

   Project members can submit requests for Available upon Request roles. For a project, the project administrators can set an Available upon Request role to be automatically granted to the project member requesting it.

8. Click **Create**.
   The new global project role is created. The **Edit global project role permissions** page appears.

9. Select the application permissions that are relevant to the role, from those listed on the **ROLE PERMISSIONS** tab.

   ☞ **Tip:** You may want to restrain providing project or application administration permissions, until required.

The role is created. The project administrators can assign it to their project members any time.

Modify a global project role

You may need to add or remove certain permissions from an existing global project role to assign new tasks or change the access permissions given via the role.

☞ **Note:** Only site administrators or restricted site administrators with "Role-Edit" permission can edit global project roles.

1. Go to **My Workspace > Admin**.

2. Click **Roles** from the **Projects** menu.

3. Click the **GLOBAL PROJECT ROLES** tab.

   All the existing global project roles are listed here.

4. Select the role that you want to edit and click **Edit** or just click the hyperlinked role name.
   The **Edit global project role permissions** page appears.

5. Click **Edit** to make changes to the role details.

6. Modify the **Role Name** or **Description**, if required.

    The role name is case-sensitive and must not be the same as a site-wide role.

7. Change the inheritance setting to prevent or allow inheritance of the role's permissions into private sub-projects.

8. To make the role requestable or non-requestable, change the **Project members can request this role** setting.

    Project members can submit requests for Available upon Request roles. For a project, the project administrators can set a Available upon Request role to be automatically granted to the project member requesting it.

9. Click **Update**.
    The global project role is modified.

10. Select the application permissions that are relevant to the role, from those listed on the **ROLE PERMISSIONS** tab and click **Save**.

    ☞ **Tip:** You may want to restrain removing project or application administration permissions as the change impacts existing users too.

The role is modified.

Delete a global project role
If you no longer need a global project role, you should delete it.

On deleting a global project role, all the user associations in the projects are removed.

☞ **Note:** Only site administrators or restricted site administrators with "Role-Delete" permission can delete global project roles.

1. Go to **My Workspace > Admin**.

2. Click **Roles** from the **Projects** menu.

3. Click the **GLOBAL PROJECT ROLES** tab.

    All the existing global project roles are listed here.

4. Select the role that you want to delete and click **Delete**.
    A confirmation message appears.

5. Click **OK** to continue with deleting the selected role.

The selected global project role is deleted and all its associations are removed.

Manage projects as a group
To manage two or more independent projects, create a project group in CollabNet TeamForge 8.2. Similar to projects, a project group provides the platform for sharing project members, roles and permissions across a group of projects. With some well planned settings, you can manage several of your projects and also effectively control the project members accessing each project.

To begin controlling multiple projects at one go, create a project group.

Create a project group
To begin managing your several projects together, create a project group in TeamForge 8.2.

☞ **Note:** When you create a project group, you are granted administration rights for the group and will be able to perform actions such as adding or removing a project from the group. You will be listed as an administrator in the group's **Project Group Details**. (Site administrators are exempt from this, since they can perform all actions on the site without requiring specific roles).

1. Go to **My Workspace > Admin**.

2. Click **PROJECT GROUPS** from the **Projects** menu.

    The existing project groups are listed here.

3. To create a new project group, click **Create Project Group**.

4. On the **Create Project Group** page, write a name and description for the project group.

    The project group name must be unique, however, it can be the same as any of the projects.

The project group is created.

In your **Project Group** page, you can add projects, add project group members or specify roles for the project group.

☞ **Tip:** You can always come back to this page later to specify projects, users or roles that affect your project group.

Edit a project group

You can make changes to a project group to keep it updated with the various projects you may be managing with that in CollabNet TeamForge C8.2. You must have the administrator permissions for project groups to make any modifications.

☞ **Tip:** You could be either a site administrator or a project administrator, but you must have the project groups administration permissions to manage projects as a group.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu.

   The existing project groups are listed here.
3. Click the project group that you want to modify. The **Project Group Details** page appears.
4. Click **Edit** and make the changes as required.

   You can update the project group name, description as well as the administrators.
5. To add one or more administrators for the project group, click the Search icon next to **Administrators**.
   a) On the **Find a User** page, select the required administrators, click **Add** and click **OK**.

   The project group administrators are listed on the **Project Group Details** page.

The project group is modified.

From the **Project Group Details** page, you can add the projects, add project group members or specify the desired roles for the project group.

Delete a project group

If the projects being managed under a project group have achieved their targets you may not need the project group anymore. You can delete a project group in CollabNet TeamForge C8.2. You must have the administrator permissions for project groups.

☞ **Tip:** You could be either a site administrator or a project administrator, but you must have the project groups administration permissions to manage projects as a group.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu.

   The existing project groups are listed here.
3. To delete a project group, select the project group from the list and click **Delete**.

   You may get a warning message as the projects being managed under the group may have active role assignments. Click **OK** on the message to proceed with deleting the project group.

The project group is deleted.

☞ **Tip:** The relationship between projects and the project group is broken and you can no longer manage projects using the deleted project group.

Add projects to a project group

You may be interested to start using the project group that brings your projects together. Add your projects to your project group as the initial step.

☞ **Tip:** You could be either a site administrator or a project administrator, but you must have the project groups administration permissions to manage projects as a group.

If you have just created your project group, you might already be on the **Project Group Details** page. Skip the first three steps in that case.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu.

The existing project groups are listed here.

3. Click your project group. The **Project Group Details** page appears.

4. On the **Project Group Details** page, click **Add** to associate your projects with the project group.

   A list of projects appears.

   ☞ **Tip:** You can only add the projects for which you are the project administrator.

5. From the **PROJECT** list, select the projects and click **Add**.

The selected projects are added to the project group.

If you add an irrelevant project to the group, you can select it from the **Project Group Details** page and click **Remove** to remove it from the project group.

☞ **Tip:** You can always come back to this page later to add more projects, specify users or roles that affect your project group.

Manage user membership for a project group

Your project group needs to be set up with project group members to facilitate any administrative tasks that you may want to do.

☞ **Tip:** You could be either a site administrator or a project administrator, but you must have the project groups administration permissions to manage projects as a group.

If you have just created your project group, you might already be on the **Project Group Details** page. Skip the first three steps in that case.

1. Go to **My Workspace > Admin**.

2. Click **PROJECT GROUPS** from the **Projects** menu.

   The existing project groups are listed here.

3. Click your project group. The **Project Group Details** page appears.

4. From the left navigation pane, click the **User Membership** link to add users to the project group.

5. On the **PROJECT GROUP MEMBERSHIP** tab, click **Add**.

6. On the **Add User** page, find the users you want by one of these methods:

   • Under **Search for Users**, filter the list of site users eligible to join this project group. You can filter by full or partial name or user name.

      ☞ **Note:** Search text is not case-sensitive.

   • Browse the list of registered users on the site. Sort them by name, user name, email address or membership status.

      ☞ **Note:** If a site has a great many users, you must filter them first to narrow down the list. This helps avoid slowing down the system.

7. Select the users you want to add.

8. Under **Assign Roles (Optional)**, select the roles you want the users to have.

   You can select any available global project role or role created just for this project.

   ☞ **Tip:** If you prefer, you can skip this step and assign roles later on.

9. Save your changes.

   • Click **Save** to return to the **Project Group Membership** page.
   • Click **Save and Add More** to keep adding users.

The selected users are granted membership to the project group.

If you add a user who may not need to be a member of the project group, you can select the user from the **Project Group Membership** page and click **Remove** to remove the user's membership.

Manage users' access to project groups

As a project group administrator, you know that the key to any user's access to the project group and the projects belonging to the group, is in the role you assign to the user. You can do almost all the role-related activities with project groups that you could do with individual projects.

You can create site-wide roles and assign the users to those roles and then bring them into your project group. You can create roles specifically for your project group and assign users to those or use global project roles, as required. The permissions inherited through the assigned roles are subject to the "Prevent Inheritance" option setting done while creating the roles.

Give a role to project group members

A role can be assigned to many project group users at once.

While the user-role matrix provides a convenient way to add project group members to a role, it can become unwieldy if the project has a large number of users or roles. When that is the case, try assigning roles to multiple project group users at one time.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu.

   The existing project groups are listed here.
3. Click your project group. The **Project Group Details** page appears.
4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.
5. On the **Roles** tab, click the **Global Project Roles** or **Roles Created For a Project** option of **View**, to display existing global roles or roles created just for this project.
6. Click the name of the role that you want to assign to project group members.
7. On the **Edit Role** page, click the **Assigned Users** tab.
   The **Assigned Users** page shows all users who currently have the role.
8. Click **Add**.
9. In the **Find a User** window, select the project group members you want to add, and move them from the **Found Users** list to the **Selected Users** list.

   • Click **Add** to move selected users.
   • Click **Add All** to move all users.

   ☞   **Note:** You can search by full or partial user name or full name to find the desired project group members.

10. Click **OK**.

The project group members are now assigned the role.

Give roles to a project group member

A project group member can have any number of roles. As project group administrator, you must assign each project group member's roles with care. The roles would impact not just an individual project, but would also grant same permissions across the projects in a project group.

Permissions are cumulative. The project group member has all of the access permissions allowed by all of the assigned roles, plus any permissions that may have been assigned globally using application permissions.

1. Go to **My Workspace > Admin**.
2. Click **PROJECT GROUPS** from the **Projects** menu.

   The existing project groups are listed here.
3. Click your project group. The **Project Group Details** page appears.
4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.
5. Click the **User-Role Matrix** tab.

   Observe the users listed on the left and all the available roles (global and direct) on the right. Users can be assigned global roles and roles created just for this project.
6. Select roles for each project member.

7. Click **Save**.

The roles are now assigned to each project group member.

Assign roles in multiple projects to a user group

Project managers can assign a role to multiple users at once by assigning the role to a user group that contains all those users. As a site administrator, you can do the same thing across multiple projects, by treating the projects as part of a project group.

1. Go to **My Workspace > Admin**.

2. Click **PROJECT GROUPS** from the **Projects** menu.

The existing project groups are listed here.

3. On the **Project Group** page, click **Permissions** .

4. On the **User Group-Role Matrix** tab, add the user group you want, then select the roles you want to assign to that user group.

5. Click **Finish** or **Finish and Add More**.

> ☞ **Restriction:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

Assign user groups to a role

To manage permissions for a lot of groups or roles at once, try assigning user groups to roles.

1. Go to **My Workspace > Admin**.

2. Click **PROJECT GROUPS** from the **Projects** menu.

The existing project groups are listed here.

3. Click your project group. The **Project Group Details** page appears.

4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.

5. On the **Roles** tab, to display existing global or direct roles, click the **global project roles** or **Direct Roles** option of **View**.
You can assign the project group users or user groups to a global project role or a role created just for this project.

6. Click the name of the role that you want to assign to the user group members in the project group.

7. On the **Edit Role** page, click the **Assigned User Groups** tab.
The **Assigned User Groups** table shows all user groups who currently have the role.

8. Click **Add**.

9. Type some of the group's name in the **Name (search)** box and click **Find**.

10. From the **Add User Group to Role** table, select the user groups that you want to add, and click **Finish**.

> ☞ **Restriction:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

The user groups are now assigned the role.

Create a role in a project group

A role defines the applications that project group members with that role can use, and the specific things project group members can do in each application.

Any project groups administrator can create and assign a role. It is a good idea to check the existing global project roles before creating any new role for a project group.

> ☞ **Note:** Any existing site-wide or global project role can be associated with a project group. It is advisable to check the permissions granted via a role before assigning it to users or user groups in a project group as it would impact more than a single project.

1. Go to **My Workspace > Admin**.

2. Click **PROJECT GROUPS** from the **Projects** menu.

The existing project groups are listed here.

3. Click your project group. The **Project Group Details** page appears.
4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.
5. On the **Roles** tab, click **View: Roles Created For this Project Group**.

   You can view global project roles by selecting **View: Global Project Roles** before creating a new role for this project.
6. Click **Create**.
7. On the **Create Role** page, write a name and description for the role.
8. To allow the inheritance of the role into private subprojects, de-select the **Prevent Inheritance** option.

   ☞ **Note:** By default, the role inheritance into private subprojects is prevented. For example, you may not want administrator roles to be inherited in subprojects, until required.

   ☞ **Note:** Selecting the option to prevent role inheritance does not affect public and gated projects.

9. Click **Create**.
   The role is created. The **Edit Role** page appears.
10. For each application listed on the **Role Permissions** page, select the permissions and resources you want to make available to users with this role.

    ☞ **Note:** You can specify access to individual top-level folders, but not to specific subfolders.

11. Click **Save**.

The role is created. You can assign it to project group members or user groups associated with the project group at any time.

Edit a role in a project group

As a project group administrator, you may need to update the permissions granted by a role being used across the projects.

While modifying a role, it is better to be cautious about granting more access than required by the users or a user group. In the case of project groups, as a role could be mapped across projects, consider being restrictive.

1. Go to **My Workspace > Admin**.
2. Click your project group. The **Project Group Details** page appears.
3. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.
4. On the **Roles** tab, click **View: Roles Created For a Project**. The existing roles created for this project are listed.

   You can view global project roles by selecting **View: Global Project Roles**, if required.
5. Select the role that you want to modify and click **Edit**.
6. On the **Edit Role** page, make the desired changes.

   You can update the role name, description and inheritance settings on clicking **Edit**.

   The role's permissions, assigned users or assigned user groups can also be modified as required.
7. Click **Save** after making the changes.

The role in the project group is updated.

Delete a role in a project group

When you no longer need a role that you created for a project, it's a good idea to delete that role.

Any project groups administrator can create and assign a role. It is good to keep the project role tray as small and as manageable as possible.

☞ **Note:** Any existing site-wide or global project role can be associated with a project group. It is advisable to check the permissions granted via a role before assigning it to users or user groups in a project group as it would impact more than a single project.

1. Go to **My Workspace > Admin**.

2. Click **PROJECT GROUPS** from the **Projects** menu.

   The existing project groups are listed here.

3. Click your project group. The **Project Group Details** page appears.

4. From the left navigation pane, click the **Permissions** link to specify user roles applicable to the project group.

5. On the **Roles** tab, click **View: Roles Created For a Project**.

   You can view global project roles by selecting **View: Global Project Roles**. However, you can only delete roles created for this project here.

6. Select the role and click **Delete**.

   You may get a warning message if the role you are trying to delete is in use. Click **OK** to proceed with deleting the non-required role.

   The selected role is deleted.

Manage users

You can help projects succeed by creating and managing user accounts.

Communicate with all site users at once

To say something to everyone who uses your site, post a site-wide broadcast message.

For example, if you plan to upgrade your site, you may want to let users know a few days before your upgrade that the site will be unavailable for a short time.

The broadcast message can be viewed even without logging into TeamForge

1. Go to **My Workspace > Admin**.

2. Click **SYSTEM TOOLS** from the **Projects** menu.

3. Click **Broadcast Message**.

4. In the **Broadcast Message** page, type the message you want to broadcast and click **Broadcast**.

The message is displayed at the top on all the pages of the site until it is deleted or replaced.

To edit a broadcast message quickly, you do not need to go to the specific broadcast message page. You can edit it from any page, using the Edit icon that appears at the far right end of the broadcast message box.

Create a new user account

To participate in a TeamForge site, a person must have a user account on that site. TeamForge administrators can create these user accounts. This topic applies to sites with no LDAP authentication.

> **Note:** If your TeamForge site uses LDAP authentication, TeamForge administrators cannot create new user accounts. On a site with LDAP authentication, each user must log into TeamForge using his or her LDAP user name and password.

1. Go to **My Workspace > Admin**.

2. Click **USERS** from the **Projects** menu.

3. Click the drop-down arrow next to **Create** and click **Single User**.

   You can select **Multiple Users** to create several user accounts at one time.

4. On the **Create User** page, enter the field values appropriately.

   a) Enter a user name for the user.

   👉 **Note:** Your user name must meet these criteria:

   - User name is case sensitive.
   - Minimum number of characters as specified in the `site-options.conf` file.
   - No spaces.
   - Should have at least one letter.
   - The first character is a letter.

   b) Enter and confirm a password for the user, if you prefer to set the user's password yourself.

   👉 **Tip:** To invite users to create their own password, leave the **PASSWORD** field blank. A *password ticket email* will be sent to users to let them create a password.

   c) Enter the **FULL NAME** and **EMAIL ADDRESS** of the user.

   👉 **Tip:** You can add more email addresses for the user after you finish creating their profile.

   d) Enter the user's organization.

   Organization can be a geographic designation, a corporate division, or whatever you want. It's advised to keep it consistent across your site.

   e) Select the language from the **LOCALE** drop-down list.

   f) From the **TIME ZONE** drop-down list, select the preferred time zone for the user.

   👉 **Important:** Selecting the time zone overrides the default time zone set by the site-options token, *DISPLAY_TIMEZONE*. It reflects in all the email notifications and TeamForge pages excluding integrated application pages.

   g) Choose the user's TeamForge **LICENSE TYPE**.

   h) Choose a user type. You can choose only one user type for each user.

   - SITE ADMIN: Administrators have unlimited access to all the data in TeamForge.
   - RESTRICTED USER: Restricted users can only access projects of which they are members.

   👉 **Important:** If you do not select **RESTRICTED USER**, the user will be unrestricted and able to access all projects that have not been made private by a project administrator.

   i) To send a welcome message to the user, select **SEND WELCOME MESSAGE?**.

5. Click **Create**.

The user account is created.

Create multiple user accounts

To participate in a TeamForge site, a person must have a user account on that site. TeamForge administrators can provide access to multiple users by creating their accounts together.

👉 **Note:** If your TeamForge site uses LDAP authentication, TeamForge administrators cannot create new user accounts. On a site with LDAP authentication, each user must log into TeamForge using his or her LDAP user name and password.

1. Go to **My Workspace > Admin**.

2. Click **USERS** from the **Projects** menu.

3. Click the drop-down arrow next to **Create** and click **Multiple Users**.

4. On the **Create Multiple Users** page, enter up to 25 lines like this, one user per line:

   ```
   username username@yourdomain.org name organization Restricted
   ```

   👉 **Note:** Usernames must meet these criteria:

- 1 to 31 characters.
- Only alphanumeric characters.
- No spaces.
- At least one letter.
- The first character is a letter.

☞ **Tip:**

- Organization field is optional.
- To create an unrestricted user, omit "Restricted." Restricted users can only access projects of which they are members, while unrestricted users can access all projects that have not been made private by a project administrator.
- Use quotes around the full name or the organization information if it is more than a single word.
- A maximum of twenty-five user accounts can be created at one time.

5. Click **Create**.

The user accounts are created and password e-mails are sent to all the new users.

Create a user group
To manage multiple users at once, create a group that represents them.

1. Go to **My Workspace > Admin**.

2. Select **USER GROUPS** from the **Projects** menu.

3. Click **Create** and provide a name for the group and a description of its purpose.

☞ **Note:** If your project is a child of another project, it may have inherited one or more user groups from its parent project. To work with inherited users and user groups, you must go to the project that they belong to.

4. Click **Create**.

Add a user to a user group
Put together multiple users who share characteristics in a user group.

1. Go to **My Workspace > Admin**.

2. Select **User Groups** from the **Projects** menu.

☞ **Note:** In TeamForge 8.2, **Groups** (in earlier versions of the product) has been renamed to **User Groups** to better distinguish user groups from project groups.

3. Under **User Groups**, click the group to which you want to add the user.

4. On the **Edit Group** page, click**Add**.

5. Use the picker to move users into the group, and click **OK**.

You can select the inherited project members also from the list.

6. Click **Return**.

Find a user
To find a user, filter the list all CollabNet TeamForge users on your site.

1. Go to **My Workspace > Admin**.

2. Click **USERS** from the **Projects** menu.

3. Specify the filter criteria in one or more filter fields (at the top of each column) and click **FILTER**.

- You can find a filter field at the top of each column in most of the tables in the TeamForge application.
- The filter field could be a text box or a drop-down list with multi-select check boxes.

- You can type your filter criteria in the text boxes. The search text is case-insensitive.
- You can also select the filter values from one or more drop-down lists. By default, you can only select up to 10 filter values in a drop-down list. However, you can set a value that suits your requirement for the *FILTER_DROPDOWN_MAX_SELECTION* token in the `site-options.conf` file to increase or decrease the count.
- **Filter-as-you-type**: You can find the **Enter keywords** text box in all filter drop-down lists. As you type your filter keyword, instant search results are shown in the drop-down list. For example, in the following illustration, typing "R" instantly shows all statuses having the alphabet "R". The search text is case-insensitive.

- Some search filters may not appear if your site administrator has not enabled them.

4. After filtering, if you want to clear the filters, click **FILTER** and select **Clear** from the drop-down list.

All users meeting your filter criteria are displayed.

Find a user's groups

You can get a consolidated view of all user groups of which a user is a member.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. On the **Users Details** page, click the **USER GROUP MEMBERSHIP** tab.

The groups listed on this tab are the groups that this user is a member of.

Edit a user account

When a user has trouble accessing the site, you may need to reset the user's password or change the user's account status.

☞ **Note:** If your TeamForge site uses LDAP for single-sign-on, passwords must be reset in the LDAP system, not on the Web administration pages. Ask your system administrator for help.

☞ **Important:** To avoid disasters, TeamForge makes it impossible to delete or deactivate the TeamForge admin account. You also can't remove the TeamForge admin flag or mark the admin user as a restricted user.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. On the **USERS** tab, click the name of the user whose account you want to edit.
4. On the **User Details** page, click **Edit**.

5. On the **Edit User Information** page, make your changes and click **Update**.

   You can specify up to a maximum of three alternate email addresses, if required.

Act on multiple user accounts at once

A TeamForge administrator can edit the status of multiple user accounts simultaneously.

For example, if you have multiple pending new accounts to approve, you can approve them in a batch instead of individually editing each account.

☞ **Note:** A pending user is a user who has requested an account but has not yet confirmed his or her email addresses.

☞ **Important:** In the case of TeamForge admin accounts, you cannot make any of these edits:

   - Delete the account.
   - Change the account status to anything but active.
   - Remove the TeamForge admin flag.
   - Mark it as a restricted user.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. On the **Users** page, select the users whose status you want to edit.
4. Click the desired status change.

   - **Delete** - Deleted users are removed from all projects. All assigned items are removed from the user. Deleted users do not count against your TeamForge license count.
   - **Deactivate** - Deactivated users cannot log in to TeamForge and do not receive notification messages, but they remain members of projects and selection lists.
   - **Activate** - Active users have full use of TeamForge, subject to RBAC permissions.

Reset the admin account password

If your TeamForge installation authenticates against an LDAP directory, follow these instructions to reset your admin account password.

If your installation does not validate against LDAP, click **Forgot Your Password** on the TeamForge home page to reset the password for the admin account.

1. With a web browser, go to the URL http://<host>sf/sfmain/do/forgotAdminPassword.
2. On the **Admin Account Password Retrieval** page, Click **Send Email**.
   TeamForge sends an email to the address specified for the admin user.
3. Check your email and click the link provided to reset your password.
4. On the **Reset Password** page, enter and confirm a new password.
5. Click **Reset Password**.

You can now log into CollabNet TeamForge with your new password.

License information

You can obtain a summary of the license information from the **License Info** page.

The **License Info** page provides you with all the basic information about the licenses you purchased for your TeamForge site. This includes details such as the number of TeamForge licenses you had obtained, how many you have used, expiration date and so on.

Add ALM functionality to your site

If you started using TeamForge as a source code management application, you can expand it to support the full range of application lifecycle management functions.

Your license key allows for some number of users with ALM licenses and some number of users with SCM licenses. (*What's the difference?*) To convert your site to ALM mode, one or more ALM user licenses must be included in your license key and not assigned to any user.

- If you don't already have a license to use TeamForge in ALM mode, talk to your CollabNet representative about getting one.
- To apply your ALM site license, see *Supply your TeamForge license key from Teamforge user interface* on page 435.

1. Go to **My Workspace > Admin**.
2. Click **LICENSE INFO** in the Site Administration toolbar.
3. Click the **Change Site Mode to** drop-down list and select **ALM**.

Your site's users can now see and use the full range of applications on your site, including Tracker, Documents, Tasks, and Reports.

☞ **Note:** When your site is in ALM mode, any users you create get ALM licenses by default until all the available ALM licenses are used.

Restrict your site to source code functionality

TeamForge suppports the full range of application lifecycle management functions, but you can also use it for only the core source code management (SCM) functions.

☞ **Note:** Your site may have been switched to SCM mode due to a license issue. (*Why?*) If this is the case, see *Add ALM functionality to your site* on page 275 to fix it.

1. Go to **My Workspace > Admin**.
2. Click **LICENSE INFO** in the Site Administration toolbar.
3. Click **Change Site Mode to SCM**.

Your site's users can now see and access only the Source Code, Discussions, and Wiki tools in the toolbar.

☞ **Note:** When your site is in SCM mode, each user you create gets an SCM license by default.

View all roles assigned to a user

A user can have multiple roles in different projects either by being directly assigned those roles or by inheriting them. You might find it useful to see all the roles assigned to a user in a TeamForge site before adding or removing a role.

1. Go to **My Workspace > Admin**.
2. Click **USERS** from the **Projects** menu.
3. In the **ROLES** tab, select a role type in the **View** drop-down - Roles created for a Project, Roles Inherited From Parent Project or Site-wide Roles.

   - Roles created for this project include the roles the user is directly assigned or assigned through a user group in projects and project groups.
   - Inherited roles include the roles the user inherits from parent projects and project groups.

Support projects

CollabNet TeamForge administrators can do a variety of things to help projects on the site be successful.

Create a new project

TeamForge administrators can create new projects without having to submit them for approval.

☞ **Note:** When a TeamForge administrator creates a new project, he or she is not made a member of the project, and the Founder Project Admin role is not created. To designate a project administrator, you must add the user to the project, then create and assign a project administrator role manually.

1. Go to **My Workspace > Admin**.
2. In the list of TeamForge projects, click **Create Project**.
3. On the **Create Project** page, provide a name for the project.
   This is the name that will appear in all project lists and on the project home page.
4. Enter a URL name for the project, if appropriate.
   This is the name that will appear in the project's URL.
5. Write a description of the project.

6. Select a project template.

A project template is used to pre-populate new projects with the structure and configuration of an existing project. If you do not want to use a project template, choose **None**.

> ☞ **Note:** If you create a project from a template that contains an integrated application, you may have to provide some information specific to the integrated application. For example, for Project Tracker you must set a new artifact prefix that is different from the prefix in the template.

7. Click **Create**.

The project is created.

Approve a new project

Any registered TeamForge user can request a new project. A new project is activated only after a site administrator approves it.

Before approving a new project, you have the option to review the project details.

1. Go to **My Workspace > Admin**.

2. Click the **PENDING PROJECTS** tab.

3. Select the projects you want to approve from the pending projects list.

- Click **Approve** to approve the project and move it to the **All Projects** page with status **Active**.
- Click **Reject** to reject the project and remove it from the list.

The project requester receives an email notification when the project is approved or rejected. If you entered a comment, that also appears in the email notification.

Rename a project

As the focus of a project shifts, its name or description can become obsolete. A site administrator can update the name or description to help keep the project current.

1. Go to **My Workspace > Admin**.

2. In the TeamForge project list, click the project you want to edit.

3. Click **PROJECT ADMIN** from the **Project Home** menu.

4. On the **Project Admin Menu**, click **Project Settings** page, make the changes you need.

5. Click **Save**.

Delete a project

If you no longer need a project or any of the data in it, you should delete it.

Deleting a project deletes all of the data within it, with the exception of source code data, which is maintained separately from the rest of the site's content.

> ⚠ **Caution:** Delete a project only if you are sure that you no longer need any of the data within it. Move any items that you want to save.

> ☞ **Note:**
> - You can delete a parent project only when its members, user groups and roles are not in use in any other project.
> - When you delete a parent project, the direct subprojects are moved one level up; that is, under the immediate parent of the deleted project.
> - If the deleted project has no parent, its subprojects become root-level projects.

1. Go to **My Workspace > Admin**.

2. From the TeamForge project list, choose the project that you want to delete and click **Delete**.

The project is deleted.

Lock or unlock a project

To ensure that no changes occur in a project while you are collating or migrating project data, lock the project. You must have project administration permissions or be a site administrator to lock or unlock a project.

To lock or unlock a project in TeamForge, go to **Project Settings** and lock/unlock the project.

☞ **Note:** A locked project does not allow any member (including project administrators and site administrators) to make any changes to the project. Besides that, a locked project can not be set as the parent project for any other project and tasks like adding, editing or deleting integrated applications are also not allowed.

Click **PROJECT ADMIN** from the **Project Home** menu.

The project is locked or unlocked as desired. The lock icon 🔒 **Locked** appears on all the project pages while the project is locked.

☞ **Note:** If a locked project has an integrated application, for example, project tracker, all the project tracker pages are also non-editable while the project is locked. The user who has access permissions for the integrated application can only view the pages.

Update a project template

To revise or correct an existing project template, overwrite it with a template of the same name.

You must be a site administrator to overwrite an existing project template.

1. Revise the project that will serve as the basis for the new project template.

   ☞ **Tip:** You can disable a project template while you make changes to the project. To disable a project template, use the **My Workspace** > **Projects** > **Templates** page.

2. Select **PROJECT ADMIN** from the **Project Home** menu.
3. On the **Project Settings** page, click **Create Project Template**.
4. Select **Replace Existing Template** and choose the template you want to overwrite.
5. Provide the description for the template.

   (If you want to change the template name too, create a new template from the same project and disable the existing template.)

   ☞ **Tip:** It's a good idea to use the description to note the changes from the previous version of the template.

6. Select the items you want to be available when new projects are created from this template.

The replacement project template is created. Its name and description appear on the **Template** tab of the **Projects** list, accessible from your personal navigation bar.

Enable or disable project templates

Site administrators or users with site-wide roles with the project administration permissions only can enable/disable project templates.

In TeamForge, enable a project template to make it available for use while creating new projects.

1. On **My Page**, click **Projects** and select the **PROJECT TEMPLATES** tab.
2. To make the project template available for use, select the template and click **Enable**.
3. To stop making the project template available for use, click **Disable**.

When projects are created afresh, only the enabled project templates are displayed.

Require documents to be associated with artifacts

To help reduce the problem of "orphan" documents, require users who create a document anywhere on the site to associate the document with an artifact.

"Orphan" documents are documents that are abandoned because they are not connected to any tracked activity.

1. Open the `conf/site-options.conf` file in a text editor.
2. Change the value of the sf.requireAssociationOnDocumentCreate variable to true
3. Change the value of the sf.allowedAssociationTypeOnDocumentCreate variable to [TrackerArtifact]
4. If you want to prevent users from associating documents with closed artifacts, change the value of the sf.requireArtifactToBeOpenOnDocumentAssociation variable to true

5. Save `conf/site-options.conf`.

6. Recreate the runtime environment.

    ```
    ./install.sh -V -r -d /opt/collabnet/teamforge
    ```

Manage project categories

To help users navigate your site, help them sort projects into categories that make sense.

Add a project category

When you set up project categories for your site, project administrators can use this taxonomy to organize their projects.

You can create any number of top-level categories and any number of sub-category levels.

1. Go to **My Workspace > Admin**.

2. Click **CATEGORIES** from the **Projects** menu.

3. In the **Project Categories** tree, find the location where you want to create the new category.

    - Highlighting **Project Categories** creates a new top-level category.
    - Highlighting any category creates a sub-category beneath it.

4. Click **New**.

5. In the **Create Category** window, write a name and description for the category.

6. Click **Save**.

The category is created. It appears in the **Project Categories** navigation tree, and is available for use by all project administrators when categorizing their projects.

Edit a project category

A project category's membership and function may change over time. If it does, you can update the category's name or description.

1. Go to **My Workspace > Admin**.

2. On the site administration navigation bar, click **CATEGORIES**.
   The **Project Categories** tree displays the hierarchy of existing categories.

3. In the **Project Categories** tree, find the category that you want to edit.

4. Make the changes you need and click **Update**.

Move a project category

You can reorganize projects by moving a project category to another place in the project category hierarchy.

You can move a project category in the following ways:

- From a top-level category to a sub-category
- From a sub-category to a top-level category
- From a sub-category to another sub-category

When you move a project category, any sub-categories that it contains are also moved to the destination category.

1. Go to **My Workspace > Admin**.

2. Click **CATEGORIES** from the **Projects** menu.

3. On the **Edit Category** page, in the **Project Categories** section, click the project category you want to move.

4. On the **Edit** menu, click Cut.

5. Find the location to which you want to move the selected project category.

   You can move a project category either to the root category or into any other project category.

6. Choose the Paste option from the Edit menu.

The project category is now moved to the selected destination.

Delete a project category

If you no longer need a project category, you should delete it.

When you delete a project category, all of its sub-categories are also deleted.

1. Go to **My Workspace > Admin**.

2. Click **CATEGORIES** from the **Projects** menu.
   The **Project Categories** tree displays the hierarchy of existing categories.

3. Using the document tree, find the project category that you want to delete.

4. Choose **Delete** from the **EDIT** menu.

The project category and all of its sub-categories are deleted.

Stop using project categories

If you do not need to sort projects into categories, remove the ability to do so on your site.

By default, project categorization is disabled for new TeamForge installations.

☞ **Note:** Disabling project categorization does not delete categories you have already set up.

1. Go to **My Workspace > Admin**.

2. Click **CATEGORIES** from the **Projects** menu.

3. On the **Edit Category** page, select **Disabled** for **SITE-WIDE CATEGORIZATION** and click **Update**.

Project categorization is disabled for your TeamForge installation.

Add a parent project to your project

You can coordinate work among multiple projects; enable the user, user group and role inheritance by adding a common parent project to several projects.

☞ **Note:** A subproject can have only one parent. You can change or remove that parent at any time.

☞ **Note:** You must be a project administrator or site administrator to add/edit or remove the parent projects.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Settings** page, click **Add Parent**.

3. Choose a parent project that it makes sense for your project to belong to, and click **Update**.

   ☞ **Note:** You cannot add a parent project for the 'Look' project as it is a special project in itself.

Separate a subproject from its parent project

When a subproject grows beyond its original scope, you may want to make it stand-alone project or move it to a different project hierarchy.

By removing the association with a parent project, you can manage the subproject as a separate project.

☞ **Note:** Only one parent project can be selected for a subproject. However, the parent project can be changed or removed, as required.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Settings** page, click **Edit Parent**.

3. Change the parent project as required.

   You can be a project administrator or a site administrator to change or remove a parent project. As a project administrator, you can remove or change a parent project only if you have administrator permissions for both the projects that are being linked.

   ☞ **Note:** A parent project can be removed or changed only when its members, user groups and roles are not in use in any other project. In other words, you can not remove/change a parent project while its members, user groups or roles are in use in any other project.

   • • On the **Choose a Parent Project** page, select the desired parent project and click **Update**.

      ☞ **Note:** If the project hierarchy exists, the project and its subprojects are moved only under the project from which members, user groups or roles are inherited. If project hierarchy does not exist and no inheritance is in use, the project is made a 'Root' project.

- • Click **Remove Parent** in the **Project Settings** page to remove the association with parent project.
  - • On the pop-up message box, click **OK**, if you wish to make the project a 'Root' project.

The project hierarchy is changed or removed in accordance with role based access control and inheritance rules.

Provide source control services
You can add a source control server to the CollabNet TeamForge environment at any time after CollabNet TeamForge installation is complete.

Integrate a source code server
A site must have one or more servers to handle source code repositories and users. The source code server can be the same server as the application server or a separate server.

When you set up a managed software configuration management (SCM) server, you enable users to create, manage and share repositories through CollabNet TeamForge.

> ☞ **Note:** The ability to add integration servers depends on the value of the *DISABLE_CREATE_INTEGRATION_SERVERS* flag in the `site-options.conf` file. You can add new integration servers when the flag is set to its default value of "false".

You can integrate more than one source code server of a given type. For example, you can have two or more Subversion servers on your site. Consult a system administrator about the requirements for setting this up.

> ☞ **Tip:** If you use a source code solution other than Subversion or CVS you can integrate it using the CollabNet TeamForge SOAP APIs. This enables you to exchange commit data with any SCM application. Consult your CollabNet TeamForge system administrator.

> ☞ **Note:** CVS servers that integrate with CollabNet TeamForge must use the native UNIX/Linux authentication method, and not external authentication mechanisms such as NIS, NIS+, Winbind, Active Directory, or LDAP. TeamForge creates and manipulates local system accounts using the default `useradd`, `usermod`, `groupadd`, `groupdel`, and `userdel` commands. It expects to find any accounts or groups it created in `/etc/passwd` and `/etc/group`.

1. Go to **My Workspace > Admin**.
2. Click **INTEGRATIONS**.
3. On the **SCM INTEGRATIONS** page, click **Create**.
4. On the **Create Integration** page, write a name and description for the integration.
5. Choose the type of SCM server you want.

   > ☞ **Restriction:** When you give a group access to a CVS or Wandisco Subversion repository, members of the group can view the repository but cannot do repository actions, such as commit and update. You must assign those permissions to users individually.

   > ☞ **Note:** The SCM Adapter option only works if you have created your own SCM integration using the CollabNet TeamForge SOAP APIs.

6. Supply the host name for the **Soap Service Host**.

   This is the network address of the machine on which the integrated service, such as Subversion, is running.

   > ☞ **Note:** The default `localhost` will work only if the integration server is on the same server as the CollabNet TeamForge server.

7. Leave the default values in the **SOAP Service Port** field.
8. Specify whether users will use SSL to connect to their repositories.
9. Change the **Repository Root** value if you want to store the repository on your server in a different location.

   The repository root is the top-level directory under which all source code repositories reside.

10. Select the **Requires Approval** check box if you want an administrator to approve all repositories created on the server. By default, unmanaged servers require approval for all repositories, because repositories must be created and integrated manually.

☞ **Note:** By default, repositories created (or deleted) by site administrators and users with site wide role (with Integrations - SCM INTEGRATIONS permission) need no approval.

11. Supply the URL by which your users will access the service.

This will be of the form `http://<myscmserver.com>/integration/viewvc/viewvc.cgi`

☞ **Important:** If your system administrator has upgraded your site from SourceForge Enterprise Edition 4.4 or earlier, remove the port number in the SCM Viewer URL.

☞ **Tip:** If you are working with a CVS server that uses Pserver authentication, ask your system administrator for the right URL.

12. The **Use Internal Code Browser** option is selected by default to allow the users to access the TeamForge code browser for a Subversion or Git server. For more information about this feature, see *Get the code*.

☞ **Important:**

- Software requirements for using TeamForge code browser, if Subversion or Git is running on a separate server: Subversion Edge 5.1.0 and TeamForge - Git integration 8.4.4 and later.
- You need to specify the **SCM Viewer URL** *(Step 11)*, the way you would for ViewVC or Gitweb. The 'http://' or 'https://', the host name and the port number (if any) need to be set correctly. As a minimum requirement, the URL should point to the root of the SCM http server and the domain name.
- If the SCM server is not the same as the TeamForge server, make sure you install SSL certificates or visit the URL for the site directly from your browser, and import and trust the certificate into the browser.

13. Click **Save**.

CollabNet TeamForge attempts to validate the SCM viewer URL. If it cannot validate the URL, you can:

- Correct it if you have entered it incorrectly.
- Select **Save with errors** if the URL is different for an end user than it is for the CollabNet TeamForge server; for example, if you have a firewall in place.

- If you are adding a managed source code server, it is now added. All projects can now establish repositories on the server.
- If you are adding an unmanaged CVS server, all projects can now request repositories on the server. A CollabNet TeamForge administrator must create and integrate them manually.

☞ **Note:** Only CVS servers can run unmanaged.

Approve or reject a source control repository request

When a user requests a source code repository on a source control server for which you have required approval, a CollabNet TeamForge administrator must approve the request before the repository is created.

When adding a source control server integration, you have the option to require CollabNet TeamForge administrator approval for all repositories created on the server.

When a user requests a source code repository on a managed SCM server, the repository is created automatically after it is approved by a CollabNet TeamForge administrator.

☞ **Note:** Before approving a source code repository request for an unmanaged SCM server, a CollabNet TeamForge administrator must create and integrate the repository manually.

1. Click **Admin** in the TeamForge navigation bar.
2. Click **Integrations** from the **Projects** menu.
3. Click the **PENDING SCM INTEGRATIONS** tab.

☞ **Note:** Non-site administrators can now access the SCM Integrations tab if they have permission to manage SCM integrations.

4. From the list of pending SCM repository requests, select the SCM repositories that you want to approve.

- Click **Approve** to approve the repository.
- Click **Reject** to reject the project and remove it from the list.

The person who requested the repository receives an email notification when the repository is approved or rejected. If you entered a comment, that also appears in the email notification.

Approve or reject an unmanaged SCM server access request

When a user asks for access to an unmanaged SCM server (such as an unmanaged CVS server), an administrator must approve or reject the request.

When a project administrator assigns a role that provides SCM access to a project member, a CollabNet TeamForge administrator must manually create the user account on the unmanaged SCM server. Because user creation on unmanaged SCM servers is not managed by CollabNet TeamForge, TeamForge cannot verify that the user account has been created. A TeamForge administrator must confirm that he or she has created the account.

Requests for SCM access removal are also submitted for approval by the TeamForge administrator.

1. When you get a repository access request on an unmanaged SCM server, log on to the server and create the requested user account on the unmanaged SCM server.
2. Log on to the TeamForge application server.
3. Go to **My Workspace > Admin**.
4. Click **Integrations** from the **Projects** menu.
5. Click the **SCM ACCESS REQUESTS** tab.
6. As the user account is created, select the repository access request from the **Repository Access Requests** section and click **Approve**.

   You may also click **Reject** to reject the repository access request.

The user receives an email notification that the user account has been created and that the repository access request has been approved or rejected.

Approve or reject repository delete requests

SCM system maintenance requests (such as a repository delete request) must be approved (or rejected) by an administrator.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. Click the **SCM ACCESS REQUESTS** tab.
4. Select the repository deletion request from the **System Maintenance Requests** section and click **Approve**.

   You can click **Reject** to reject the repository deletion request.

The user receives an email notification that the repository deletion request has been approved or rejected.

Edit SCM integration details

You can move or reconfigure a source control server without having to reintegrate the server into CollabNet TeamForge .

> ☞ **Tip:** If you edit or lose the permissions on your SCM server, use the **Synchronize Permissions** button on the **SCM Integrations** page to recreate the correct permissions on your SCM server from CollabNet TeamForge .

1. Click **Admin** in the CollabNet TeamForge navigation bar.
2. Click **Integrations** from the **Projects** menu.
3. On the **SCM INTEGRATIONS** tab, click the name of the SCM integration you want to edit.
4. On the **Edit Integration** page, make the changes you need and click **Save**.

Move a source code repository

When the existing code base for an application may need to be managed by a different team or project, you can move the source code repository from the first project into the other one.

1. Open the project from which you want to move a source code repository.
2. Click **SOURCE CODE** from the **Project Home** menu.
3. Select the repository you want to move and click **Cut**.
4. Open the project into which you want to move the repository.
5. Click **SOURCE CODE** from the **Project Home** menu.

6. Click **Paste**.

Connect to a Subversion Edge console

When a Subversion Edge server has been converted to a SCM Integration server in TeamForge, you can log into its management console from within TeamForge.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. In the list of servers on the **SCM INTEGRATIONS** page, click the **Open Console** link for the Subversion Edge server you want to connect to.

    ☞ **Note:** Only Subversion Edge servers have this link.

    The login page for the server's management console appears. You can log in using your TeamForge administrator credentials and view statistics such as network throughput and disk space usage for the server.

Discover a Subversion Edge server

Subversion Edge (version 1.1 and higher) servers register their service on your local network using the DNS-Service Discovery protocol. TeamForge administrators can see them.

Subversion Edge servers that were converted to TeamForge servers are displayed in the **SCM Integrations** page.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. On the **SCM INTEGRATIONS** tab, click **Discover Subversion Edge Servers**.

    A Java applet is launched, and you will see a list of Subversion Edge servers in your LAN.

You can click on a server to connect to it. To bring in the power and capabilities of TeamForge, you can *convert the Subversion Edge server to a managed SCM integration server in TeamForge*.

☞ **Note:** The number of commits made in a Subversion Edge Server can be tracked through reports. The commits made after integrating with TeamForge are available for reporting.

Manage replicas

A replica server in TeamForge is a Subversion Edge server that replicates the content of an existing core SCM integration server.

Approve a replica server request

When there is a request for a replica of a core SCM integration server, a TeamForge administrator must approve the request before the replica is created.

Replica requests from a TeamForge admin user or site administrator are automatically approved. Replicas requested by other users need approval by a TeamForge administrator.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. On the **SCM INTEGRATIONS** tab, click the **Pending SCM Replicas** tab.

4. From the list of pending SCM replica requests, select the SCM replicas that you want to approve.

   - Click **Approve** to approve the replica.
   - Click **Reject** to reject the replica and remove it from the list.

When a replica is approved, it is listed in the **SCM INTEGRATIONS** tab beneath the master SVN server.

Edit replica settings

As a TeamForge site administrator, you can configure replica settings for the polling frequency of the master, and repository initialization and synchronization events.

Replication events, such as creating new repositories and synching commits, are stored in a queue on the TeamForge Application Server. The replica server polls the TeamForge Application Server for new events and then processes those events on the replica server.

These events are divided into two separate pools:

- New repository initializations - this includes creating the repository and performing the initial synchronization of the content.
- All other events

When existing repositories are selected for replication, this can take a long time. It could take many hours or even days to fully replicate an entire repository across a WAN. These big events are processed in their own thread pool, so that other repositories which are already synchronized don't have to wait in line for them to finish.

For each pool, you can define how many simultaneous events will be processed. The higher the number, the greater the potential load on both the TeamForge replica server and the Subversion master. However, this can also decrease the wait time for a given commit to appear on the replica server.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. On the **SCM INTEGRATIONS** tab, click the name of the Subversion Edge replica you want to edit.

   The **Edit System** page for the replica appears. Here's an example:

4. Change the replica name or description if required.

   👉 **Tip:** Including the geographic location would help users select a nearby replica.

5. Set **Command polling interval** to define how frequently the replica polls the master looking for new events. The replica will process all new events when it polls.

   The default value for this setting is 60 seconds, but it can range from 5 to 1000000 seconds.

6. Set **Maximum simultaneous new repository creations** to a low value.

New repository initializations can take a long time and generate a lot of load. So you wouldn't want to allow too many of them to run at once. This value can range from 1 to 100, but we suggest you keep it at 3 or less.

7. Set **Maximum simultaneous repository synchronizations** taking into account how many repositories you will be replicating and how many you think are likely to have commits occurring within the polling interval.

   This value can range from 1 to 100. You may want to set this higher than the previous field, but we suggest you keep it at 10 or less. There's no reason to enter too high a number because you are merely specifying how many synchs can run at the exact same time -- and it never runs more than one per repository.

8. When you've made your changes, click **Save**.

Remove a replica

When you remove a replica from TeamForge, it is restored to a Subversion Edge server in standalone mode.

1. Go to **My Workspace > Admin**.
2. Click **Integrations** from the **Projects** menu.
3. In the **Edit System** page for the replica, click **Delete**.

The replica is removed from TeamForge. The repositories that existed on the replica are deleted.

Integrate an unmanaged CVS repository with CollabNet TeamForge

After creating the repository, you must modify two triggers, `verifymsg` and `loginfo`, to complete the integration with CollabNet TeamForge .

The triggers must be modified manually, because much of the information in the triggers is specific to your CollabNet TeamForge installation.

Each trigger must be modified to run a `.java` program. These topics provide an annotated example and instructions on modifying and installing the trigger files.

☞ **Note:** You must create the CVS repository before you can modify and install the trigger files.

☞ **Note:** CVS servers that integrate with CollabNet TeamForge must use the native UNIX/Linux authentication method, and not external authentication mechanisms such as NIS, NIS+, Winbind, Active Directory, or LDAP. TeamForge creates and manipulates local system accounts using the default `useradd`, `usermod`, `groupadd`, `groupdel`, and `userdel` commands. It expects to find any accounts or groups it created in `/etc/passwd` and `/etc/group`.

Create a loginfo trigger file

Here's now to create the loginfo trigger file.

1. Check out the module `CVSROOT` from the CVS repository.

   The `CVSROOT` module is automatically created in all new CVS repositories, and contains the `verifymsg` and `loginfo` files. When created, the files contain only comments.

2. Begin modifying the file by inserting `ALL` followed by a tab.

   This enables CollabNet TeamForge to see all modules in the repository.

3. Modify the path to your Java home directory as in this example and add it to the file, followed by a single space.

Example	Modification
`/usr/local/java/j2sdk1.4.1_01/bin/` `java -cp`	Replace `/usr/local/java/j2sdk1.4.1_01/` with the directory in which you have installed Java.

4. Modify the paths to the `.jar` files as shown here, then add them to the file separated by colons.

Examples	Modifications

`/yourdirectory/sourceforge_home/`
`lib/saaj.jar /yourdirectory/`
`sourceforge_home/lib/axis-1.1rc1-`
`va.jar /yourdirectory/`
	Replace `/yourdirectory/` with the directory in which you have installed CollabNet TeamForge .

Examples	Modifications

`sourceforge_home/lib/`
`saturn.jar /yourdirectory/`
`sourceforge_home/lib/jaxrpc.jar /`
`yourdirectory/sourceforge_home/`
`lib/externalintegration.jar /`
`yourdirectory/sourceforge_home/`
`lib/commons-discovery.jar /`
`yourdirectory/sourceforge_home/lib/`
`commons-logging.jar /yourdirectory/`
`sourceforge_home/lib/log4j-1.2.8.jar`	

5. Modify the parameters as follows:

Example	Modification
-Dintegration.name=exsy1001	Replace exsy1001 with the ID of your SCM server integration with CollabNet TeamForge . ☞ **Note:** To find the ID, click **Admin** > **Integrations** > **Integration Name**. The ID appears at the end of the URL.
-Dcvs.cvsroot=$CVSROOT	No change
-Dappserver.url=http://localhost:8080/ce-soap60/services/ScmListener	Replace localhost with the name of your machine running CollabNet TeamForge . ☞ **Note:** If needed, replace 8080 with your SOAP server port.
-Dcvs.username=$USER	No change
-Dlog4j.configuration=file:/yourdirectory/sourceforge_home/etc/externalintegration/log4j.xml	Replace `/yourdirectory/` with the directory in which you have installed CollabNet TeamForge .
-Dexternalintegration.triggers.log.dir=/tmp	No change

6. Add the following Java class containing the body of the trigger.

```
com.vasoftware.sf.externalintegration.triggers.cvstriggers.LogInfo
                       %{sVv}
```

7. Check your file against the completed example shown on Completed loginfo trigger file.

8. When you are finished, check the modified file back into your CVS repository.

If you have not already done so, repeat the process for the verifymsg trigger file.

☞ **Note:** If you make an error in the file, commits made to the repository will not fail, but you will receive error messages.

Create a modified verifymsg trigger file
Here's how to modify the verifymsg trigger file.

1. Check out the module CVSROOT from the CVS repository.

The CVSROOT module is automatically created in all new CVS repositories, and contains the verifymsg and loginfo files. When created, the files contain only comments.

2. Begin modifying the file by inserting .* followed by a tab.

This enables CollabNet TeamForge to see all modules in the repository.

3. Modify the path to your Java home directory as indicated below.

Add it to the file, followed by a single space.

Example	Modification
/usr/local/java/j2sdk1.4.1_01/bin/java -cp	Replace /usr/local/java/j2sdk1.4.1_01/ with the directory in which you have installed Java.

4. Modify the paths to the .jar files as shown in the following table.

Then add them to the file separated by colons.

Examples	Modifications
/yourdirectory/sourceforge_home/lib/saaj.jar / yourdirectory/sourceforge_home/lib/axis-1.1rc1- va.jar /yourdirectory/sourceforge_home/lib/ saturn.jar /yourdirectory/sourceforge_home/lib/ jaxrpc.jar /yourdirectory/sourceforge_home/ lib/externalintegration.jar /yourdirectory/ sourceforge_home/lib/commons-discovery.jar / yourdirectory/sourceforge_home/lib/commons- logging.jar /yourdirectory/sourceforge_home/lib/ log4j-1.2.8.jar	Replace /yourdirectory/ with the directory in which you have installed CollabNet TeamForge .

5. Modify the parameters as follows:

Examples	Modifications
-Dintegration.name=exsy1001	Replace exsy1001 with the ID of your SCM server integration with CollabNet TeamForge . ☞ **Note:** To find the ID, click **Admin** > **Integrations** > **Integration Name**. The ID is at the end of the URL.
-Dcvs.cvsroot=$CVSROOT	No change
-Dappserver.url=http://localhost:8080/ce-soap60/ services/ScmListener	Replace localhost with the name of your machine running CollabNet TeamForge . If needed, replace 8080 with your SOAP server port.
-Dcvs.username=$USER	No change
-Dlog4j.configuration=file:/yourdirectory/ sourceforge_home/etc/externalintegration/log4j.xml	Replace /yourdirectory/ with the directory in which you have installed CollabNet TeamForge .
-Dexternalintegration.triggers.log.dir=/tmp	No change

6. Add the following Java class containing the body of the trigger: com.vasoftware.sf.externalintegration.triggers.cvstriggers.VerifyMessage

7. Check your file against the completed example shown on Completed verifymsg trigger file.

8. Check the modified file back into your CVS repository.

9. If you have not already done so, repeat the process for the loginfo trigger file.

☞ **Note:** If you make an error in the file and check it back into your CVS repository, all commits made to the repository will fail. To correct the file, you cannot simply check it out and fix it. You must fix it on the repository. After fixing the file on the repository, you must then check it out, fix it again, then check it back in.

Change your site's look and feel
You can redesign some aspects of your site to suit your organization's needs and preferences.

These instructions support only some basic types of customization. Almost infinite varieties of customization are possible. To get into specific customization options in more detail, search or post a question on the TeamForge *discussion forum* or talk to your CollabNet representative.

Customize anything on your site

Follow these general instructions to customize a page, picture, text string, or other element on your site.

☞ **Important:** Custom branding changes can be overridden when your site is upgraded to a new version. You may have to reapply any look-and-feel modifications after an upgrade.

1. Download the sample branding files.

   Choose one of these files:

   • *Basic branding package*: Contains the files you need to do most of your branding tasks. Safest to use this file if you are doing your own branding.
   • *Advanced branding package*: Contains all the files that can be customized. For use when someone from CollabNet is helping you with your branding.

   ☞ **Note:** It is important that you have the most recent available version of this archive as a starting point. Check that the version number at the top of the `readme.txt` file in your copy of the branding package is the same as your version of the application. If it is not the same, check *www.collab.net* to see if there is a more recent version.

2. In the `look` project, check out the `branding` repository.

3. Copy the default version of the appropriate file from the branding zip file to the equivalent directory in your local copy of the `branding` repository.

4. Change the file to produce the results you want.
   For example:

   • To change a logo on your site's home page, overwrite the `home.gif` file with a new file of the same name.
   • To change a logo on a project home page, overwrite the `project.gif` file with a new file of the same name.

5. Commit the changed files into your site's `branding` repository.

   ☞ **Important:** Your `branding` repository does not have to contain all the files that are in the sample branding zip file, but the structure of your repository must be an exact mirror of the structure of the sample file set.
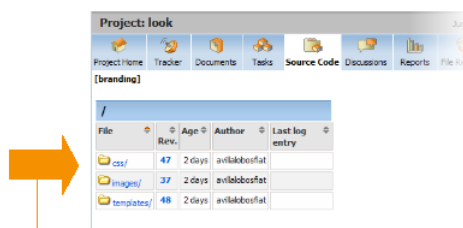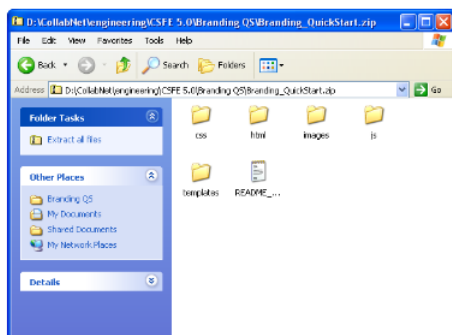
Check-in the customized branding files

Customize the home page of your site

To change the content of your site's main page, replace the `home.vm` file or add either `domain_home.html` or `DomainHome.html.html` file to the `html` folder in the `branding` repository.

☞ **Note:** For the general steps for changing the look and feel of a page, see *Customize anything on your site* on page 290.

The `\branding\templates\sfmain\home.vm` template controls the look, feel and structure of the standard home page. The default version allows users to log in and sign up for new user accounts, if CollabNet TeamForge is configured to allow user self-creation.

If the `DomainHome.html.html` or `domain_home.html` file is checked into the `branding` repository, the contents of the file are displayed as the site home page.

☞ **Tip:** If both `DomainHome.html.html` and `domain_home.html` files exist in the repository, the contents of the `DomainHome.html.html` are displayed.

Edit the `home.vm` template to produce the page you want.

You can change these objects on the site home page:

Object	Description
siteNews	The html block that shows site news. The html block itself is not customizable. ☞ **Note:** By uncommenting this object and commenting out "communityNews" object, site news can be displayed across all projects in the site.
communityNews	The html block that shows community news. ☞ **Note:** By uncommenting this object, and commenting out "siteNews" object, community news (news from the look project) can be displayed across all projects in the site.
mostActiveProjects	The html block that shows the most active projects. The html block itself is not customizable.
displayActivityGraph	A flag that indicates that the activity graph should be displayed.
displayTeamForgeLinks	A flag that indicates that CollabNet TeamForge quick links should be displayed.

Customize the home page of projects

To change the default main pages of the projects on your site, edit the `project_home.vm` file.

☞ **Note:** For the general steps for changing the look and feel of a page, see *Customize anything on your site* on page 290.

Edit the `project_home.vm` template to produce the project page you want.

You can change these objects on the project home page:

Object	Description
projectData	The object that contains the information about the project. It implements the interface com.collabnet.ce.customization.IProjectData.
adminList	The list of project administrators. Each object of the list implements the interface com.collabnet.customization.IUserRow.
memberList	The list of project members. Each object of the list implements the interface com.collabnet.customation.IUserRow.
projectMember	A flag that indicates that the user is a member of the project.
joinProjectButton	The button that contains the link to the Join Project page. It returns a com.collabnet.ce.customization.widgets.Button.

Object	Description
useCustomHomePage	A flag that indicates that the page shows the Wiki Home page instead of the standard Home page.
customHomePage	The html that displays as the Project Home page.
editCustomHomePageButton	The button that is used to edit the custom Home page. It returns a com.collabnet.ce.customization.widgets.Button.
projectAdmin	A flag that indicates whether or not the current user is a Project Admin.
useCustomProjectLogo	A flag that indicates that the Wiki project logo image will be used instead of the standard project logo.
customLogoPathString	The url from where the custom project logo image can be loaded.

Change your site's outgoing emails

When you site sends out automated emails, the text of the emails can be customized to fit your site's specific needs.

☞ **Note:** Before customizing your site, download the branding files. See *Customize anything on your site* on page 290.

You control screen labels and messages by overriding the resource bundle keys that specify the text strings that appear in Velocity macros and JSPs.

1. In your local copy of the `branding` repository, create a directory called `templates/mail`.

2. In the `templates/mail` directory, create a file containing the custom content for an email that the system sends out. Give the file the same name as the equivalent sample email file in the branding files package.
   For example, to override the email that is sent out to new members of the site, name the file `templates/mail/user_welcome.vm`. Use Velocity syntax to identify the parts of the email, like this:

   ```
   ##subject
   Welcome to our TeamForge site!
   ##subject
   ##body
   Here is the content that I want to appear in emails coming from my site...
   ##body
   ```

   ☞ **Note:** To customize a template in a specific language, identify the locale as an extension to the file name. For example, to create a user welcome file in Japanese, name the file `templates/mail/user_welcome_ja.vm`

3. Commit your new and changed files into the repository.

Provide external applications

If your site's users need access to an application or web site that is not part of CollabNet TeamForge, you can make it available by linking or integrating from within your CollabNet TeamForge site.

Provide a linked application

To bring an external tool into your environment quickly and easily, set it up as a linked application.

Create a site-wide linked application

When you create a site-wide linked application, it appears in all projects on your CollabNet TeamForge site.

Site-wide linked applications are especially useful for incorporating corporate standard external applications, such as a company intranet sites, into your CollabNet TeamForge installation.

1. Go to **My Workspace > Admin**.

2. On the site administration navigation bar, click **INTEGRATIONS** > **SITE-WIDE LINKED APPLICATIONS**.

3. Click **Create**.

4. On the **Create Site-wide Linked Application** page, provide a name for the linked application.
This name appears on the link in the CollabNet TeamForge navigation bar.

5. Enter the server location or URL for the linked application.

6. Select whether you want to enable single sign-on for the linked application.

   - If you use single sign-on, access to the linked application is managed through the CollabNet TeamForge authentication system. Users are not required to log into the linked application after they have logged into CollabNet TeamForge.
   - If you do not use single sign-on, users will be required to log in to the linked application using its native authentication system.

7. Choose how you want the linked application to appear when a user clicks it.

   - In the same window. The linked application takes over the entire browser window, replacing whatever the user was looking at.
   - In a new window. The linked application launches in a separate browser window.
   - In an iframe. The linked application appears in a box in the same window, framed by the CollabNet TeamForge site's header and navigation controls.

8. Click **Save**.

A link for the site-wide linked application is added to your CollabNet TeamForge navigation bar. Clicking the link displays the application in the main CollabNet TeamForge window.

Edit a site-wide linked application

When the use patterns of a linked application change, you may need to change the way the application integrates with CollabNet TeamForge .

1. Go to **My Workspace > Admin**.

2. On the site administration navigation bar, click **INTEGRATIONS** > **SITE-WIDE LINKED APPLICATIONS**.

3. Click the name of the site-wide linked application that you want to edit.

4. On the **Edit Site-wide Linked Application** page, make the changes you need.

   You can edit these elements:

   - The name of the application.
   - The application's URL.
   - Whether the application uses single sign-on.
   - Whether the application is displayed in a new window, in the same window, or in an IFrame.

   ☞ **Note:** You cannot change the application icon.

5. Click **Save**.

Provide integrated applications for use in TeamForge projects

To make a tool comprehensively available to your users, set it up as an integrated application.

Before you can make an integrated application available to project admins, your system administrator must integrate the application with your site. This may involve modifying the application. How this is done depends in part on the application.

Integrate an external application into a TeamForge site

When you integrate an external application into your CollabNet TeamForge site, your site's project administrators can choose to include the integrated application alongside the built-in tools in their projects.

When you have integrated the application, project administrators on your site can add it to their set of collaboration tools. Objects they create will share the core TeamForge features, such as authorization, authentication, go-urls, association, linkification, templating, Project Pages components, search, and source code management support.

1. Find the XML files that describes your integrated application.

   For an example of a working XML file, see *pebble-app.xml* on page 818.

2. Log into TeamForge as an admin user.

3. Go to **My Workspace > Admin**.

4. Click **INTEGRATED APPS** from the **Projects** menu.

5. Click **Create**.

6. Use the **Browse** window to find the configuration file you created, then click **Next**.

7. On the **Preview** screen, review the parameters you set in the configuration file.

☞ **Note:** You may have to revise one or more values to ensure they are valid.

8. Click **Save**.

The application is now available for all projects on your site. You can direct project administrators to *these instructions* to add it to their own project toolbars.

Enable or disable integrated applications
Site administrators can enable or disable integrated applications.

In TeamForge, enable an integrated application to make it available for use in the projects.

☞ **Tip:** You can also disable an integrated application to restrict it from being added to projects.

1. Go to **My Workspace > Admin**.

2. Click **INTEGRATED APPS** from the **Projects** menu.

3. To make the integrated application available for use, select the integrated application and click **Enable**.

4. To stop making the integrated application available for use, click **Disable**.
   The system asks for confirmation before disabling an integrated application. Click **OK** to disable the integrated application.

A disabled integrated application can be re-enabled when there is a need to use that integrated application in a project.

Remove an integrated application from your project
To stop making an application or site outside of CollabNet TeamForge available to your users from inside your CollabNet TeamForge projects seamlessly, disintegrate an application.

☞ **Note:** You may have integrated several external applications per project to maximize the integrated applications feature's utility. However, because each integrated application adds an icon to the project navigation bar, a large number of integrated applications can cause horizontal scrolling. Consider removing the integrated applications that are not in use.

1. Click **PROJECT ADMIN** from the **Project Home** menu.

2. On the **Project Admin Menu**, click **Tools**.

3. To remove an integrated application from the list of tools displayed , click the relevant Delete icon..
   The selected integrated application is removed from the project.

The icon of the removed integrated application disappears from your **Project Home** menu.

☞ **Note:** Removing an integrated application, removes all associations to the integrated application too.

Set site-level permissions for an integrated application
In CollabNet TeamForge, as the site administrator you can set the site-level permissions for an integrated application.

The default access permissions are usually set using the configuring xml file for each integrated application. However, you may want to provide permission access at site-level for some of the users.

1. Go to **My Workspace > Admin**.

2. Click **INTEGRATED APPS** from the **Projects** menu.

3. Set the permissions for the integrated application and click **Submit**.
   The permissions are set for the selected integrated application.

☞ **Note:** It is possible that an integrated application may be added to a project by a restricted user whose project administration permissions come from a global project role. In such a case, you could be left with no one authorized to administrate the integrated application. TeamForge handles that situation in one of two ways:

- If one or more users with the "Founder project administrator" role is a member of the project, then the restricted user who did the integration gets that role too.
- If there is no project member with the "founder project administrator" role, the restricted user who did the integration gets a new administrator role called "<integrated_application_name> Administrator."

View integrated application information

TeamForge site administrators, and users with site-wide roles and the IAF permission, can view information about an integrated application's configuration and other details.

1. Go to **My Workspace > Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. Click the name of the integrated application that you want to view.
   The default category tab, **GENERIC**, displays all the existing integrated applications associated with the site. You can have your integrated application displayed under a separate **CATEGORY** tab by defining it in the deployment configuration file.

The configuration details of the integrated application are displayed.

Edit an integrated application

TeamForge site administrators can update an integrated application's URLs (SOAP endpoint URL, administration URL, browser URL) and other parameters by uploading XML files containing the changes.

1. Go to **My Workspace > Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. Select the integrated application you want to edit.
4. Click **Edit**.
5. On the **Edit Integrated Application** page, make the changes you need.

   You can edit these elements:

   - The Deployment Configuration File.
   - The Application Configuration File.
6. Click the **Browse tab** and select the files to attach.
7. Click **Next**.

Export an integrated application

TeamForge site administrators can export an integrated application's variables in XML format and have them available for editing.

1. Go to **My Workspace** > **Admin**.
2. Click **INTEGRATED APPS** from the **Projects** menu.
3. To export the integrated application, select the integrated application name.
4. From the **View Integrated Application** page, click **Export**.

The integrated application will be exported.

Add a custom event handler to your TeamForge site

An event handler is a program that watches for events on a TeamForge site and communicates them to another system. You can add your own event handlers to the set that are built into TeamForge.

For example, some of your site's members may be using TeamForge alongside a legacy issue tracking system. You may want to write an event handler that listens to the "Artifact Create" event on your TeamForge site and sends the details about any newly created artifact to the legacy system through a webservice.

1. Create your custom event handler and package it as a `.jar` file.
2. Check with your system administrator that the *ENABLE_UI_FOR_CUSTOM_EVENT_HANDLERS* variable in the site configuration file is set to true.
3. Go to **My Workspace > Admin**.

4. Click **SYSTEM TOOLS** from the **Projects** menu.

5. Click **Custom Event Handlers**.

6. Click **Create** and use the **Browse** control to locate your `.jar` file.

7. Click **Add**.

> ☞ **Note:** If the system reports "Error Parsing Event Jar File," debug your event handler until the error message no longer appears.

Your `.jar` file is uploaded to your TeamForge site and the event cache is cleared. All the events you specified in your event handler are now captured and sent to the external web service.

System tools to monitor your site

To facilitate troubleshooting, keep an eye on key data about your CollabNet TeamForge site.

Check your server's status

As a first troubleshooting step, check the **Server Status** page to see if your servers are running and connected to the main CollabNet TeamForge server.

> ☞ **Note:** The **Server Status** page provides only the current status of each server. It is not a diagnostic tool, and you must correct any connection or configuration issues on the server in question.

1. Go to **My Workspace > Admin**.

2. On the site administration navigation bar, click **SYSTEM TOOLS**.

3. On the **System Tools** menu, click **Server Status**.

4. Find the status of the server you are interested in.

   For each server, you see one of the following status messages:

   - **OK** - The server is running and connected to the main TeamForge server.
   - **Could Not Connect** - The server is not running or is not connected to the main TeamForge server.

Read your site's logs

Inspecting the system logs for your CollabNet TeamForge site may product useful information for solving problems.

1. Go to **My Workspace > Admin**.

2. On the site administration navigation bar, click **SYSTEM TOOLS**.

3. On the **System Tools** menu, click **System Logs**.
   All the logs the server has written are listed.

4. On the **Site Configuration** page, click the log file you are interested in.

   For explanations of some of the most useful logs, see *Log files*.

Configure your site's log level

Change your site's log level in the **Configure Logging** page.

The **Configure Logging** page lets you change the application server (JBoss) log level. Changing the application server log level affects only the `server.log` and `vamessages.log` files.

1. Go to **My Workspace > Admin**.

2. On the site administration navigation bar, click **SYSTEM TOOLS**.

3. On the **System Tools** menu, click **Configure Logging**.

4. On the **Configure Logging** page, select either **INFO** or **DEBUG** from the **LOG LEVEL** drop-down list.

   > ☞ **Note:**
   > - The default log level is **INFO**.
   > - The `server.log` and `vamessages.log` files grow in size if you change the log level from **INFO** to **DEBUG**.
   > - JBoss restart is not required after changing the log level using the **Configure Logging** page.

5. Click **Save**.

Get build information

For solving some kinds of problems, it may be useful to know which software components are installed, the exact build number, and other technical information about your CollabNet TeamForge site.

1. Go to **My Workspace > Admin**.
2. On the site administration navigation bar, click **SYSTEM TOOLS**.
3. On the **System Tools** menu, click **Build Information**.
4. On the **Build Information** page, inspect the detailed information about the software underlying your instance of TeamForge.

Option	Description
Version	The release of the product.
Build Number	Number of times the software was compiled and packaged.
Operating System	This is always Linux.
Operating System Version	The variety of Linux in use. See *Software requirements for CollabNet TeamForge 8.2* on page 748 for eligible Linux distributions and versions.
Applied Patches	Identifiers of any software updates that have been installed to deliver added functionality or bug fixes.
Instance	Identifiers for any packages of customizations that have been applied. See *Customize a site*.
RPM	All the software packages that have been installed to support CollabNet TeamForge.

Get reports on site activity

You can keep track of various kinds of user activity, such as the number of user logins, source code commits, and tracker activity on your site.

- Monitor the total number of commits on your site by all users, in all projects, per day. You can track SCM commits to Subversion and CVS repositories, and to Git repositories as well, if you have the *TeamForge Git integration* set up.
- Check how many users log into your site in a given period. For example, to assess the impact of changes you've made to your site's look and feel (see *Customize anything on your site* on page 290), look for trends in the user login numbers for the week following the changes.
- Check the artifacts that are created and closed in a given period.

1. Go to **My Workspace > Admin**.
2. Click **REPORTS** from the **Projects** menu and select one of the available reports.
3. Click a **Zoom** option to see a week, a month, a quarter, 6 months, or a year of activity.

   The **Max** option shows all the data that's ever been collected.
4. Use the **From** and **To** date fields to zero in on the exact period of activity you are interested in.
5. Roll your cursor over any data point on the graph to see the exact numbers that determine that point's position.
6. Click **Grid View** to see the data as a table.

   You can toggle between chart and grid view as needed.

Keep your site secure

As site administrator, you are responsible for making sure passwords are effective and hackers can't get in.

Set password requirements

For security reasons, you may want to enforce a password policy that requires certain password characteristics.

You can specify the following characteristics:

- Minimum number of characters
- At least one number
- At least one non-alphanumeric character
- Mixed upper and lower-case characters
- Allowed invalid login attempts
- Disallow old passwords
- Use of dictionary words
- Default password expiry
- Disable/delete user accounts

- In the `site-options.conf` file, set the following variables.

  👉 **Note:** Default values are indicated by brackets.

- `MINIMUM_PASSWORD_LENGTH =[8]`
- `PASSWORD_REQUIRES_NUMBER = true | [true]`
- `PASSWORD_REQUIRES_NON_ALPHANUM = true | [true]`
- `PASSWORD_REQUIRES_MIXED_CASE = true | [true]`
- `REQUIRE_PASSWORD_SECURITY = true | [true]`
- `LOGIN_ATTEMPT_LOCK = 3`

  👉 **Tip:** This option controls locking out the user account after "n" invalid login attempts. Set this to zero or a negative number to lock the user account when the user provides an incorrect password for the first time. Set this to a positive number, say "2", to allow the user two wrong password attempts. The user account would be locked at the "x+1" (here, third) attempt. When a user's account is locked, either an administrator must unlock it or the user can use the "Forgot Your Password?" link to reset the password.

- `PASSWORD_HISTORY_AGE=5 | [5]`

  👉 **Tip:** The maximum allowed value of *PASSWORD_HISTORY_AGE* token is 10. This option disallows the previous "n" passwords, while setting a password. However, if this option is set to zero, a negative number or it is left empty, the user can use any previous password. The password being set must satisfy the existing password policy each time.

- `ALLOW_PASSWORD_DICTIONARY_WORD=true`

  👉 **Note:** You must set the "REQUIRE_PASSWORD_SECURITY" option to "true" in the `site-options.conf` file, for "LOGIN_ATTEMPT_LOCK", "PASSWORD_HISTORY_AGE" and "ALLOW_PASSWORD_DICTIONARY_WORD" security setting to take effect.

Prevent cross-site scripting (XSS)

By using the upload document function, an attacker could potentially upload an HTML page to CollabNet TeamForge that contains active code, such as JavaScript. This active code would then be executed by clients' browsers when they view the page.

To prevent an attack of this sort, you can specify whether or not HTML code is displayed in CollabNet TeamForge. This flag applies to all documents, tracker, task, and forum attachments, and files in the file release system.

In the `<SOURCEFORGE_SITE_DIR>/sourceforge_home/etc/ sourceforge_configuration.properties` file, set the `sf.safeDownloadMode` flag to one of the following values.

- `none`: All file types that can be displayed inline are displayed inline, including HTML and text.

  👉 **Note:** This is the CollabNet TeamForge default.

- `html`: HTML files are not displayed inline. All other file types that can be displayed inline are displayed inline, such as text. When a user tries to launch an HTML file, he or she is prompted to download and save the file locally.

- `all`: No file types are displayed inline. When a user tries to launch any file, he or she is prompted to download and save the file locally.

Support hardware provisioning

You can provide project managers with a TeamForge Lab Management site where project members can remotely build and test their code.

☞ **Tip:** For more information about the new features in TeamForge Lab Management 2.2, see *this introductory video*. (You'll need a free open.collab.net membership to watch it.)

Set up a new TeamForge Lab Management project

To start using TeamForge Lab Management project, create or select a TeamForge (or CollabNet Enterprise Edition) project to link to.

☞ **Note:** You can use Lab Management with either CollabNet TeamForge or CollabNet Enterprise Edition. These instructions assume you are using TeamForge. For information about working with CollabNetEnterprise Edition projects, see the *help for CEE project managers*.

1. Log into TeamForge Lab Management as an administrator.
2. Click the **Administration** tab.
3. On the **Administration** menu, click **Manage Projects**.
4. Select the **Add New Projects** tab.
5. If Lab Management is connected to a TeamForge server, the **Name** field displays a drop-down list of projects instead of requiring a text entry. Select the project you want to add to Lab Management.
6. In the **Summary** field, write a brief summary of the project.

   ☞ **Note:** The TeamForge Lab Management project summary may be different from the CEE or SFEE project summary.

7. Click **Add New** button.

Your project is added. You can now pass the project on to the Project Admin of the project to begin administrating and configuring.

☞ **Note:** If your project is an SFEE project, you will see that TeamForge Lab Management has added some Roles to your project.

Move hosts from one project to another

When a project has too many machines or needs more, you may wish to reallocate hosts between projects.

You must be a domain administrator to move hosts between projects.

1. Notify the project administrator of the project you are taking hosts from, so their project members won't lose important work which they may not have saved on that system.
2. Go to the **Project Home** page for the project which contains the host you are moving.
3. Select the **Hosts** tab to bring up the hosts list and click the host you want to re-allocate.
4. Click the **Admin** tab for the host.
5. From the **Project** drop-down box in the **Allocation Details** section, select the project that you want to move to.

   ☞ **Note:** If you do not see a drop-down box here, it means:

   - The host is a virtual guest. Virtual guests must remain in the same project as their parents.
   - The host contains virtual guests that are not all allocated to the same user. From the web interface, you can only change the project of a virtual host if the parent and all its guests are allocated to the same user.
   - The host is a cloud source.

6. Click **Update** to move the host (and its guests, if any).
7. If you are concerned about the security or contents of the machine that has been moved to the new project, rebuild the machine as soon as possible.

   This is the only way to ensure that only the exact software and users that you specified will be on the machine.

☞ **Note:** For a more flexible way of moving hosts between users and projects, you can use a TeamForge Lab Management API client to perform the move.

Forward a port

To make a service accessible to users, sometimes you may need to forward a port on a Lab Management box to an external-facing port.

☞ **Note:** Port forwarding is only needed if Lab Management has been setup in an isolated deployment mode. If you can access the nodes directly, without tunneling through the manager, then you do not need to worry about setting up port forwards.

1. In the left-hand column of the **Administration** screen, click **Port Forwarding.**

2. Select a host and specify the type of traffic you want to forward.

Host	The TeamForge Lab Management host that you want to access the network service from. For example, `cu014.lmdomain.yourdomain.net`
Host Port	The port on the Host -- the hostname you selected above -- that you want to access the network service on. For example, if you would like to forward a web server, you would select Host Port `80`.
External Port	The port on the TeamForge Lab Management Manager host that you would like to forward the traffic to. For example, if you would like to forward the above web server on Host `cu014.lmdomain.yourdomain.net` and Host Port `80` to port 8080 on the TeamForge Lab Management Manager host, you would enter `8080` in this field.

☞ **Note:** Only certain ports may be selected for the External Port. The list of ports is given at the top of the screen.

Protocol	The protocol or protocols to support when forwarding traffic. Choose one of these:

*   TCP
*   UDP
*   Both TCP and UDP

☞ **Note:** If you're unsure, ask the requestor for clarification. Most network services will only require TCP.

Description	A textual description of the service that is being forwarded. All users in the project will see this description when they view the Summary page for the Host.

☞ **Note:** The Description is mandatory.

3. Click **Add New** to add your new forwarding.

    The new forwarding is active immediately. To test your new forwarding, try accessing the external port on the TeamForge Lab Management Manager from your desktop.

    ☞ **Tip:** If the network service requires special client software, especially if the forwarding is using the UDP protocol, you may not be able to easily test it. In that case, contact the requestor and ask them to verify that the forwarding works properly.

Stop port forwarding

When you don't want to forward a port, you can delete or suspend forwarding for that port.

1. In the left-hand column of the **Administration** screen, click **Port Forwarding.**

2. If you want to remove the port forwarding permanently, click **Delete** for the forwarding you wish you delete.

    Forwarding for this port is immediately disabled and deleted forever.

    ☞   **Note:** You can always create a new forwarding with the same properties in the future.

3. If you think you might need this forwarding again soon, clear **Enable** and click **Update** for the forwarding you wish to disable.
    Enable the forwarding again by selecting **Enable** again and click **Update** again.

Set up host URL mapping

If you have real users accessing services on a box you have set up virtually with TeamForge Lab Management, consider giving them a human-friendly URL to go to. You can do this by mapping an arbitrary URL to the host's real URL.

☞   **Note:** Only TeamForge Lab Management domain administrators can administer host URL mappings.

1. On the **Administration** tab, click **Host URL Mapping** in the navigation bar.

2. Provide the following information about the service you wish to map:

Host	The TeamForge Lab Management host that you want to access the network service from. For example, `cu014.lmdomain.yourdomain.net`
Host Port	The port on the Host -- the hostname you selected above -- that you want to access the web service on. For example, if you would like to map a web server running on port 8080, you would select Host Port `8080.`
Host URL	The URL path on the Host that you want to map to. For example, if you want to map the URL `http://cu014.lmdomain.yourdomain.net:8080/build/`, you would enter `/build` for the Host URL.
Mapped URL	The location (URL path) on the TeamForge Lab Management manager to map the Host URL to. The location will be under the directory `/apps`. For example, if you enter `mybuild` for Mapped URL in our example, you will then be able to reach `http://cu014.lmdomain.yourdomain.net:8080/build/` as `http://mgr.lmdomain.yourdomain.net/apps/mybuild/`. Valid characters for Mapped URL's are: Letters, numbers, spaces, underscores, and hyphens.
Use SSL	Whether or not to force requests for the mapped URL to use SSL. Mapped URL's will always be accessible via SSL, even if this box is not checked. But, if you check this box, it will be impossible to access this URL without SSL. In general, unless there is a very good reason for turning the `Use SSL` option off, we recommend keeping it on.

Dynamic Rewriting	Set to either **Safer** or **More Aggressive**. Using the **Safer** option, only HTML attributes which are known to contain URL's are dynamically rewritten. But in some cases, especially with applications that use embedded Javascript code, this will not be sufficient and some parts of your application will not work properly. In this case, try the **More Aggressive** option. This option still attempts to be careful about what it rewrites, but looks into the HTML more deeply and will rewrite a wider variety of HTML tags and attributes.
Description	A textual description of the web service that is being mapped. All users in the project will see this description when they view the Summary page for the Host.

    ☞ **Note:** The Description is mandatory.

3. Click **Add New** to add your new URL mapping.

The new URL mapping is active immediately. To test it, try accessing the resulting URL from your desktop.

Stop URL mapping

When you don't need a URL mapped, it's a good idea to remove the mapping.

☞ **Note:** Only TeamForge Lab Management domain administrators can administer host URL mappings.

1. On the **Administration** tab, click **Host URL Mapping** in the navigation bar.

2. Click **Delete** for the URL mapping you want to stop.

The URL mapping is immediately disabled and deleted forever.

☞ **Note:** You can always create a new URL mapping with the same properties in the future.

☞ **Tip:** To temporarily disable a URL mapping, clear **Enable** and click **Update** for the URL mapping you wish to disable. To enable the URL mapping again, select **Enable** and click **Update** again.

Delete a project

To conserve resources on your TeamForge Lab Management site, it's a good idea to delete projects that are not in use.

Only TeamForge Lab Management Domain Admins can delete and undelete projects. When a project has been deleted, only TeamForge Lab Management Domain Admins can access it, and then only to undelete it.

Projects that contain hosts cannot be deleted.

1. Check that there are no hosts allocated to this project.

You cannot delete the project if there are hosts currently allocated to it.

2. On the **Projects** tab, find the project you want to delete, and select **Edit** in the Actions column on the right for that project."

3. On the **Edit Project** page, click **Delete**, then **Confirm Delete**.

The project is deleted from the system.

Undelete a project

To bring a project back into your TeamForge Lab Management site, you can undelete the project.

Only TeamForge Lab Management Domain Admins can delete and undelete projects. When a project has been deleted, only TeamForge Lab Management Domain Admins can access it, and then only to undelete it.

1. On the **Administration** tab, click **Manage Projects** in the navigation bar.
In the list of all the projects in the system, deleted projects are in italics and are marked "Currently deleted.".

2. Click the project you want to undelete.

3. On the **Edit Project** page for the deleted project, click **Undelete** button, then **Confirm Undelete**.

The project is restored to being a fully active TeamForge Lab Management project.

Expand your domain with virtual guests

By using virtual guests, you can provide your users with many more machines than you actually have.

Create a virtual guest

To add a virtual guest, go to the configuration page for the host where the guest will live.

☞ **Note:** You can only create virtual guests on machines that are allocated to you, are not guests themselves, and are not marked immutable.

1. On the **Manage Hosts** page, (or any other page where the host is visible), click the down arrow next to the host where your virtual guest will live and select **Manage Virtual Guests**.

2. On the host's **Manage Virtual Guests** page, specify the details for your new virtual guest under **Add New Virtual Guest**.

Migrate a virtual guest

When a virtual guest consumes too many system resources, or you wish to rebuild or perform other more drastic maintenance on the virtual guest's current virtual host's use, migrate the guest to a different host.

Before moving a virtual guest, make sure the following is true:

• You are a TeamForge Lab Management domain administrator.
• The current virtual host and guest, and the desired new virtual host, are all in the same project.
• The target virtual host has enough RAM, disk, and CPU to accommodate the virtual guest you wish to migrate to it. (TeamForge Lab Management displays a drop-down list of hosts that fulfill all these criteria.)
• The virtual guest being moved and the target virtual host are on the same subnet.

1. Go to the **Host|Admin** page for the virtual guest that you wish to migrate.

2. In the **Move to Virtual Host** drop-down box, select the host you wish to migrate this virtual guest to, and click **Update Allocation Details**.

☞ **Note:** If you do not see this drop-down box, but instead see the text "No hosts are eligible for re-parenting," there are no other virtual hosts in this project that could accommodate this virtual guest. You must delete or reduce the hardware resources assigned to one or more existing guests.

3. Click **Confirm Update**.

The virtual guest immediately shuts down and begins migrating. The owner of the virtual guest (not necessarily the user that requested the migration) receives an e-mail when the migration is complete.

☞ **Note:** There is no exact time frame or estimate for how long a guest migration takes. A typical "base" system without much data (e.g., under 10GB) usually migrates in 5-15 minutes. The time required is mostly dependent on the size of the virtual machine, coupled with the load on the source and destination virtual hosts.

Delete a virtual guest

You delete a virtual guest the same way you delete a physical host.

☞ **Important:** Before you delete your guest, be aware of one very important difference: As soon as you delete a virtual guest, the virtual guest system and all its data are wiped out forever! This is in contrast to deleting a physical host, in which case the machine gets removed from TeamForge Lab Management, but the contents of its hard drive are not wiped out, and could still be rescued by an administrator with physical access to the system.

1. On the **Host|Admin** screen, click **Delete Host**.

2. Click **Confirm Delete**.

The virtual guest is permanently deleted.

Manage profiles

Like hosts and projects, profiles must be added at the domain level before they can be used by individual projects. Likewise, they can be administratively disabled from use at the domain level.

Add a profile to the Lab Management domain

In Lab Management, a "profile" is a reproducible collection of software packages that comprise a runnable operating system and accompanying set of applications. Every system inside the Lab Management environment has an operating system profile installed on it.

There are three ways to add a new profile to Lab Management :

- Directly add it to the profile repository using a Subversion client, then import it into Lab Management through the **Add Profile** page.
- Create the Profile XML file directly, and upload the file through the **Add Profile** page.
- Make a copy of an existing profile on that profile's Admin page, and edit it to suit your needs.

Among the three, the third is the easiest, and used most often by our own profile creators.

1. Log into TeamForge Lab Management as a user with TeamForge Lab Management Domain Admin privileges.
2. Click the **Administration** tab.
3. On the **Administration** menu, click **Manage Profiles** and then the **Add New Profile** link.
4. Enter the following information.

   ☞ **Note:** All fields are mandatory.

Profile name	The name of the profile. Must already exist in the site's Subversion repository before it can be added into TeamForge Lab Management. This name can never be changed, so be sure to get the profile named properly when you initially create it.
	You can also upload a profile from your local system into Lab Management's Subversion repository. The profile must be in valid Lab Management profile XML format.
Summary	A brief summary of the profile. The Summary can be changed later on using the Profile Admin page.
Project	What project the profile belongs to. If you leave it blank, the profile will belong to the domain, and either be accessible to *all* projects (if the profile is public) or *no* projects (if the profile is private). This property can be changed later on using the Profile Admin page.
Public	Whether the profile is public (can be used by any project) or private (can only be used by the project which owns the profile). This property can be changed later on using the Profile Admin page.
Prebuilt	Whether or not the profile is a VMware or other prebuilt image file. *This profile property cannot be altered once the profile is created.* If your profile is an image file, this will be Yes. If your profile installs via a network install method like Kickstart(tm) or Jumpstart(tm), this will be No.
Size (in GB)	The minimum disk size required to build and run this profile. The minimum disk size should be at least 10 GB.

5. Click **Add New** to add the new profile.

Modify the XML definition for a profile

Rather than using a Subversion client to check out and edit a profile, you can edit the profile XML definition in a text box in the Lab Management user interface. This allows you the greatest flexibility in changing parts of the profile.

1. Log into TeamForge Lab Management as a user with TeamForge Lab Management Domain Admin privileges.

2. Click the **Administration** tab.

3. On the **Administration** menu, click **Manage Profiles**.

4. Right-click a profile and select **Admin** from its context menu.

5. In the Modify Profile: Profile XML Definition section, click **View/Edit XML Definition of Profile**.
   This brings up the content of the profile XML definition in an editable text box.

6. To edit the content, we recommend that you copy it to a text editor, edit and save it. Then copy it back into the text area.

7. To double-check your changes your changes before checking in the new version, click **Show Diff**.

8. When you're done, click **Check In New Version**.

   If, by some chance, you made a mistake and the profile is invalid, don't worry -- since profiles are versioned, you can always mark that version as unbuildable.

Stop using a profile

In Lab Management, profiles can be permanently deleted or disabled.

Delete a profile

If a profile in Lab Management has never been used, it can be deleted. This can be helpful, for example, if you wish to clean up some profiles that were created with incorrect names.

1. Log into TeamForge Lab Management with TeamForge Lab Management Domain Admin privileges.

2. Click the **Administration** tab.

3. On the Administration menu, click **Manage Profiles**.

4. In the list of all profiles currently in the system, click the profile you wish to disable.

5. Click the **Admin** tab.

Disable a profile

When you disable a profile in Lab Management, it will prevent the users from building or rebuilding a host with that profile. It is possible to make disabled Lab Management profiles accessible again.

A disabled profile gets removed from all projects that currently allow the profile. If you disable a profile and then enable it later, project managers who previously allowed that profile will have to go back into their project's Profile Library and add the profile again.

☞ **Important:** Use caution before disabling a profile. Contact the owners of the systems using that profile, and owners of projects that include that profile, before disabling the profile.

1. Log into TeamForge Lab Management with TeamForge Lab Management Domain Admin privileges.

2. Click the **Administration** tab.

3. On the Administration menu, click **Manage Profiles**.

4. In the list of all profiles currently in the system, click the profile you wish to disable.

5. Click the **Admin** tab.

6. Click **Disable Profile**.

The profile is disabled and removed from all projects that currently allow it.

Enable a profile

To build a machine with a profile that has been disabled, you must re-enable the profile.

1. Log into TeamForge Lab Management with TeamForge Lab Management Domain Admin privileges.

2. Click the **Administration** tab.

3. On the **Administration** menu, click **Manage Profiles**.

4. In the list of all profiles currently in the system, click the profile you wish to re-enable.

5. Click the **Admin** tab.

6. On the **Admin** page, click **Enable Profile** to re-enable the profile.

Administer clouds

As an administrator, you can set up and manage the clouds in your Lab Management site.

Add a new cloud

When you pool the physical resources into a cloud, they can be partitioned in to different virtual machines and used in any project on the domain that needs them.

1. Select **Administration** > **Clouds**

2. In the **Manage Clouds** page, click **New Cloud**.

3. Enter a name and brief summary for the cloud.

4. Enter any additional details in the **Description**.

5. Select an owning project for the cloud.

   The owning project allows the physical source systems in the cloud to be from that project. If you don't specify an owning project, the sources for the cloud can only be systems that are unallocated and don't belong to any project .

6. Specify whether this is a public or private cloud.

   • A public cloud appears in any project on the site that allows clouds, and is available for other project administrators to include in their list.

   • A private cloud appears only in the project that owns it, so that only the users in that project can allocate systems from it.

7. Specify whether hosts allocated from this cloud should be deleted when they are deallocated.

8. Specify the maximum time (in days) for which a host can be allocated from this cloud.

   The default value is 0 and indicates that there is no limit.

9. Click **New Cloud**.

When your cloud gets created, you can proceed to add hosts to it.

Add a cloud host

When you've created a cloud, you can add hosts from physical sources or Amazon EC2 sources.

1. On the **Manage Clouds** page, click a cloud name and select **Admin**.

2. Add a cloud source using either of these options:

   a) To add a physical host source, select one from the drop-down and click **Add Host**.

      Physical hosts can be partitioned into virtual machines that can be allocated into different projects on the site.

   b) To add an Amazon EC2 source, click **Add EC2 Source**.

      ☞ **Note:** You will need information from your *Amazon Web Services* account to complete this task.

      1. Add a description for the source.

      2. Specify the maximum number of hosts that may exist at any given time from the set of EC2 credentials specified on this page.

      3. Fill in the EC2 credentials from the information in your Amazon Web Services account. See *Set up Amazon EC2 credentials* on page 308 for details on how to get this information - you'll need to provide the API Key, API Secret, and the Keypair Name and Keypair Value (your RSA private key).

3. Click **Save**.

When you add an EC2 source, make sure that one or both EC2 profiles - Amazon_EC2_FC4_base_i386, Amazon_EC2_FC8_base_i386 - are enabled for your Lab Management projects.

Define a host type

To add a host type for a cloud, specify its attributes such as CPU, disk size and memory, and price for usage in an XML configuration file.

1. On the cloud's **Admin** page, click **Modify Cloud**.

2. Edit the XML fille.

Here's an example:

```
<host_type  value="Sun">
        <contains>
            <size  value="extra_small">
                <price base="0.1"   />
                <provides>
                    <memsize  value="256" />
                    <disksize  value="10" />
                    <ncpu  value="1" />
                </provides>
            </size>
            <size  value="small">
                <price base="0.2"   />
                <provides>
                    <memsize  value="512" />
                    <disksize  value="10" />
                    <ncpu  value="1" />
                </provides>
            </size>
            <size  value="medium">
                <price base="0.3"   />
                <provides>
                    <memsize  value="1024" />
                    <disksize  value="20" />
                    <ncpu  value="1" />
                </provides>
            </size>
            <size  value="large">
                <price base="0.4"   />
                <provides>
                    <memsize  value="1024" />
                    <disksize  value="40" />
                    <ncpu  value="1" />
                </provides>
            </size>
            <size  value="extra_large">
                <price base="0.5"   />
                <provides>
                    <memsize  value="2048" />
                    <disksize  value="80" />
                    <ncpu  value="1" />
                </provides>
            </size>
        </contains>
        <requires category="cpuarch" op="in">
            sun4u
            sun4v
        </requires>
        <requires category="type" op="eq">
            physical
        </requires>
    </host_type>
```

3. Click **Update**.

Delete a cloud

When a cloud, and its hosts, are no longer in use, you can delete it.

☞ **Note:** You can delete a cloud only when all of its physical sources have been removed, and there are no hosts that were created from the cloud.

1. On the **Clouds** page, click on a cloud name and select **Admin**.

2. Click **Delete Cloud**.

Set up Amazon EC2 credentials
To be able to create an EC2 cloud source in Lab Management, you need to set up an Amazon Web Services account and supply the EC2 credentials to Lab Management.

To get started, you need to create an *Amazon Web Services* (AWS) account (or sign in with an existing Amazon account). You will be required to provide a credit card as part of sign up; this card will be charged for all your running EC2 instances. You will also be required to provide a callback number to which you have ready access. Amazon will generate an automated callback to this number to complete registration, and ask you to enter a generated PIN into your phone. Finally you'll receive a verification email letting you know that your account is set up.

1. Go to `https://console.aws.amazon.com/ec2/` to view the AWS console.

    ☞ **Note:** You should default to the US East region. This is the default that Lab Management assumes for all AMIs, instances, and key pairs.

2. Get the API credentials you'll give Lab Management to use your AWS account.

    a) In **My Account** > **Security Credentials**, scroll down to your Access Keys, and get the **Access Key ID** (known as API Key in Lab Management).
    b) Show the **Secret Access Key** (known as API Secret in Lab Management).

3. Generate or upload a **Key Pair**.

    This is a named public/private key pair that will be used to access instances as root.

    ☞ **Note:** This key pair is associated with this AWS account, and therefore the credit card and the user that creates a cloud in Lab Management - take the appropriate precautions!

    Amazon stores only the public key for passwordless SSH access to the instances. If you generate a key pair, you'll download the private key, which you'll be responsible for keeping secure. If you upload your own public key, you're still responsible for keeping your private key secure. For Lab Management-controlled EC2 instances, this key is used for paswordless root SSH access to the instance.

When you *create an EC2 cloud source in Lab Management*, you'll need to provide the API Key and API Secret viewed above, as well as the Keypair Name and Keypair Value (your RSA private key) generated above.

Reference information about CollabNet TeamForge 8.2

This is some information about the scripts, configuration files and log files you can use to work with CollabNet TeamForge.

Reference information about your CollabNet TeamForge site

Here is some detailed information to help you make the most of TeamForge.

Wiki syntax
You can use special wiki text markup to do a wide variety of cool and useful things on TeamForge wiki pages.

For a quick introduction, see the *documentation for JSPwiki*, which is the wiki engine that drives the TeamForge wiki tool.

☞ **Note:** These tools are for use only when you are editing a wiki page in text mode. If you try to use them in WYSIWYG mode, they are displayed just the way you typed them in, which is not what you want.

Text effects on wiki pages
Wiki markup is great for making text look the way you need it to look.

☞ **Note:** These tools are for use only when you are editing a wiki page in text mode. If you try to use them in WYSIWYG mode, they are displayed just the way you typed them in, which is not what you want.

Syntax	Effect	Details
`----`	Creates a horizontal rule.	
`\\`	Creates a line break.	
`!!!text`	Creates a level 1 (large) header.	
`!!text`	Creates a level 2 (medium) header.	
`!text`	Creates a level 3 (small) header.	
`''text''`	Creates italic text. (That's two single quotes on each side.)	
`__text__`	Creates bold text. (That's two underscores on each side.)	
`{{text}}`	Creates monospaced text.	
`*text`	Creates a bulleted list item.	
`#text`	Creates a numbered list item.	
`;term:ex`	Creates a definition for the word "term"'with the explanation "ex."	
`{{{text}}}`	Creates pre-formatted text.	
`%%( <css-style> )<your text>%%`	Defines a CSS style command.	`%%( font-size: 150%; color: red; ) Hello, world!%%`
`Blank line`	Starts a new paragraph.	

Bring TeamForge data into wiki pages

Use these markup formats to bring in data from elsewhere on your TeamForge site.

☞ **Note:** These are for use only when you are editing a wiki page in text mode. If you try to use them in WYSIWYG mode, they are displayed just the way you typed them in, which is not what you want.

Syntax	Effect	Details
`[{INSERT ExcelToHTMLPlugin src='c:\somesheet.xls'}]` or `[{INSERT ExcelToHTMLPlugin border='1' src='\\the_server \somesheet.xls'}]`	Reads a Microsoft Excel file and displays it as an HTML table.	Parameters: • src: URL / Attachment file name • srcsheet: Sheet name • height: height attribute for the html table • width: width attribute for the html table • border: border attribute for the html table More at *http://www.ecyrd.com/ JSPWiki/wiki/ExcelToHTMLPlugin*

Text navigation tools for wiki pages

You can use wiki syntax to help readers get around.

☞ **Note:** To use these tools, copy and paste the sample syntax into your Wiki page in "Plain Editor" mode, then customize it appropriately.

Syntax	Effect	Details
`%%insert-toc %%`	Creates a table of contents consisting of the header text on the page.	
`[link]`	Creates a link to a new Wiki page called "link."	If the link is a complete URL, a link to the URL is created. If the link points to a .gif, .jpg, or .png image, the image is rendered directly in the page.
`[title|link]`	Creates a link to a new Wiki page called "link" with the text "title" displayed for the URL.	If the link is a complete URL, a link to the URL is created. If the link points to a .gif, .jpg, or .png image, the image is rendered directly on the page with "title" as ALT text.
`~TestText`	Disables link creation for a CamelCase word.	CamelCase words are two or more uppercase words with no spaces. By default, a CamelCase word automatically creates a link to a new Wiki page.
`[[link]`	Creates the text "[link]."	
`[{IFramePlugin url='http:// open.collab.net/' width='100%' height='500' border='1' scrolling='yes' align='center'}]`	Embeds an iframe into a wiki page.	• attachment: Attachment path, e.g. 'IFramePlugin.jar(info)' • url: A URL, e.g 'http://www.google.com' • align: Align the iFrame to left/center/right • border: Whether there is a border or not • width: Width of the iFrame • height: Height of the iFrame • marginwidth: Margin width of the iFrame • marginheight: Margin height of the iFrame • scrolling: Whether the iFrame can be scrolled or not See *http://www.ecyrd.com/JSPWiki/wiki/iFramePlugin*
`Tab Completion <Keyword +Tab>`	You can type a keyword and hit the Tab key (under the Tab Completion mode). The editor will fill in with a sample template for the specific markup represented by the keyword.	• link:Inserts a sample link. • h1:Inserts level 1 heading sample. • h2:Inserts level 2 heading sample. • h3:Inserts level 3 heading sample.

Syntax	Effect	Details
		• bold:Inserts a bold text sample. • italic:Inserts an italics text sample. • mono:Inserts a mono text sample. • mono:Inserts a mono text sample. • sup:Inserts a superscript sample. • sub:Inserts a subscript sample. • strike:Inserts a strike through text sample. • br:Inserts a line break. • hr:Inserts a horizontal line. • pre:Inserts a pre-formatted text sample. • code:Inserts a code block sample. • dl:Inserts a definition list block sample. • toc:Inserts the Table of Contents plugin syntax. • tab:Inserts a sample tabbed section block syntax. • table:Inserts a sample table syntax. • img:Inserts a sample image plugin syntax. • quote:Inserts a sample quoted text block. • sign:Inserts the user's signature.

Attachments for wiki pages

You can use wiki markup to bring in information from external sources.

> ☞ **Note:** These tools are for use only when you are editing a wiki page in text mode. If you try to use them in WYSIWYG mode, they are displayed just the way you typed them in, which is not what you want.

Syntax	Effect	Details
`[WikiPageName/ attachmentName]`	Embeds an attachment in the page.	If the attachment is a .gif, .jpg, or .png image file, the attachment will be embedded in the page; otherwise, the name of the attachment will display as a downloadable link. After adding attachments, the exact syntax for including the current page's attachments is shown next to each attachment's name in the Attachments section of the Edit Wiki page. You can use the same syntax to embed attachments from other wiki pages in the same project.
`[{InsertAttachment`	Inserts the contents of an attachment (text file) into a page.	If the attachment is a text file, the content of the text file is inserted into

Syntax	Effect	Details

`page='WikiPage/`
`attachment'}]`
		the page. For more information, *click here*.
		
		☞ **Note:**
		  • This markup is for inserting content of a text file.
		  • Files with the following extensions are allowed as attachment: .txt, .html, .xml, .cpp and .java.
		  • In addition to image files, inserting files with the following extensions is not supported by this markup: .doc, .xls, .pdf, .zip, .jar.

`[{Mediaplayer` `src='fileName.wmv'}]`	Embeds a Windows Media Player or Quicktime Player on a wiki page.	• src: Media URL / Attachment file name • playertype: "mediaplayer" / "quicktime" • width, height: Dimension of the embedded media displayed • movieheight, moviewidth: Dimension of the display screen • caption: Caption to be displayed below the media player • control: Displays Control bar. mediaplayer: 1 (Show) / 0 (Hide); quicktime: true (Show) / false (Hide) • autostart: Play automatically. mediaplayer: 1 (Auto) / 0 (Manual, Click to play); quicktime: true (Auto) / false (Manual, Click to play) • autorewind: Automatically rewinds when play ends. mediaplayer: 1 (Auto Rewind) / 0 (Play once); quicktime: true (Auto Rewind) / false (Play once) • playcount: Number of times the movie will play. 0 represents always play. See *http://www.ecyrd.com/JSPWiki/wiki/MediaPlayerPlugin*

`[{INSERT`
`ExcelToHTMLPlugin`
`src='WikiPage`
`\somesheet.xls'}]`
`or`
	Reads a Microsoft Excel file and displays it as an HTML table.	Parameters:
		
		• src: URL / Attachment file name
		• srcsheet: Sheet name

Syntax	Effect	Details

`[{INSERT`
` ExcelToHTMLPlugin`
`border='1'`
`src='\\the_server`
`\somesheet.xls'}]`
		• height: height attribute for the html table
		• width: width attribute for the html table
		
		More at *http://www.ecyrd.com/ JSPWiki/wiki/ExcelToHTMLPlugin*

`[{arnaud.Flash` `src='yourAttachedFlash.swf'}]`	Embeds a Flash Player on your wiki page.	• width='n' • height='n' • controls='true|false' • play='true|false' • loop='true|false' • parameters='param1=value1, &param2=value2' See *http://www.ecyrd.com/JSPWiki/ wiki/AFlashPlugin*

`[{IFramePlugin`
`url='http://`
`open.collab.net/'`
`width='100%'`
` height='500'`
`border='1'`
` scrolling='yes'`
`align='center'}]`
	Embeds an iframe into a wiki page.	• attachment: Attachment path, e.g. 'IFramePlugin.jar(info)'
		• url: A URL, e.g 'http:// www.google.com'
		• align: Align the iFrame to left/ center/right
		• border: Whether there is a border or not
		• width: Width of the iFrame
		• height: Height of the iFrame
		• marginwidth: Margin width of the iFrame
		• marginheight: Margin height of the iFrame
		• scrolling: Whether the iFrame can be scrolled or not
		
		See *http://www.ecyrd.com/JSPWiki/ wiki/iFramePlugin*

Tables on wiki pages

You can use wiki markup to organize information in tables on a wiki page.

☞ **Note:** These tools are for use only when you are editing a wiki page in text mode. If you try to use them in WYSIWYG mode, they are displayed just the way you typed them in, which is not what you want.

Syntax	Effect	Details

`[{Table <table-`
`parameters>`
 	
`|| Table Header Example`
` || More...`
`| Table Data Example |`
` More...`
`}]`
	Inserts a table on the wiki page. See this page for the markup for table elements.	• `rowNumber: <integer>` Row number starts counting at this value; default = 0 (used in conjunction with '#' syntax)
		• `style: <css-style>` Add formatting to the table,

Syntax	Effect	Details
		e.g. `style:'border=2px solid black;'` • `dataStyle: <css-style>` Format all data cells (prefixed by single pipe signs '|') • `headerStyle: <css-style>` Format all header cells (prefixed by double pipe signs '||') • `evenRowStyle: <css-style>` Format the even rows, e.g. evenRowStyle='background: #ffff00;' • `oddRowStyle: <css-style>` Format the odd rows, e.g. oddRowStyle='color: red;'
`||head1||head2`	Creates a table column with header text "head1" in the first cell and "head2" in the second cell.	
`|col1|col2`	Creates a table row containing the text "col1" in the first cell and "col2" in the second cell.	
`|<`	Collapses a cell with the previous cell so it spans multiple columns.	*http://www.ecyrd.com/JSPWiki/wiki/TablePlugin*
`||<`	Collapses a header cell with the previous header cell so it spans multiple columns.	*http://www.ecyrd.com/JSPWiki/wiki/TablePlugin*
`|^`	Collapses a cell with the cell above so that it spans multiple rows.	*http://www.ecyrd.com/JSPWiki/wiki/TablePlugin*
`||^`	Collapses a header cell with the header cell above so that it spans multiple rows.	*http://www.ecyrd.com/JSPWiki/wiki/TablePlugin*
`|( <css-style> )`	Inside a table cell, adds specific formatting to a cell.	*http://www.ecyrd.com/JSPWiki/wiki/TablePlugin*
`#`	Inside a table cell, displays the current row number with auto row numbering.	*http://www.ecyrd.com/JSPWiki/wiki/TablePlugin*

`[{INSERT ExcelToHTMLPlugin src='WikiPage\somesheet.xls'}]`
or
`[{INSERT ExcelToHTMLPlugin border='1' src='\\the_server\somesheet.xls'}]`
	Reads a Microsoft Excel file that is attached to a wiki page, and displays it as an HTML table.	Parameters:
		• src: If the Excel file is attached to the current wiki page, this is the attachment file name. If it is attached to some other wiki page, this is the URL of the attachment.
		• srcsheet: Sheet name
		• height: height attribute for the html table
		• width: width attribute for the html table

Syntax	Effect	Details
		• border: border attribute for the html table
		• cellpadding: cellpadding attribute for the html table
		• cellspacing: cellspacing attribute for the html table
		• background: background attribute for the html table. Attachment file name is accepted as value
		• backgroundcolor: backgroundcolor attribute for the html table
		• keepformat: Formating specified in the excel sheet is applied for the html table. Note: The complete formating from excel sheet is not applied to the html table, for example, font, font size, etc are not applied and background color, foreground color, etc are applied. yes/no is accepted as value
		• headercolor: Foreground color for the header (eg: #dcdcdc)
		• headerbackgroundcolor: Background color for the header (eg: #adadad)
		• evenrowcolor: Foreground color for the even rows (eg: #adadad)
		• evenrowbackgroundcolor: Background color for the even rows (eg: #adadad)
		• oddrowcolor: Foreground color for the odd rows (eg: #adadad)
		• oddrowbackgroundcolor: Background color for the odd rows (eg: #adadad)
		• tableclass: Style class name for the HTML table
		• headerclass: Style class name for header
		• evenrowclass: Style class name for the even rows
		• oddrowclass: Style class name for the even rows
		• stylesheet: Stylesheet for the table. Attachment file name is accepted as value
		See *http://www.ecyrd.com/JSPWiki/ wiki/ExcelToHTMLPlugin*

Syntax	Effect	Details
`[{TableOfContents }]\\`	Creates a table of contents consisting of the header text on the page.	

Wiki plugins

You can use pre-defined wiki plugins to format your current page.

Plugin Name	Details
Section Headings Template	Inserts a page template that includes a table of contents and several section headings. Typically, this can be used in a new or blank page.
Sortable Table	Inserts a new table that can be sorted when you click on the column headers.
Zebra Table	Inserts a new table that has alternating background colors for each row.
Table of Contents Plugin	This plugin automatically generates table of contents that provides links to all the headings on your page.
Insert Page Plugin	This plugin will insert a copy of another page into the current page. You must specify the name of the page to insert.
Current Time Plugin	This plugin displays the current date and local time of the server when the page is viewed.
Insert Attachment	The insert attachment plugin allows you to insert the contents of an attachment into a page.
Media Player	This is an embedded player in your wiki page, supporting Windows Media Player and Apple QuickTime.
Insert Table	Additional table support including multi-line table editing, cell merging, and automatic row numbering.
Flash Player	An embedded flash player for the wiki page.
Insert Excel	Allows you to insert a Microsoft Excel file as a table.
Insert iFrame	Allows embedding attachments, external urls, and files (relative to the docbase).
Code to HTML	Allows source code syntax to be rendered as a HTML output and supports syntax from 130 different programming languages.
Index Plugin	Displays all the pages in wiki in a alphabetical order.
Recent Changes Plugin	Inserts the latest changes in order.
Referred Pages Plugin	Finds and lists all the pages that are referred to by the current page. The depth parameter allows to display a recursive tree of referred pages.
Referring Pages Plugin	Finds and lists all the pages that refer to the current page.

Plugin Name	Details
Undefined Pages Plugin	Lists all the pages that are referred to, but not yet created.
Unused Pages Plugin	Lists all the pages that are not currently referred to by any other page.
ExcelToHTMLPlugin	Provides a HTML view for the Excel spread sheet files.
PDFPlugin	Provides the PDF output for the files.
WikiContentToHTML Plugin	Exports a specific page to HTML.

My Page

Your **My Page** is a personal workspace where you can manage your personal information, your preferences, and your assigned activities.

Getting there

After logging into TeamForge, you are taken to your **MY PAGE**.

Contents

After you log in, two main menus - **My Workspace** and **My Page** appear on the top of your **My Page**. Based on your permissions, the menu options are available. These two menu headers change dynamically depending on the menu option you select.

From the **My Workspace** menu, you can access the following:

- If you are a site administrator, click **Admin** to go to your site admin menu options.
- View a list of items you had recently edited.
- View the projects you are a member of or view all the projects existing in the TeamForge site.

From the **My Page** menu, you can access the following:

- Click **DASHBOARD** to view your Project Dashboard.

  ☞ **Note:** You can access the Project Dashboard only if you have the appropriate permissions.

- Click **PROJECTS** to create a new project or to manage your project membership.
- Click **MONITORING** to view your monitored items or to manage your monitoring preferences.
- Click **MY SETTINGS** to manage your user account.
- Click **MY PAGE** to return to your **My Page**.

From your **My Page** workspace, you can do the following:

My Recent Projects	A list of your most recently accessed projects. Click a project to go to its project home page.
My Recent History	A list of the items that you have most recently created or edited. Click an item to go to its details page.
My Items	Your My Page workspace provides consolidated views of all of your CollabNet TeamForge items. All items that are assigned to you, were created by you, or are awaiting your approval are displayed, from all projects of which you are a member.

- Click **ITEMS ASSIGNED TO ME** to view all of the open tasks, tracker artifacts, and document reviews currently assigned to you.

- Click **ITEMS CREATED BY ME** to view all of the tasks, tracker artifacts, and document reviews that you have created.
- If you are an administrator on the site, click **ITEMS AWAITING MY APPROVAL** to view requests for task changes, requests to join projects, and requests to leave projects. Click the title of an item to go to the item's approve request page.
- Click **NEWS** to view recent news from all projects of which you are a member.

The Document Details page

Each document in TeamForge has a **Document Details** page where you can find information about the document.

Contents

The **Document Details** page has these components:

Document Details	View details about the document such as the name, description, file type, and size.
Status	View the document status, including whether it is locked for editing.
Created By	Identify and contact the author of each document version.
Versions	View the version history of the document.
Associations	View any items associated with the document.
Version Comment	Explanation of changes from one version of the document to another.
Monitor	Begin or stop monitoring the document.

Filters for automatically sorting emails from your TeamForge site

X-headers are special instructions that that are added to many email messages. Messages that come from your TeamForge site have X-headers describing the purpose of the email, the event that triggered it, and lots of other information.

You can use these X-headers to sort and filter emails sent by TeamForge.

X-header	Example	Description
X-TeamForge-Application	Tracker	Application, if applicable. Values include Documents, Wiki, Source Code, TeamForge, Tasks, Tracker, Discussion, File Releases.
X-TeamForge-Artifact-Monitoring	true	For artifacts, true if the user is monitoring the artifact directly. If value is false, you are monitoring the folder, not the artifact.
X-TeamForge-AssignedTo	admin	User assigned to artifact or task.
X-TeamForge-CommitPath	/path/to/my/branch	For Subversion commit emails: reports the greatest common path of all files in the commit. For example, if you modify the paths `/a/b/file1` and `/a/c/file2`, the value of the `X-TeamForge-CommitPath` header is `/a`. If you modify `/a/b/file1` and `/file2`,

X-header	Example	Description
		the value is / (the root path of the repository).
X-TeamForge-FolderId	tracker1005	If the object is a folder, the object's ID, or the ID of the object's container.
X-TeamForge-Forum	New user forum	For posts, this is the forum title.
X-TeamForge-PlanningFolder	`/path/to/my/folder`	The path to a planning folder.
X-TeamForge-ProjectId	proj1006	Project ID, if applicable.
X-TeamForge-ServerName	collab.net	Hostname portion of TeamForge URL.
X-TeamForge-Status	Open	Status of artifacts, tasks, documents, and document reviews. Values include: - For artifacts: Open, Closed, Pending, or user-defined. - For tasks: Not Started, OK, Complete, Warning, or Alert. - For documents: Draft, Review, or Final. - For document reviews: Open or Closed.
X-TeamForge-Type	Artifact	The type of the object, if applicable. Values include Tracker, Artifact, Document Folder, Document, Document Version, Wiki Page, Wiki Page Version, Task Folder, Task, FRS Package, FRS Release, Discussion Forum, Discussion Topic, SCM Repository, SCM Commit.

Reference information for TeamForge project administrators

Here are some sample files and other resources to help you figure out what is what in this CollabNet product.

The Project Dashboard page

The **Project Dashboard** offers a centralized view into all development projects managed in TeamForge .

Overview

The **Project Dashboard** provides managers with an at-a-glance overview of the status of each of their projects. It provides summary information on the number and status of the tasks and tracker artifacts in each project, and calculates project overrun and underrun statistics.

The **Project Dashboard** also provides overview information such as project start and end dates and project ranking.

You can see the **Project Dashboard** if you have both the View Tracker and View Task permissions for one or more projects. Only those projects for which you have both the View Tracker and View Task permissions appear on your **Project Dashboard**.

Getting there

In the TeamForge navigation bar, click **My page** > **Dashboard**.

You can view the **Project Dashboard** if you have both the View Tracker and View Task permissions for one or more projects. Only those projects for which you have both the View Tracker and View Task permissions are displayed on your **Project Dashboard**.

Contents

For each project, the **Project Dashboard** displays the following information:

- **Project Activity ranking** - The activity of the project in relation to all other TeamForge projects.
- **Start Date** or **End Date** - The start and end date of the project, based on the start and end dates of all project tasks.
- **Task Status** - The status of the project, based on the "rolled-up" status of all project tasks and task folders. You can configure the "roll up" criteria for each project from the project's **Task Manager Settings** page.
- **Status History** - The history of the project's "rolled-up" status color. These figures are calculated in real time, but do not calculate time that the project's status was **Not Started** or **Completed**.

  Click the status bar to go to the project's **Task Summary** page.
- **Task and Tracker Effort** - The project's current overrun or underrun, based on the difference between estimated and actual effort spent on project tasks and tracker artifacts.

  Only completed and closed tasks and tracker artifacts, with values in the estimated and actual effort fields, contribute to the overrun or underrun calculations.
- **Tracker Status** - The number of open tracker artifacts in the project, per priority value. The number of open tracker artifacts is indicated in parentheses.

  Click the status bar to go to the project's **Tracker Summary** page.

User permissions in CEE and CTF 8.2 projects
Each permission a user can have in a CollabNet Enterprise Edition project has an equivalent permission in a CollabNet TeamForge project, with a few exceptions.
TeamForge 8.2 Permissions for Audit log

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Audit Log - View	None	User loses permission	The user loses the ability in absence of "Domain - Administer" permission.	Conditional mapping. This is handled in the mapping for "Domain - Administrator" permissions.

TeamForge 8.2 Permissions for Announcements
The news application in TeamForge does not have its own permissions. Users' ability to view project news depends on their level of access to the project itself. The permissions for the CollabNet Enterprise Edition Announcements tool are mapped to the closest possible project permissions in the TeamForge.

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Project News - Approve	None	User loses the permission	The user will not be able to create the news if the role does not have the "Project Admin" permissions	These tasks are project administrator only tasks in TeamForge. It is not correct to grant the "Project Admin" permissions to map this permission as that will grant much higher privilege. Hence, this permission is dropped.
Project News - Delete	None	User loses the permission	The user will not be able to delete the news if the role does not have the "Project Admin" permissions	These tasks are project administrator only tasks in TeamForge. It is not correct to grant the "Project Admin" permissions to map this permission as that will grant much higher privilege. Hence, this permission is dropped.
Project News - Edit	None	User loses the permission	The user will not be able to edit the news if the role does not have the "Project Admin" permissions	These tasks are project administrator only tasks in TeamForge. It is not correct to grant the "Project Admin" permissions to map this permission as that will grant much higher privilege. Hence, this permission is dropped.
Project News - Suggest	None	Feature is not available in TeamForge		TeamForge 8.2 doesn't have any such permission.
Project News - View	None	CollabNet Enterprise Edition permission is retained		In TeamForge 8.2, if the user has a role, then the "Project - view" permission is automatically granted.

TeamForge 8.2 Permissions for Discussions

Permissions are mapped to TeamForge's Discussions application.

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Discussions - Discussion Add	Forums Admin	User loses permission	The user loses the permission if the role does not have "Discussion Edit" permission and "Discussion - Use Private" in specific cases.	Conditional Mapping. For Global/ Site-wide role, projects having private discussion, and parent projects having child projects with private forum: Grant only if the role has "Discussion - Discussion Edit" and "Discussions - Discussion Use Private" permissions also. If a project-level role and the project/child projects do not have any private forum, do not consider "Use Private".
Discussions - Discussion Delete	Delete/View	CollabNet Enterprise Edition permission is retained		All public forums are mapped. Additionally, all private forums are mapped if the user has "Discussions - Discussion Use Private" permission.
Discussions - Discussion Edit	Forums admin	User loses permission	The user loses the permission if the role does not have "Discussion Add" permission and "Discussion - Use Private" in specific cases.	Conditional Mapping. For Global/ Site-wide role, projects having private discussion, and parent projects having child projects with private forum: Grant only if the role has "Discussion - Discussion Add" and "Discussions - Discussion Use Private" permissions also. If a project-level role and the project/child projects do not have any

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				private forum, do not consider "Use Private".
Discussions - Discussion Trusted	ALTERNATE MEANS	CollabNet Enterprise Edition permission is retained		The users having this permission are made the trusted users of all the forums in the project subject to "Discussion – Use Private" permission.
Discussions - Message Delete		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.
Discussions - Message Edit		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.
Discussions - Message Post	Post/View	CollabNet Enterprise Edition permission is retained.		All public forums are mapped. Additionally, all private forums are mapped if the user has "Discussions - Discussion Use Private" permission.
Discussions - Discussion View	View only	CollabNet Enterprise Edition permission is retained.		All public forums are mapped. Additionally, all private forums are mapped if the user has "Discussions - Discussion Use Private" permission.
Discussions - Discussion Use Private		CollabNet Enterprise Edition permission is retained.		Other permissions are specified at the folder level. If the role has this permission, then permission can be given to all public and all private forums, otherwise, access is limited to only public forums.

TeamForge 8.2 Permissions for Documents and Files

Permissions are mapped to TeamForge 8.2's Documents application.

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Project Document - Approve	Create/View	User loses permission	No suggest/approve feature in TeamForge 8.2.	Conditional Mapping. This permission is granted only if the user has Project Document – Suggest also.
Project Document - Delete	Delete/View	CollabNet Enterprise Edition permission is retained		This permission is applicable to all root document folders.
Project Document - Edit	Edit/View	CollabNet Enterprise Edition permission is retained		This permission is applicable to all root document folders.
Project Document - Edit Locked	Document Admin	User loses permission	The user is likely to lose the rights to edit a locked document in certain cases.	Conditional Mapping. This permission is granted only if the user has "Project Document - Approve", "Project Document - Edit", "Project Document - Reserve" and "Project Document – Suggest" permissions also.
Project Document - Reserve		User loses permission	The user will lose the ability to reserve documents in absence of "Project Document - Edit" permission	
Project Document - Reserve for Others		User loses permission	Users without "Project Document – Edit" permission will lose the rights	
Project Document - Reserve Mine		User loses permission	Users without "Project Document – Edit" permission will lose the rights	
Project Document - Suggest		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.
Project Document - View	View Only	CollabNet Enterprise Edition permission is retained		This permission is applicable to all root document folders.

TeamForge 8.2 Permissions for Issue Tracker/Project Tracker

CollabNet Enterprise Edition uses the same set of permissions for Issue Tracker and Project Tracker. There are some permissions for individual tools too. The CollabNet Enterprise Edition project tracker permissions are appropriately

transferred to Project Tracker (integrated application in TeamForge 8.2) or Issue Tracker, based on the tool that the project uses or the inherited tool permissions in child projects or the permissions that are a part of a global or site-wide role.

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Project Issue Tracking - Assignable	Pluggable: Assignable	CollabNet Enterprise Edition permission is retained		This permission is assigned to the user for Project Tracker if the project uses Project Tracker (or) if permissions are inherited to child projects using Project Tracker (or) if the role is a global/site-wide role.
Project Issue Tracking - Assignable	TeamForge 8.2: conditionally dropped	User loses permission	The user loses permissions in the case of global/site-wide role, if the role does not have "Edit/ View" permission	User becomes assignable through the "Edit/View" permission. There is no separate permission in TeamForge 8.2.
Project Issue Tracking - Add Comment	TeamForge 8.2: dropped	User loses permission	The user loses rights if the role does not have "Attach File", "Change" and "Change Status" permissions.	
Project Issue Tracking - Attach File	TeamForge 8.2: dropped	User loses permission	The user loses rights if the user does not have "Change", "Add Comment" and "Change Status" permissions	
Project Issue Tracking - Change	Pluggable: Edit Artifact	CollabNet Enterprise Edition permission is retained		This permission is assigned to the user for Project Tracker if the project uses Project Tracker (or) if permissions are inherited to child projects using Project Tracker (or) if the role is a global/site-wide role.
Project Issue Tracking - Change	TeamForge 8.2: Edit/ View	CollabNet Enterprise Edition permission is retained	Granted only if the role also has "Attach File", "Add	Conditional mapping.

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
			Comment" and "Change Status" permissions.	
Project Issue Tracking - Change Status	TeamForge 8.2: dropped	User loses permissions	The user loses rights if the user does not have "Change", "Add Comment", and "Attach File" permissions	
Project Issue Tracking - Configure	Pluggable: Configure project	CollabNet Enterprise Edition permission is retained		This permission is assigned to the user for Project Tracker if the project uses Project Tracker (or) if permissions are inherited to child projects using Project Tracker (or) if the role is a global/site-wide role.
Project Issue Tracking - Configure	TeamForge 8.2: Tracker Admin	User loses permission	The user can lose the rights if the role does not have "Submit", "Change" and "Add Comment" permissions. Also, the user may gain "Attach", "Change status" permissions	Conditional mapping. Granted only if the role also has "Submit", "Change" and "Add Comment" permissions.
Project Issue Tracking - Move Issue	Pluggable: Move artifact	CollabNet Enterprise Edition permission is retained		This permission is assigned to the user for Project Tracker if the project uses Project Tracker (or) if permissions are inherited to child projects using Project Tracker (or) if the role is a global/site-wide role.
Project Issue Tracking - Move Issue	TeamForge 8.2: Delete/View	Permissions are gained	Granted only if the user gets "Submit/ View" and "Edit/ View" permissions.	Conditional mapping. User may get additional permissions to delete artifacts and also the additional permissions gained via the "Submit/ View" and "Edit/

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				View" permissions. Drop if Project role.
Project Issue Tracking - Query	Pluggable: Query	CollabNet Enterprise Edition permission is retained		This permission is assigned to the user if the role is a project role (or)if the project uses Project Tracker (or) if permissions are inherited to child projects using Project Tracker (or) if the role is a global/site-wide role.
Project Issue Tracking - Query	TeamForge 8.2: View Only	CollabNet Enterprise Edition permission is retained		There is no exact parity, users having "View" permission will be able to use Filter and Search. Also, Issue Tracker uses this permission for "View" access.
Project Issue Tracking - Submit	Pluggabe: Enter artifact	CollabNet Enterprise Edition permission is retained		This permission is assigned to the user if the role is a project role (or)if the project uses Project Tracker (or) if permissions are inherited to child projects using Project Tracker (or) if the role is a global/site-wide role.
Project Issue Tracking - Submit	TeamForge 8.2: Submit/View	CollabNet Enterprise Edition permission is retained		
Project Issue Tracking - View	Pluggable: View artifact	CollabNet Enterprise Edition permission is retained		This permission is assigned to the user if the role is a project role (or)if the project uses Project Tracker (or) if permissions are inherited to child projects using Project Tracker (or) if the role is a global/site-wide role.
Project Issue Tracking - View	TeamForge 8.2: dropped	CollabNet Enterprise Edition permission is retained		This permission is redundant, Issue Tracker uses "Query"

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				permission to determine view access.
Domain Issue Tracking - Configure	Pluggable: Configure site	CollabNet Enterprise Edition permission is retained		Conditional mapping. This permission has an impact only if it is part of a site-wide role, otherwise it can be dropped.
Domain Issue Tracking - Configure	TeamForge 8.2: dropped	This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.

TeamForge 8.2 Permissions for Project Dashboard and Tasks

Permissions are mapped to TeamForge 8.2's Tasks application. As there are several gaps between CollabNet Enterprise Edition and TeamForge 8.2's Task tracking application, it is difficult to come up with an exact role mapping. The role played by the user having the permission is used as the baseline for the mapping.

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Project Dashboard - View		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.
Project Dashboard - View Manager Reports		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.
Tasks - Configure	Task Admin	CollabNet Enterprise Edition permission is retained		There are differences between CollabNet Enterprise Edition and TeamForge 8.2 features, so there isn't any exact comparison.
Tasks - Edit Others	Edit/View	CollabNet Enterprise Edition permission is retained		
Tasks - Edit Self		User loses permission	The user loses rights to edit other's tasks assigned to self.	In TeamForge 8.2 users with "Edit" permission can also edit other tasks in the specified folder(s).
Tasks - Remove Plan	Delete/View	CollabNet Enterprise Edition permission is retained		

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Tasks - View	View only	User gains permission	The user gains the ability to view tasks assigned to others	
Tasks - View Others	View only	CollabNet Enterprise Edition permission is retained		"View Only" in TeamForge 8.2 means the user can view any task in the specified folder(s).

TeamForge 8.2 Permissions for Projects

TeamForge 8.2 does not have fine level controls with regard to project functions as a project-level role. However, project operations are a part of a site-wide role. Hence the mapping is done to site-wide roles, except for project administrator.

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Project - Associate Project		User loses permission	The user loses the ability to map project to a category/project group	TeamForge 8.2 allows only project administrators to do so, which is way above this permission.
Project - Associate With Project		User loses permission	The user loses the ability to map project to a category/project group	
Project - Delete	Projects: Delete/ View, Categories: Delete/View	User loses permission	The user loses rights to delete project if the role is project-level/ Global	Conditional mapping. This permission is granted only if the role is a site-wide role.
Project - Designate Subproject		User loses permission	The user loses the ability to set parent-child relationships between projects	TeamForge 8.2 allows project administrators to set parent-child relationships.
Project - Designate Superproject		User loses permission	The user loses the ability to set parent-child relationship between projects	TeamForge 8.2 allows project administrators to set parent-child relationships.
Project - Edit	Project Admin: Project Admin Permissions, Project	User gains permission	The user gains other project administrator capabilities like	Mapping is done to PCE also since it is a project administrator's

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
	Pages: Project Pages Admin		setting parent-child relationship etc.	responsibility to configure PCE pages.
Project - Invite		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.
Project - Lock		The user loses the ability to lock/unlock projects	Project administrators can do this in TeamForge 8.2.	
Project - View	Projects: View Only	CollabNet Enterprise Edition permission is retained		Conditional mapping. The proposed mapping is done when the role is a domain role. For project/global project roles, user will have the view access just by the role assignment. In the case of a role project/global project role with just this permission, the users belonging to the role are added as a member of the project in case of private projects.
Project Content - View		User gains permission	Without having such a permission, users will gain access to remote published contents.	TeamForge 8.2's remote publishing implementation needs only "View" access to the project.
Project Member - List		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.
Project Page - View	Projects: View Only	User gains permission	Exact changes not clear	Conditional mapping. The same approach as that of "Project – View" permission is followed. There is no clear separation of responsibilities between "Project – View" and "Project Pages – View". In TeamForge 8.2, these are merged.

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Private Project Document - View		CollabNet Enterprise Edition permission is retained		
Project - Site-specific		User gains permission	The user will lose the ability to control access to site specific contents. Users will see all project content if the user gets permissions to the project.	TeamForge 8.2 does not have any such feature.

TeamForge 8.2 Permissions for Role

TeamForge 8.2 does not have fine level controls with regard to roles functions at project-level. However, role operations are a part of a site-wide role. Hence the mapping is done only in the case of domain roles. For project level roles, project administrator has these capabilities.

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Role - Add	Roles: Create/Edit/ View	User loses permission	The user loses the ability if it is a project-level role and the user does not get project administrator permissions.	Conditional mapping. The permission is granted if the role is a domain role and the role also has "Role – Grant" and "Role – Edit" permissions.
Role - Delete	Roles: Delete/View	User loses permission	The user loses the ability if it is a project-level role and the user does not get project administrator permissions.	Conditional mapping. The permission is granted if the role is a domain role.
Role - Edit	Roles: Create/Edit/ View	User loses permission	The user loses the ability if it is a project-level role and the user does not get project administrator permissions.	Conditional mapping. The permission is granted if the role is a domain role and the role also has "Role – Add" and "Role – Grant" permissions.
Role - Grant	Roles: Create/Edit/ View	User loses permission	The user loses the ability if it is a project-level role and the user does not get project administrator permissions.	Conditional mapping. The permission is granted if the role is a domain role and the role also has "Role – Add" and "Role – Edit" permissions.

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Role - Request	Alternate Means	CollabNet Enterprise Edition permission is retained		The user may get additional permissions if the user does not have "Project – View" permission.
Unapproved Project Role - Grant		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.

TeamForge 8.2 Permissions for Resource Patterns

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Resource - Add		User loses permission	The user will lose the ability unless it is a project role and the user gets project administrator permissions.	Path-based permissions (PBP) is the equivalent and is configurable only by project administrators. No such feature at domain level.
Resource - Delete		User loses permission	The user will lose the ability unless it is a project role and the user gets project administrator permissions.	Path-based permissions (PBP) is the equivalent and is configurable only by project administrators. No such feature at domain level.
Resource - Edit		User loses permission	The user will lose the ability unless it is a project role and the user gets project administrator permissions.	Path-based permission (PBP) is the equivalent and is configurable only by project administrators. No such feature at domain level.

👉 **Tip:** Some examples given below may give you an idea of how PBP can be used to replace the regular patterns that were being used.

Example of regex that can be transformed completely using PBP:	/tags($|/.*) - This pattern implies anything that starts with /tags/ or exact '/tags'.
	As the regex is like a super path pattern, here a PBP like [/tags] can suffice.

Example of regex that can be transformed only partially:	/branches($\|/$) - This pattern implies only '/branches' and it does not match '/branches/xyz'.
	This can not be achieved using PBP as [/branches] would match everything under '/branches'. However, as a complex approach, you may need to grant access to [/branches] and disable all its child path access with explicit path components.

☞ **Note:** It is recommended that you try this on stage before doing it in production.

TeamForge 8.2 Permissions for Search

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Search - Subversion Revision Properties		User gains permissions		TeamForge 8.2 doesn't have any such permission and it is controlled by version control permissions.

TeamForge 8.2 Permissions for Version Control
Permissions are mapped to TeamForge 8.2's Source Code application. Resource pattern association with the permissions are not handled and will appear in the exception report. The "Change in Behavior" column ignores the regex aspect.

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
VersionControl - Add	Commit/View	User gains permission		The user will get permission to modify existing files unless the role also has VersionControl – Modify.
VersionControl - Copy	Commit/View	User loses permission		Conditional mapping. This is mapped only if the user also has the "VersionControl – Add" permission.
VersionControl - Delete	Delete/View	CollabNet Enterprise Edition permission is retained		
VersionControl - Modify	Commit/View	User gains permission		The user will get permission to add new files unless the role also has the "VersionControl – Add" permission.

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
VersionControl - Read	View only	CollabNet Enterprise Edition permission is retained		
VersionControl CC - Admin				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC - Change Event				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC - Change Type				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC - Checkout				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC - Protect				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC - Reserve				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC - Unreserve				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC Attribute - Create				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC Branch - Create				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC BranchType - Create				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC Hyperlink - Create				This Clear Case permission

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				is dropped in TeamForge 8.2.
VersionControl CC Label - Create				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CC Trigger - Create				This Clear Case permission is dropped in TeamForge 8.2.
VersionControl CVS - Admin	Source Code Admin	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS - Edit	Commit/View	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-one mapping between CollabNet Enterprise Edition actions and

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS - Module	View only	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS - Release	View only	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-one mapping between CollabNet Enterprise

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS - Tag	Commit/View	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS - Watch	Commit/View	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-one mapping between

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS Module - Add	Commit/View	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS Module - Delete	Commit/View	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS Module - Edit	Commit/View	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS Module - View	View only	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				there is no one-to-one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS Repository - Delete	Delete/View	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/ site-wide role. As there is no one-to-one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS Repository - Rename	Source Code Admin	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
				site-wide role. As there is no one-to-one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.
VersionControl CVS Repository - ToggleExecutable	Commit/View	User gains permission		This mapping is done only if project role and the project uses CVS (or) if the permissions are inherited to child projects using CVS (or) the permission is part of a global/site-wide role. As there is no one-to-one mapping between CollabNet Enterprise Edition actions and TeamForge 8.2 clusters, all read-only operations are mapped to "View Only" permission and write operations are mapped to the "Commit/View" permissions. The user may get additional permissions in some cases.

TeamForge 8.2 Permissions for Wiki

Wiki will come in as an integrated application post TeamForge 8.2. At that time, these permissions will be transferred to the integrated application. As of now, the permissions are mapped to TeamForge 8.2 wiki permissions and hence users will get the same permission in TeamForge 8.2's wiki.

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Project Wiki - Administer	Wiki Admin, Delete/ View	CollabNet Enterprise Edition permission is retained		CollabNet Enterprise Edition wiki allows replacing CollabNet Enterprise Edition actions with ACL in wiki pages. This mapping ignores that setting since TeamForge 8.2 does not have that feature.
Project Wiki - Contribute	Create/Edit/View	CollabNet Enterprise Edition permission is retained		CollabNet Enterprise Edition wiki allows replacing CollabNet Enterprise Edition actions with ACL in wiki pages. This mapping ignores that setting since TeamForge 8.2 does not have that feature.
Project Wiki - View	View only	CollabNet Enterprise Edition permission is retained		CollabNet Enterprise Edition wiki allows replacing CollabNet Enterprise Edition actions with ACL in wiki pages. This mapping ignores that setting since TeamForge 8.2 does not have that feature.

Reference information about CollabNet TeamForge site administration

Here is some detailed reference information to help you figure out what is what in this CollabNet product.

Replica status icons

The icons represent the health of a replica server, the state of a replicated repository, or the status of a replication command.

You can get more information related to a specific status by clicking on its icon. However, not all icons are links.

Replica server

Icon	Description
☀	There are no errors. The replica server has polled the master within 3 times the **Command polling interval**. For example, if the polling frequency is 10 seconds, TeamForge has a record of contact within the last 30 seconds.

Icon	Description
	This could mean one of two things: • The server has an error in the last 30 commands, although the most recent command is successful. • The server has not polled the master within 3 times the **Command polling interval**, although it has within 30 times the polling interval.
	This could mean one of two things: • The most recent command is a failure. • The server has not polled the master within 30 times the **Command polling interval**.

Replicated repository

Icon	Description
	Replicated repository is in sync with the repository on the master.
	Replicated repository is being added.
	Replicated repository is being deleted.
	Replicated repository is being synchronized using `svnsync`.
	Synchronization events are queued on the TeamForge server and waiting for the replica to process them.

Replication command

Icon	Description
	The command is queued in the TeamForge server for the replica.
	Depending on the replica server queue, the command is being processed or scheduled to be processed.
	The execution of the command failed in the replica server. You will see a link to the *server log* that contains details about the error.
	The execution of the command succeeded in the replica server.

Migrating tools from Enterprise Edition to TeamForge 8.2
As you move onto CTF 8.2, you may want to know how the CollabNet Enterprise Edition (CEE) project information is brought over to CollabNet TeamForge 8.2 (CTF 8.2).
Migrating tools from CollabNet Enterprise Edition to CollabNet TeamForge 8.2
As you move onto CTF 8.2, you may want to know how the CollabNet Enterprise Edition (CEE) project information is brought over to CollabNet TeamForge 8.2 (CTF 8.2).

CEE and CTF 8.2 Tool Mapping Details

Most of the tools and their functionality are similar between CEE and CTF 8.2. However, within each tool, some work-arounds might have been done to bring over the CEE project information and resources.

This table shows how tools in CEE map to CTF 8.2 tools:

CEE Tool	CTF 8.2 Tool
Announcements	Project News
Discussions	Discussions
Documents & Files	Documents
Wiki	<Data is not migrated>
CVS	CVS
Subversion	Subversion
Project Dashboard/Tasks	Tasks
Issue Tracker	Tracker
Project Tracker	Project Tracker (Integrated Application)
Project metrics	<Equivalent tool is not available.>
Project Pages	Project Pages
Remote Publishing	Remote Publishing

This table shows how objects in CEE map to CTF 8.2 objects:

CEE Objects	CTF 8.2 Objects
Project	Project
Project Group	Project
Category	Category + Project

How to find what you are looking for?

There are several tools that are migrated from CollabNet Enterprise Edition to CollabNet TeamForge 8.2.

You can look through the list for the required tool details. Within each tool, the information is divided into two sections - **Migration Overview** and **Notes**. Migration Overview section provides the mapping information whereas the Notes section lists the work-arounds or dissimilarities, if any.

Overview

- Post migration, CEE's audit log entry table is available in CTF 8.2's data folder as "cee_backedup_audit_log" table.
- Post migration, project tracker permissions are not migrated to CTF 8.2's tracker and vice versa. Project Tracker permissions are transferred to Project Tracker which is brought in to CTF 8.2 as an integrated application.

   ☞ **Note:** Project hierarchy and flow of permissions to child projects is a known issue as of now. Suppose before migration, the parent project is Issue Tracker (CEE) based, child project is Project Tracker (PT) based and some roles from the parent projects are in use in the child project. In all probability, the role inheritance will not be migrated.

- The look project contains a single consolidated exception report - `Documents>Root Folder>cee_migration_exception_report> GLOBAL-REPORT-CONSOLIDATED.txt.`

- The `Documents>Root Folder>cee_migration_exception_report><project name>REPORT.txt` folder holds all per project exception reports.
- The CEE files that were committed in a CVS repository, display junk characters when directly viewed in CTF 8.2. It is recommended to download the file and view it.
- The "www" content from CEE's www project is moved to the "html" directory in the branding repository of "look" project in CTF 8.2.

The assumptions made while migrating CEE to CTF 8.2 are given here:

List of the Tools Migrated from CEE to CTF 8.2
Migration Assumptions

As you move onto CTF 8.2, you may want to know the assumptions that were made while migrating CollabNet Enterprise Edition (CEE) project information to CollabNet TeamForge 8.2 (CTF 8.2).

- Global Data:

  - Default usergroups '*Domain Users' and '**Host Users' are not migrated.
  - System users, if part of any user group, are dropped.
  - System users like guest, webmaster, postmaster and mailer-domain are not migrated and their references are mapped to 'Nobody' user.
  - Host Superuser (root) is not migrated in favor of administrator account in CTF 8.2 and its references are mapped to the administrator user.
  - Users who are associated with Host Admin role are migrated as Site administrators.
  - Domain roles; 'Anonymous Guest' and 'Registered User' are dropped and their permissions are transferred to each public project's Default Access Permissions user class.
  - Host roles are not migrated.
  - Project announcements of www project are moved to the Look project as news.
  - CEE's HAC_AUDIT_LOG_ENTRY table is moved to cee_backedup_audit_log.
- Projects Data:

  - Unapproved and deleted projects are not migrated.
  - Default projects like testproject, www, project-templates are not migrated.
  - Categories and project groups are migrated as normal projects.
  - License detail of projects is not migrated.
  - CVS repositories are accessible with a new URL only.
  - CVS working copies must be recreated.
  - SVN working copies continue to work as is.
  - The Alias URL of the project is not migrated.
  - When the project creation date is not found for a project in the audit log, the default fallback value (2000-01-01) is used.
  - Project owners message is moved to the news component as a news item.
- Discussion services tool:

  - Soft deleted discussions are not migrated.
  - Trusted discussions are migrated as moderated discussion and trusted subscribers are moved to the trusted users list.
  - The following discussion settings are dropped as CTF 8.2 does not support these settings:

    - Web UI View
    - Web Posting
    - Web Post CAPTCHA
    - Display Email Address
    - Remove Deleted Messages
    - Remove Unmoderated Messages
    - Content in Email

- Message Size
- Attachments
- Advanced Email Filters
- The following discussion topic value is dropped as CTF 8.2 does not support it:
  - Message headers
- In CTF 8.2, discussion email address must begin with an alphabet and it can not contain special characters except an underscore. So, if a CEE's discussion address contains hypens or periods, they are replaced with underscores. Also, if it begins with a number, it is replaced with the equivalent number in word followed by an underscore.
- Lengthy discussion topic titles, if any, are truncated to 250 characters.
- Soft deleted approved and unapproved topics including replies are not migrated.

- Documents and files tool:
  - Unapproved folders and documents are not migrated.
  - In CTF 8.2, a document's status can be either draft or final. So, CEE's documents in 'Stable' status are migrated as final and documents in any other status are migrated as draft.

- Announcements tool:
  - Announcements are migrated to the news component in CTF 8.2.
  - CEE features like Image url, publishing and event dates are not supported in CTF 8.2. Those details are appended as part of announcements itself.

- Project pages tool:
  - The documents existing in CEE's "Document" component are moved to a sub-folder in the CTF 8.2's "Documents" tool's "cee-migrated-docs" folder.
  - The page components, HTML and Activity table components are migrated as text/html component.
  - The page components Metrics report, Milestones table and Project overview are not migrated.
  - The availability property of a page 'All users' and 'Users with project-edit' is mapped to 'Visible'; and 'None' will be mapped to 'Hidden' in CTF 8.2.
  - Backward compatibility of free-form page references in html components is not supported.
  - Sub-page icons are not migrated.

- Issue Tracker tool:
  - Issue tracker in CEE is migrated to a tracker called 'Issue tracker' in CTF 8.2.
  - Issues are migrated as artifacts.
  - The email notification preferences are not migrated.
  - The associations between Component and Subcomponent, Component and Version and, Component and Target Milestone are dropped as CTF 8.2 does not support these associations.
  - Issue tracking configuration parameters not migrated.
  - Internal users of Issue tracker are dropped.
  - Dependencies are migrated as associations.

Mapping functionality between Enterprise Edition and TeamForge 8.2

Most of the tools and their functionality are similar between CEE and CTF 8.2. However, within each tool, some work-arounds might have been done to bring over the CEE project information and resources.

There are several tools that are migrated from CollabNet Enterprise Edition to CollabNet TeamForge 8.2.

You can look through this section for the required tool details and work-arounds or dissimilarities, if any.

This table shows how the tools in CEE map to CTF 8.2 tools:

CEE Tool	CTF 8.2 Tool
Announcements	Project News
Discussions	Discussions

CEE Tool	CTF 8.2 Tool
Documents & Files	Documents
Wiki	\<Data is migrated post-migration, as required\>
CVS	CVS
Subversion	Subversion
Project Dashboard/Tasks	Tasks
Issue Tracker	Tracker
Project Tracker	Project Tracker (Integrated Application)
Project metrics	\<Equivalent tool is not available.\>
Project Pages	Project Pages
Remote Publishing	Remote Publishing

This table shows how the objects in CEE map to CTF 8.2 objects:

CEE Objects	CTF 8.2 Objects
Project	Project
Project Group	Project
Category	Category + Project

Migrating projects and categories

Here are the migration details and migration notes for projects and categories.

Migration Overview

- The public and private projects are retained appropriately.
- The project members in CTF 8.2 consist of users assigned a role in the current project.

  ☞ **Note:** In CEE, a project member must have a role whereas, in CTF 8.2, a user can be a project member without being assigned a role.

- Locked projects in CEE are migrated as locked projects to CTF 8.2.

  ☞ **Note:** The locked projects seem to allow certain actions even after being locked. For example, forums can still be monitored by administrators.

- CEE projects that are imported into CTF 8.2 can be selectively deleted, if required.
- CEE's Project groups project is migrated as a regular project in CTF 8.2. Members of the Project groups project are migrated as it's child projects in CTF 8.2.
- In CEE, categories could behave as projects, having tools like tracker, documents, discussion and so on. However in CTF 8.2, categories are plain containers which have projects in them, enabling any project to be placed in a category. CTF 8.2 categories do not have project tools like tracker, documents, discussions etc.

  ☞ **Note:** Categories from CEE to CTF 8.2 are migrated in two steps:

    - A CTF category (with the same CEE category name) is created
    - The tool data (eg. artifacts, documents, discussions etc) are migrated into a CTF 8.2 project with the CEE category name.

  .

Consider this example. Assuming there exists a CEE category (named - "cat") having projects (proj1, proj2) with tool data of its own (document folders: folder1, folder2). When this category is migrated to CTF 8.2, a CTF 8.2

category called "cat" is created, and the projects (proj1 and proj2) are placed in this category. Also, a CTF 8.2 project "cat" is created which holds the tool data that the category had in CEE (document folders folder1, folder2).

- The project-subproject, category-subcategory association is maintained.

This table shows how Project fields in CEE map to Project fields in CTF 8.2:

CEE Projects	CTF 8.2 Projects
Project Name	Project Name
Summary	Description
Parent Project	Parent Project
Project Categories	None
Public Project	None
Project Language	None
Versioning Component	None
Tracking Component	None
Artifact Code	None
Project Owner	None
Project Proposer	Created by user name
Date Last Modified	Date Created/Date Last Modified
Project Approver	Last modified by (user)
Owner's Message	None
CVS Projects	None
SVN Projects	None

This table shows how Category fields in CEE map to Category fields in CTF 8.2:

CEE Categories	CTF 8.2 Categories
Category Name	Category Name
Parent Category	None
Public Category	None
Use Category Index.html	None
Category Language	None
Initial Projects	None
Use Category Home Page	None

Notes

- In CTF 8.2, unapproved projects (requested project) are just project-creation requests, with the project name and description. However, in CEE, unapproved projects can be actively used by the project creators. As the unapproved projects are conceptually different between CEE and CTF 8.2, these projects are not migrated.
- The IsDeleted field in CTF 8.2 corresponds to whether the project is deleted or not in CEE.
- If the project creation date is not found in the audit log, the fall-back value specified in the `/opt/collabnet/cee_dump_load/conf/c6migrate.conf` file is used.

- The Date Created field in CEE does not have an equivalent field in CTF 8.2.
- There is no default project language in CTF 8.2.
- The monitoring feature in CTF 8.2 supports desired user notification.
- In CTF 8.2 a project is created "Private" by default.
- In CTF 8.2, the "Artifact Code" is constant across the site. It is not configurable project-wise.
- In CTF 8.2, the project creator becomes default project owner.
- CEE categories have project tools while the CTF 8.2 categories do not have tools.
- The name of the category can be edited in CTF 8.2.
- Only the site administrator can create project category in CTF 8.2.

Migrating roles and permissions

Here are the migration details and migration notes for roles and permissions data.

Migration Overview

- Project roles are migrated appropriately.
- CEE's equivalent of Project roles with global/all projects visibility are global project roles in CTF 8.2 and these roles are migrated appropriately.
- CEE's equivalent of Domain Roles are Site-wide roles in CTF 8.2 and these are migrated appropriately.
- There is no equivalent of CEE's Host roles, users belonging to "Host Admin" role are migrated as Site Administrators in CTF 8.2. Users belonging to other Host roles, if any, will lose such roles after migration and the details are exception reported.
- CEE's Host SuperUser (root) is mapped to CTF 8.2's TeamForge Administrator (admin).
- For details of the mapped permissions, see *User permissions on a CEE and CTF 8.2 site* on page 362 or *User permissions in CEE and CTF 8.2 projects* on page 320
- User and group associations with the roles are retained.
- Role associations with internal user groups - Host Users and Domain Users, are not migrated since the feature is not available in CTF. These mappings will appear in the exception report.
- Project role requests from CEE projects are migrated as same to CTF 8.2.
- CTF 8.2 administrators can configure permission mapping using the configuration file. The `permissionmapping.conf` configuration file is available here: `/opt/collabnet/cee_dump_load/conf/`.

This table shows how fields in CEE map to fields in CTF 8.2:

CEE	CTF 8.2
Host administrators	Site-administrators
Domain roles	Site-wide roles
Project roles	Project roles
Project roles with All projects visibility	Global project roles
Name	Role Name
Description	Description
Requestable	Requestable
Ownership Role	None
Grant role on subproject creation	None
Block recursion into private projects	Prevent Access (Site-wide roles) or Prevent Inheritance (Project/Global roles)

Notes

- CEE's Host roles are not migrated, multi-tenancy is implemented differently in CTF 8.2.
- Domain Roles - Registered User role is dropped in favor of "Default Access Permissions" at individual project level. In CTF 8.2, there is no global control like a domain role.
- Domain Roles - Anonymous Guest role is dropped in favor of "Default Access Permissions" at individual project level. In CTF 8.2, there is no global control like a domain role.
- The Ownership role and Grant role on sub-project creation functionalities are dropped.
- If the role creation/modification date is not found in the audit log, either the project creation/modification or the fall-back date is used. The default fall-back value is "2000-01-01 00:00:00", as specified in the `/opt/collabnet/cee_dump_load/conf/c6migrate.conf` file.

Migrating SCM integrations

Here are the migration details and migration notes for Source Code Manager integrations.

Migration Overview

- Two SCM integrations can be created during migration. One for Subversion and another for CVS. CVS creation is optional.
- The version control's view page in CTF 8.2 shows extra information in migrated folders in the **Age**, **Author** and **Last Log Entry** columns.

Notes

- In the migrated repository (in CTF 8.2), the **Commits this week** column displays "0", even if there had been commits in CEE during that week. This happens because the historical commit details are not captured for migration.
- In CEE, there can be only one version control component for a project whereas in CTF 8.2, there can be multiple version control components associated to a project.
- Post-migration, new commits are not archived in commit discussions. However, during migration, the existing archived commits are migrated.
- Global search does not work for source code. The ability to search source code is not supported as the revisions are not converted to commit objects during migration.

Migrating users and user groups

Here are the migration details and migration notes for users and user groups.

Migration Overview

- CEE's Host Superuser "root - root@domainname" is migrated as CTF 8.2's TeamForge administrator "admin – root@collab.net.
- CEE's Host administrators are migrated as CTF 8.2's site administrators.
- User group mapping and project-role association is retained.

This table shows how fields in CEE map to fields in CTF 8.2:

CEE	CTF 8.2
Host Superuser	Administrator
Host Administrator	Site Administrator
User - User Group association	User - User Group association
User Group - Project - Role association	User Group - Project - Role association
Group Name	Full Name
Short Description	Description

CEE	CTF 8.2
Initial users	None
Allow new user to group	Add
User Location	None
Username	User Name
Email	Email Address
Full Name	Full Name
Organization	Organization
Password	Password
Confirm Password	Confirm Password
Email Language	None
Creation Options	Send Welcome Message?
None	Restricted User
None	Site Admin
Secondary Email	Alternate emails
Has seen click-through	None
Has seen welcome page?	None
Include user in subversion reporting	None
Project Tracker Self notification	None
Project Tracker maximum report time	None
Group memberships	None
Times logged in	None
Domain-wide role	
Host-wide roles	Site Admin
None	Profile Picture
None	Status
None	Date Time Format
None	Detail

Notes

- Post migration, CEE users must raise a "Forgot password" request to login into CTF 8.2. Using the response e-mail, the users can reset their passwords.
- CEE's default groups – "**Host Users" and "*Domain Users" are not migrated to CTF 8.2. The default user group associations (role mapping) will not be retained post-migration.
- The following are the default users in CEE: "guest", "webmaster", "postmaster", "mailer-deamon" and "root". Post-migration, "root" is mapped to "admin" and other default users are mapped to "nobody" in CTF 8.2.
- Unix system user accounts are not migrated from CEE to CTF 8.2. They are either dropped or renamed.
- Issue Tracker's internal users (for example, issues@ert) are mapped to "Nobody".

☞ **Note:** Users of the form `issues@projectname` are not migrated. All assignments or concepts to such users will be set to "nobody/none" after migration.

- Special Users of CEE are not migrated to CTF 8.2.
- CEE allows users to add up to four secondary email addresses to their accounts. In CTF the supported alternate (secondary) email addresses are three. The first three secondary emails are migrated to CTF; the fourth email address is not migrated.
- Email language feature is not available in CTF 8.2.
- In CTF 8.2. notification setting is same for all the tools.

Migrating wiki

Here are the migration details and migration notes for wiki pages.

Migration Overview

- Only the latest revision of wiki pages is migrated from CEE to CTF 8.2.
- Wiki migration tool compares each of the wiki pages taken for migration with a static set of hard-coded page names given below. If the page names are found to be same, the wiki pages are not migrated. The list of reserved wiki pages is here: *Reserved Wiki pages*
- Source wiki page "FrontPage" is migrated as "HomePage" wiki page in CTF 8.2.
- Source wiki page "HomePage" is migrated as "HomePageMig" wiki page in CTF 8.2.
- If the source wiki page contains a URL without enclosing square brackets, it is migrated as text (a broken link).

Notes

- Wiki pages which do not have CamelCase page names are not migrated.
- If an attachment file is missing at the source, then it is migrated as a broken link.
- Subpages links specified in source wiki page are not retained post-migration.
- Macros are not supported in CTF 8.2, hence macros are not migrated.

Migrating project pages

Here are the migration details and migration notes for project pages.

Migration Overview

- For PCE-enabled subversion projects, CTF 8.2 Home Page is replaced with CEE Home Page.
- For CVS and non-PCE enabled subversion projects, CTF 8.2 Home Page is replaced with remote publishing content (index.html).
- Top level pages and sub-pages are retained.
- The HTML, Activity table, Subprojects, freeform pages and documents components of CEE are migrated to CTF 8.2.

    ☞ **Note:** The documents PCE content in CEE is migrated to the Documents tools in CTF 8.2 and the functionality to display the same via CTF 8.2 Docs PCE is provided. Post migration, the documents relevant to PCE are stored under the `Documents from CEE Project Pages` folder. Appropriate permissions on the Documents tool in CTF 8.2 are required.

- CEE's HTML and Activity table components are migrated to the Text component in CTF 8.2.

This table shows how fields in CEE map to fields in CTF 8.2:

CEE field	CTF 8.2 field
Home Page	CEE Home Page - for PCE enabled Subversion projects
Home Page	Index.html - for CVS/Non-PCE enabled Subversion projects
Top level - Sub pages associations	Top level - Sub pages associations

CEE field	CTF 8.2 field
Html, Activity table, Subprojects, Free form pages, Document components	Text, Subprojects, Free form pages, Document components

Notes

- The free-form page links mentioned in the sub-pages do not work in CTF 8.2 as URL compatibility doesn't work for free-form links referred in HTML components. The CTF 8.2 site/project admin would need to point the links to correct pages.
- The icons added in sub-page are not migrated to CTF 8.2.
- The CEE components that are not migrated to CTF 8.2 include: Metrics report, milestones and project overview.
- The project home pages that displayed sub-projects in CEE will not show the sub-projects list in CTF 8.2 until the PCE is enabled.
- In CTF 8.2, you can enable the PCE for CVS projects, if desired.
- The project page content is not version controlled in CTF 8.2. During migration, CEE's latest version is moved over to CTF 8.2. The old information (older revisions) can be accessed through the current project's subversion repository from `trunk/www/cn-project-pages/`.

Migrating Issue Tracker

Here are the migration details and migration notes for Issue Tracker.

Migration Overview

- Issue Tracker in CEE is migrated to a tracker called "Issue tracker" in CTF.
- Each CEE issue becomes an artifact in the "Issue tracker" tracker.
- Each CEE issue can be identified with its corresponding artifact, as the **Issue Id** appears as a prefix in the corresponding artifact's **title**. For example, after migration the title of an artifact might be like - "[issue 1] - Issue Title", where "issue 1" would be the original artifact Id in CEE.
- The association's comment is of the form: "[issueX] depends on [issueY]", where X and Y are the source and target issue ids.

  ☞ **Note:** The date and time stamps are retained in all instances. Only the last date and time stamps are displayed.

- All the issue activities except dependency activities in CEE are displayed under the **Change Log** tab in CTF. The issue Id of dependent and/or blocker issues are migrated as "Associations".

  ☞ **Note:** CTF does not have corresponding fields for "dependent" or "blocker" issues. As the change log tab displays only the changes that are made to the tracker fields, dependency and blocker changes are not displayed there.

- The comments and status changes are displayed under the **Status/Comments** tab.
- All the attachments in an issue are made available under the **Attachments** tab of the respective artifact.
- The **data**, **file name**, **file size**, **creator name** and **creation time** details of the attachments are retained.
- Each user on the **CC** list of an issue is added to the **Users Monitoring** list of the corresponding artifact.

This table shows how fields in CEE map to fields in CTF 8.2:

CEE field	CTF field
Summary	Title
First Description	Description
Priority	Priority
Status - Unconfirmed/New/Started/Reopened/Resolved/Verified/Closed	Status - Unconfirmed/New/Started/Reopened/Resolved/Verified/Closed
State - Open/Close	State - Open/Close

CEE field	CTF field
Issue Depends On	Associations
Issue Blocking	Associations
Issue Closed Date	Artifact Closed Date
Issue Activities	Change Log tab
Issue Comments	Status/Comments tab
Attachments	Attachments tab
Attachment - Description	Status/Comments tab
Cc Users	Monitoring Users
None	Group
None	Category
None	Customer
None	Reported in Release
None	Fixed in Release
Zero	Fixed in Release
Zero	Actual Hours

New fields created in CollabNet TeamForge:

- URL
- Issue Type
- OS
- Component
- Subcomponent
- Platform
- Version
- Resolution
- Target Milestone
- QA Contact
- Status Whiteboard
- Keywords

Notes

- In CTF 8.2, you can monitor an artifact even after you are no longer assigned to it. You must explicitly stop monitoring, if desired.

  👉 **Note:** In CEE, you do not have to be a project member to be in the "Assigned To" list. However, in CTF 8.2, you must be a project member to be in the "Assigned To" list.
- In CTF 8.2, only the project members can be part of monitoring and "Assigned To" lists.
- Internal users, for example, "issues@ert" in CEE are mapped to "None" in CTF 8.2.
- CEE Projects that have set up the "Unconfirmed" state to be the initial status of an artifact are migrated with a change. The Unconfirmed issues in CEE are migrated as Unconfirmed in CTF 8.2, but the "Unconfirmed" configuration setting is not migrated to CTF 8.2. So, that means, no confirmation is required in CTF 8.2.

  👉 **Note:** In CTF 8.2, an artifact can always be created only in "New" state (and not in "Unconfirmed" state as possible in CEE); as the user must have the "Submit" permission to create an artifact in CTF 8.2.

- "QA Contact" is a single-select field in CEE, but is made as multi-select field in CTF 8.2.
- History is not retained for "QA Contact", "Issue Owner", "Cc" and "Dependencies" (dependent/blocker issues) fields.
- Votes are not retained. CTF 8.2 does not support this feature.
- If the mime types in CEE attachments are not mapped in correspondence with the attachment type; post-migration, the attachments may not open appropriately. However, the attachments get migrated and can be saved locally and viewed.
- The email notification options set in CEE are not retained while migrating to CTF 8.2.
- Associations in CEE between Component-Subcomponent, Component-Versions and Component-Target Milestones are not retained post-migration.

Migrating discussions

Here are the migration details and migration notes for discussions.

Migration Overview

- All the active discussion forums, topics, posts, user subscribers and attachments are migrated to CTF 8.2.
- Attachment formats are retained.
- All the discussion user subscribers in CEE are added to the **Monitoring Users** list in CTF 8.2.
- Un-moderated and moderated discussions are migrated appropriately. However, the trusted discussions are migrated as moderated discussions and trusted subscribers as trusted users.
- Approved and waiting-for-approval topics/posts are migrated as is.
- Only those users who had the "Discussions - Discussion Use Private" permission in CEE can view the migrated private discussions in CTF 8.2.

This table shows how fields in CEE map to fields in CTF 8.2:

CEE	CTF 8.2
Title	Forum Name
Email address	Email Address
Description	Description
Restrictions	Enable Moderation/Type
Discussion User Subscribers	Monitoring Users
Un-moderated/Moderated Discussions	Un-moderated/Moderated Discussions
Trusted discussions/trusted subscribers	Moderated discussions/trusted users
Discussion Description	Forum Description
Prefix	Subject Prefix
Email Posting	Enable Mailing List
Email Address	Email Address
Footer	Footer Text

While posting attachments via the user interface in CTF 8.2, attachments can only be posted one at a time. However, while posting using email, you can post multiple attachments.

This table lists the message posting mapping details:

CEE	CTF 8.2
Subject	Subject

CEE	CTF 8.2
Body	Message
Attachments	Attachments
Other Recipients	Nil

Notes

- The discussion mailing address format in CEE is `forumalias@project.domain` whereas in CTF 8.2 it is `forumalias-project@domain`.

  ☞ **Note:** Post-migration, the response emails to the existing CEE mailing address will be re-directed to the new CTF 8.2 mailing address and captured into appropriate mailing lists. However, it's recommended to use the new CTF 8.2 mailing address for any new posts. The future releases of CTF 8.2 may not continue supporting the old mailing address compatibility.

  ☞ **Tip:** A CEE forum list name can start with a number or "_" and can also have "." and "-" characters, whereas, CTF 8.2 imposes certain naming conventions. So, if the toAddressUsername (list name) in CEE contains "." or "-", it is replaced with "_" in CTF 8.2. If it starts with "_" in CEE, the underscore is removed in CTF 8.2. If it starts with a number, post-migration, the number is replaced with the number in word, suffixed by an underscore. For example, "1team" mailing list address becomes "one_team" post-migration.
- The posts made by guest users in CEE are migrated with their actual mail-ids being prefixed in their messages.
- The **Default Discussions** feature is not available in CTF 8.2. The different default discussions (Issues/Commits) serve as archives for any change to issues/version control commits. These are migrated to CTF 8.2 but post-migration, these forums do not continue as archives.
- Any form (approved/un-approved) of soft-deleted posts/topics/forums is not migrated to CTF 8.2 as CTF 8.2 does not have soft deletion feature.
- The CJK file names are not preserved.
- Subject Prefix and Footer text fields available in CEE's create page are mapped in CTF 8.2's edit page.
- Email messages can not be blocked based on attachments in CTF 8.2.
- In CEE, "Content in email" has two options: Send full text of posted messages and send email notification with link only. In CTF 8.2, send full text is the default email content behavior
- In CEE, "Email notifications" has three options: Message-by-message, digest format and start page only. In CTF 8.2, email notifications are set to "message-per-change" by default.
- In CEE, there is no character length restriction on "Topic Title", whereas, in CTF 8.2, it is restricted to 250 characters.
- In CEE, subscription is allowed at the discussion level whereas in CTF 8.2, it is allowed even at the thread level.
- CEE features that are not available in CTF 8.2 include: Default Discussions, Message size, Message Header, Soft deletion, Guest Posting, User Groups, Anonymous and Sub-list subscribers, Email Subscription, Other Recipients, Web Post CAPTCHA, Remove deleted messages and Remove unmoderated messages.
- In CEE, "Reply Behavior" has four options: Reply to list, To sender only, specific header and do not reply to. In CTF 8.2, Reply to post is the only reply behavior available.
- CTF 8.2 does not support CEE's advanced email filter options like mime types, headers, reject content.
- In CTF 8.2, based on the user's permissions, web UI and/or web posting can be accessed.
- By default, forum mailing list is displayed in CTF 8.2 as the display email address.

Migrating documents

Here are the migration details and migration notes for documents and files.

Migration Overview

- In CEE the text field supports three formats: "Default", "Exactly as entered" and "HTML source". In CTF 8.2, there are two types of document mime types: "Text/Plain" and "Text/Html".
- The file format selected as "html source" in CEE is migrated to CTF 8.2 as an attachment.

- The CEE document files with "Default" and "Exactly as entered" format type are displayed as "Text" format type in CTF 8.2.

  ☞ **Note:** These files can be edited by selecting the **Create online** option under **Update Document**.

- The CEE document files with "Html source" format type are displayed as "Html" format type in CTF 8.2.

  ☞ **Note:** These files can be edited by selecting the **Upload from file** option under **Update Document**.

- In CEE, a document can be in either of the following states:

  - Draft
  - Reviewed
  - Baselined
  - Archival
  - Obsolete
  - Stable

  . In CTF 8.2, a document can only be in either "Draft" or "Final" state.

- All CEE documents except those in "Stable" state are moved over to CTF 8.2 as "Draft" documents. The "Stable" documents are moved over as "Final" file status.

  ☞ **Note:** The "Obsolete" state may have a prefix identifier in the "Draft" state.

- The folder hierarchy is retained.
- The file version of moved files is set as version 1.
- Locked documents are retained as locked.

  ☞ **Note:** If a document in CEE is set with a future date for locking, when migrated, it will be migrated as an un-locked document only. Locked documents can be viewed but not edited.

This table shows how fields in CEE map to fields in CTF 8.2:

CEE	CTF 8.2
File formats: • Default • Exactly as entered • Html source	File formats: • Text/Plain • Text/Html
File Status: • Draft • Reviewed • Baselined • Archival • Obsolete • Stable	File Status: • Draft • Final
Pre-migration document states: Draft, Reviewed, Baselined, Archival or Obsolete	Post migration document state: Draft
Pre-migration document state: Stable	Post migration document state: Final
First document created date	Folder creation date
Folder name	Document Folder Name
Name	Document Name
Status	Status

CEE	CTF 8.2
Description	Description
Text	Create Online
Attachments	Upload from File
Link	Enter Document URL

Notes

- The CEE features like infinite lock and a specific period are not mapped to CTF 8.2.
- While creating documents, version comments can be entered in CTF 8.2.
- In CEE, the file format can not be changed while editing the file, whereas, in CTF 8.2, the file format is editable as well.
- In CTF 8.2, version comments also can be specified.
- The CEE features that are not available in CTF 8.2 include: Advisory/Strict lock, Future date setting and Folder description.

Migrating announcements
Here are the migration details and migration notes for announcements.

Migration Overview

- Announcements in CEE are migrated as News in CTF 8.2.
- Image URLs, publishing and event dates in CEE announcements are migrated as news content in CTF 8.2.
- If the Project Home page is configured via remote publishing (using index.html), the News component would not appear in the Home Page. However, the news component would appear in My Workspace in CTF 8.2.
- CEE project owner's message is migrated to the CTF 8.2 news component as project news.

This table shows how fields in CEE map to fields in CTF 8.2:

CEE	CTF 8.2
Announcements	News
Headline	Title – character limit is 250
Body	Body
Announcements Modified Date	News Creation Date
Image URLs, Event Dates	News - Body

Notes

- In CEE, Announcements is available as an independent tool, whereas, in CTF 8.2, News is available as a project page component.
- For public projects in CTF 8.2, news is displayed in Login page even after logout.
- Advanced options for announcements like publishing and event dates are not migrated to CTF 8.2. as CTF 8.2 doesn't support these features.
- News post headlines are migrated only up to the first 250 characters.
- Deleted announcements are not migrated from CEE to CTF 8.2.

Migrating project dashboard
Here are the migration details and migration notes for project dashboard.

Migration Overview

- CEE's PD/Task is not migrated to CTF 8.2. However, the customers can access all the versions of imported and synchronized MPP files which they had in CEE. Post migration, any version of the file can be chosen and imported manually into the corresponding project in CTF 8.2.
- The MPP files of a project can be found under the "plans" directory of the respective project's CVS or SVN repository. The version number in the sync file's name corresponds to a repository revision of the plan file.
- The files will have the following naming convention:

  - Plan files: `<domain name>_<project name>_plan.mpp`
  - Sync files: `<domain name>_project name>_sync_<version no.>.mpp`

    ☞ **Tip:** As an example, "mydomain_myproject_sync_1.18.mpp" will be the sync file corresponding to the revision "1.18" of plan file, "mydomain_myproject_plan.mpp".

- Any revision of the plan file or a sync file can be retrieved from the repository.
- You can take a backup of the exception report from the following location: `<source folder for cee dump and load directory>/backup/<domain name>/projects/<projectname>/projectname-REPORT.txt`
- The global exception report can be accessed from: `<source folder for cee dump and load directory>/backup/<domain name>/global/GLOBAL-REPORT-CONSOLIDATED.txt`

Migrating remote publishing tool
Here are the migration details and migration notes for remote publishing tool.

Migration Overview

- For PCE-enabled subversion projects, CTF Home Page is replaced with CEE Home Page.
- For CVS and non-PCE enabled subversion projects, CTF Home Page is replaced with remote publishing content (index.html).

  ☞ **Note:** If PCE is enabled in CEE, by default the "Project Page" option appears selected in CTF's **Project Admin** > **Project Settings** page. If PCE is not enabled in CEE, the remote publishing content is displayed in CTF.

- The remote publishing content of CEE is migrated to the "Publishing" repository of CTF.

Notes

- In CTF, there is no equivalent functionality for governing the access to remote published content. Hence, the corresponding CEE permissions (resource patterns) are dropped and anyone with project view page permission is allowed to view the remote published content in CTF.

Mapping CollabNet Enterprise Edition's Host or Domain Configure settings to CollabNet TeamForge 8.2
The mapping details are given in respective tables.

Note about SAS-70

To meet SAS-70 certification requirement,the following criteria need to be met:

- After 20 bad logons, account must lock
- Reset passwords must be changed on next logon
- Cannot reuse previous three passwords
- Passwords expire every 90 days
- Password must be at least 8 alphanumeric digits long
- Passwords must contain at least one of each of these:

  - At least one lower case letter
  - At least one upper case letter
  - At least one number (0-9)

- At least one special character (e.g. !@#$%&*()+)

While migrating the CEE password settings to CTF, if the CEE setting is more liberal than the SAS-70 requirement, the SAS-70 minimum requirement setting is chosen; otherwise the CEE setting is taken.

Here's how the CTF site's password settings are selected:

- If 'Authentication failures before lockout' domain setting was set to <= 0 or > 20, then set CTF's LOGIN_ATTEMPT_LOCK token to 20. Same value otherwise.
- If 'Size of user's password history' was set to < 3, then set CTF's PASSWORD_HISTORY_AGE token to 3.
- If 'Size of user's password history' was set to > 10, then set CTF's PASSWORD_HISTORY_AGE token to 10.
- If 'Size of user's password history' was set between 3 and 10, then set CTF's PASSWORD_HISTORY_AGE to the CEE value.
- If 'Minimum password length' was set to < 8, then set MINIMUM_PASSWORD_LENGTH to 8. Same vale otherwise.
- Set PASSWORD_REQUIRES_MIXED_CASE, PASSWORD_REQUIRES_NON_ALPHANUM and PASSWORD_REQUIRES_NUMBER to true regardless of CEE setting.
- Set ALLOW_PASSWORD_DICTIONARY_WORD to false regardless of CEE setting.
- If 'Password expiration period for non-admins (days)' was set to <= 0 or > 90, then set history.age to 90 in ctf-pass.conf. Same value otherwise.

This table shows the mapping of CEE's Host or Domain Configure settings to CTF 8.2's "runtime-options.conf" file.

CEE's Host or Domain Configure Settings	CTF 8.2 Tokens	Comments
Size of user's password history	PASSWORD_HISTORY_AGE	The CEE value is transferred to CTF 8.2.
Authentication failures before lockout	LOGIN_ATTEMPT_LOCK	When the CEE value is "0", the CTF 8.2 value is set to "5", else the same CEE value is transferred.
Minimum password length	MINIMUM_PASSWORD_LENGTH	The CEE value is transferred to CTF 8.2.
Allow user join	DISABLE_USER_SELF_CREATION	When the CEE value is "true", the CTF 8.2 value is set to "false"; else it is set to "true".
Project path separator string	PROJECT_PATH_SEPARATOR	The CTF 8.2 value is hard coded to ">".
Suppress gathering of full name and organization	INCLUDE_ORGANIZATION_USER_FIELD	When the CEE value is "true", the CTF 8.2 value is set to "false"; else it is set to "true".
Users can edit own organizations	ORGANIZATION_EDITABLE	The CEE value is transferred to CTF 8.2.
Number of required character classes	PASSWORD_REQUIRES_MIXED_CASE	When the CEE value is "<=1" or "2", the CTF 8.2 value is set to "false"; else it is set to "true".
Number of required character classes	PASSWORD_REQUIRES_NON_ALPHANUM	When the CEE value is "<=1" or "2" or "3", the CTF 8.2 value is set to "false"; else it is set to "true".
Number of required character classes	PASSWORD_REQUIRES_NUMBER	When the CEE value is "<=1", the CTF 8.2 value is set to "false"; else it is set to "true".

CEE's Host or Domain Configure Settings	CTF 8.2 Tokens	Comments
Reject passwords based on dictionary words	ALLOW_PASSWORD_DICTIONARY_WORD	When CEE value is "true", the CTF 8.2 value is set to "false"; else it is set to "true".
Password ticket expiration period for non-admins (days)	ACTIVATION_CODE_TIMEOUT	When the CEE value is "<=0", then set the CTF 8.2 value as 336. CEE value is multiplied by 24 (converting days to hours); else the same CEE value is transferred.
Password ticket expiration period for non-admins (days)	INITIAL_PASSWORD_CHANGE_ACTIVATION_CODE_TIMEOUT	When the CEE value is "<=0", then set the CTF 8.2 value as 336. CEE value is multiplied by 24 (converting days to hours); else the same CEE value is transferred.
Allow guest access/Allow anonymous access to domain home	ALLOW_ANONYMOUS_ACCESS_TO_DOMAIN_HOME	When either of the CEE values is "true", the CTF 8.2 value is set to "true"; else it is set to "false".

This table shows the mapping of CEE's Host or Domain Configure settings to CTF 8.2's "/etc/ctf-pass.conf" file:

CEE's Host or Domain Configure Settings	CTF 8.2 Properties	Comments
Password expiration period for non-admins (days)	history.age	The CEE value is transferred to CTF 8.2.
Users with non-expiring password	special.users	Same CEE value must be added to the existing users of CTF 8.2.
Password expiry warning period (days)	history.warn	The CEE value is transferred to CTF 8.2.

How is global data handled during migration?

Global (or "framework") data is validated for integrity, correctness and completeness after migration from Enterprise Edition (CEE) to TeamForge 8.2 (CTF 8.2). Here are the detailed assumptions behind the validation process.

- This includes; category, project, project group, global announcements, global SCM settings, global project roles, project tracker's global data, users and user groups.
- As part of data validation, any mismatch of global value count between CEE and CTF is reported in "Global.log". Absence of global value reference in CTF tables is reported in "Global Validate.log".
- CEE's HAC_AUDIT_LOG_ENTRY table is moved to cee_backedup_audit_log.

Migrating Issue Tracker

When Issue Tracker data is migrated from CollabNet Enterprise Edition to TeamForge, you will see these changes.

Common features

Here's a list of what you had in Issue Tracker and what you now have instead in TeamForge tracker.

- Issue Tracker in CEE is migrated to a tracker called "Issue Tracker" in CTF.
- Each CEE issue becomes an artifact in 'Issue Tracker' tracker.
- Each CEE issue can be identified with its corresponding artifact, as issue Id appears as a prefix in the corresponding artifact's title. For example, [issue 1].

CEE field	CTF field
Summary	Title
First Description	Description

Dissimilarities

Some behavior in TeamForge tracker is different from Issue Tracker functionality.

- In CTF, you can monitor an artifact even after you are no longer assigned to it. You must explicitly stop monitoring, if desired. This behavior is different from CEE.
- In CTF, only the project members can be part of monitoring and "Assigned To" lists.

CEE field	CTF behavior
QA Contact	Single-select field in CEE, but is made as multi-select field in CTF.
Cc and Dependencies (dependent/blocker issues) fields	History is not retained for both these fields.

User permissions on a CEE and CTF 8.2 site

Each permission a user can have on a CollabNet Enterprise Edition site has an equivalent permission on a CollabNet TeamForge site, with a few exceptions.

Notes

- For integrations on TeamForge, users with the "Domain - Administer" permission on CEE get the "SCM Admin" and "SWLA Admin" permissions.
- CEE's Guest user is not migrated.
- CEE's Registered User and Anonymous Guest domain roles are dropped in favor of Default Access Permissions.
- Permissions of CEE's Registered User role are transferred to CTF 8.2's project's Default Access Permissions. Project is created as public (User class: "All logged in users") if the guest access was turned ON in CEE, or as gated (User class: "All project members and all non-restricted users") otherwise.
- Permissions of CEE's Anonymous Guest role are transferred to CTF 8.2's public project's Default Access Permissions (User class: "All users") if the guest access was turned ON in CEE.
- CEE's Host users with Host Admin role are migrated as CTF 8.2's site Admins.
- CEE's other host users are migrated as non-adminstrators in CTF 8.2.
- The CEE's host roles that are dropped are captured in the exception report.
- CEE and CTF 8.2 permission mapping details are configurable.
- There are no permissions (actions) for CUBiT, they are just roles without any permissions.

TeamForge 8.2 Permissions for Authentication Realms

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Authentication Realm - Add		This feature is not available in TeamForge 8.2		There isn't any corresponding UI option. This is a configurable option in TeamForge 8.2.

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Authentication Realm - Delete		This feature is not available in TeamForge 8.2		There isn't any corresponding UI option. This is a configurable option in TeamForge 8.2.
Authentication Realm - Edit		This feature is not available in TeamForge 8.2		There isn't any corresponding UI option. This is a configurable option in TeamForge 8.2.
Authentication Realm - Map User		This feature is not available in TeamForge 8.2		There isn't any corresponding UI option. This is a configurable option in TeamForge 8.2.
Authentication Realm - View		This feature is not available in TeamForge 8.2		There isn't any corresponding UI option. This is a configurable option in TeamForge 8.2.

TeamForge 8.2 Permissions for Category

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Category - Add	Create/Edit/View	User gains permissions	The user will get the permissions to edit the name and description.	This is available through site-wide roles.

TeamForge 8.2 Permissions for Domain

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Domain - Administer	System Tools Permissions	User loses permission	The user may lose some permissions.	Conditional mapping. This permission is granted if the user also has "Audit Log – View" and "Session – List" permissions.
Domain Page - View		User gains permission	Gains the ability to view the community page if the user	This permission is used to control who can view the Community tab.

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
			does not have this permission.	As this is part of the out-of-the-box "Registered User" role, dropping this is safe in most of the cases.
Domain Reports - View		CollabNet Enterprise Edition permission is retained		This permission is not being used in CollabNet Enterprise Edition.

TeamForge 8.2 Permissions for License

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Licenses - Administer		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.

TeamForge 8.2 Permissions for Project and Project Group

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Project - Approve	Projects: Projects Admin	User loses permission	The user loses the permission if the role does not have "Project – Delete" permission.	Conditional mapping. This permission is available through a site-wide role. It is granted only if the role also has the "Project – Delete" permission.
Project - Suggest		User gains permission	All users will be able to suggest projects	Project suggest is not controlled through permissions in TeamForge 8.2.
ProjectGroup - Add		This feature is not available in TeamForge 8.2		TeamForge 8.2 does not have any such feature.

TeamForge 8.2 Permissions for Session

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Session - List		User loses permission	The user loses the ability if the user does not have the "Domain – Administer" permission.	This permission is handled in the mapping for "Domain - Administer".

TeamForge 8.2 Permissions for Start Page

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
Start Page - View		User gains permission	The user will have access to **My Workspace**, but is unlikely since this is part of the "Registered User" role.	TeamForge does not use permissions for this page.

TeamForge 8.2 Permissions for User

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
User - Add	Create/Edit/View	User loses permission	The user loses the ability if the role does not have the "User – Edit" permission.	Conditional mapping. This permission is granted if the user also has the "User – Edit" permission.
User - Delete	Delete/View	CollabNet Enterprise Edition permission is retained		
User - Edit		User loses permission	The user loses the ability if the role does not have the "User – Add" permission.	This is handled in the mapping for the "User – Add" permission.
User - Edit - Self	Pluggable: Edit user preferences	CollabNet Enterprise Edition permission is retained		
User - Edit - Self	TeamForge 8.2: Drop	User gains permission	Roles not having this permission get the ability	TeamForge 8.2 does not have any such feature.
User - List		User loses permission	The user loses the ability unless the user	In TeamForge 8.2, there isn't any

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
			gets "Create/Edit/ View" or "Delete/ View" permissions.	separate cluster for the view permission.
User - View		User gains permission	All users will be able to see any other user in the system	TeamForge 8.2 doesn't have such permission control.
User - View - Self		User gains permission	All users will be able to see their profiles	TeamForge 8.2 doesn't have such permission control.

TeamForge 8.2 Permissions for User Group

Permission Mapping Details

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
UserGroup - Add	Create/Edit/View	User loses permission	The user loses the ability if the role does not have "UserGroup – Edit" and "Add/ Delete Element" permissions	Conditional mapping. This permission is granted if the role also has "UserGroup – Edit", "UserGroup Element - Add" and "UserGroup Element – Delete" permissions.
UserGroup - Delete	Delete/View	CollabNet Enterprise Edition permission is retained		
UserGroup - Edit	Create/Edit/View	User loses permission	The user loses the ability if the role does not have "UserGroup – Add", "UserGroup Element - Add" and "UserGroup Element – Delete" permissions.	Conditional mapping. This permission is granted if the role also has "UserGroup – Add", "UserGroup Element - Add" and "UserGroup Element – Delete" permissions.
UserGroup - List	View Only	CollabNet Enterprise Edition permission is retained		
UserGroup - View	View Only	User gains permission	The user gains additional access for Groups list.	
UserGroup Element - Add		User loses permission	The user loses the ability unless the role gets "Create/Edit/ View" permissions.	

CollabNet Enterprise Edition Permission	TeamForge 8.2 Permission	Change in Behavior	Functionality Loss/ Gain Details	Comments
UserGroup Element - Delete		User loses permission	The user loses the ability unless the role gets "Create/Edit/ View" permissions.	

CollabNet TeamForge full-text search

Here is some detailed information to help you make the most of TeamForge's full-text search capabilities powered by the Apache Lucene full-text search engine.

👉 **Note:** For more information about Apache Lucene, refer to *Apache Lucene Documentation*.

An overview of the TeamForge full-text search

Here is some detailed information about the TeamForge's full-text search.

Searchable items in TeamForge

The following items are search-able in TeamForge.

- Discussion forums, posts and topics
- Documents and document folders
- File releases, packages, and FRS files
- News posts
- Visible project pages
- Projects
- Source code files and commits[#]
- Tasks and task folders
- Trackers and tracker artifacts
- Users
- Wiki pages
- Integrated applications[#]

[#]**Source code**	You can search source code repositories by integrating the TeamForge with Black Duck Code Sight. However, Black Duck Code Sight search function may differ from the TeamForge search function. For more information, refer to the *Black Duck Code Sight*.
[#]**Integrated applications**	Integrated applications may or may not have search capabilities. Refer to the integrated application's documentation to know more about its search features and to appreciate how the search features of the integrated application and TeamForge differ.
Attached files	You can search the contents of attached files. For more information about supported document formats, *click here*.

Notes

You may witness some delay for the TeamForge objects to appear in the search results and the extent of delay depends on the load on the server.

When you search, the contents of all the search-able fields (of an object) are collectively searched for matches. For instance, when you search for an artifact, the contents of the title, description and comment fields, put together, are searched. As an example, the search entry `CollabNet AND TeamForge` returns an artifact if the content of its title, description and comment, put together, has both the words "CollabNet" and "TeamForge". In other words, the word "CollabNet" could be in the artifact title and the word "TeamForge" could be in the artifact description (and not necessarily present on the same field).

If you search with multiple words, items containing any of the words in the search string are returned. For more information, refer to *multiple-terms search*. On the other hand, if you want to find items where the words, say `CollabNet` and `TeamForge`, both appear, type `CollabNet AND TeamForge`. For more information, refer to *Boolean operators*.

The TeamForge searches for full words. Use *wildcards* for partial word searches.

Search terms are case-insensitive. For example, if you search using the keyword "`collabnet`", pages that contain `COLLABNET`, `CollabNet` and `collabnet` are all returned.

TeamForge full-text search guidelines
Here is some guidelines to help you create effective searches.

Single term, multiple terms and phrase

Single-term search looks for all search results that match the search text. For example, a search entry of `doc` only returns search results of "doc".

Multiple-terms search looks for all search results that match any of the words in the search text. For example, a search entry of `document plan` returns search results of "document", "plan", and "document plan".

A group of words surrounded by double quotes, such as `"product requirements"`, return only search results containing the entire phrase.

Boolean operators

Terms and phrases can be combined with Boolean operators for more complex searches. Boolean operators must be in upper case. Use:

- `OR` between two terms returns search results containing either of the terms. This is the default operator used if no other operator is specified.
- `AND` between two terms returns only search results containing both of the terms.
- The + operator before a term makes the term required. Only search results containing the terms are returned.
- The – or `NOT` operator before a term returns only search results that do not contain the term. The character "-" represents the Boolean operator AND NOT.

☞ **Tip:**

You can group Boolean searches using parentheses. For example, `(doc OR test) AND plan` returns search results containing "doc plan" and "test plan".

Wildcard searches

To look for search results with a single character replaced, use the `?` symbol. For example, to look for search results with "text" or "test", enter `te?t`.

To look for search results with more than one character replaced, use the `*` symbol. For example, to look for search results such as "content" or "contest" or "continuous" or "control", enter `cont*`.

☞ **Note:**

You can use wildcard symbols in the middle or at the end of a search, but not as the first character of a search keyword.

Fuzzy searches

To look for search results with spelling similar to the search term entered, use the ~ symbol as the last character of the search keyword. For example, to look for search results with spelling similar to "roam", enter `roam~`. This returns search results such as "roam" and "roams".

Special characters

If you have any of the following special characters in your search text, you must escape them by enclosing the entire phrase in double quotes. `+ - & || ! ( ) { } [ ] ^ " ~ * ? : \`

For example, to look for search results containing the hyphenated term "product-development", enter `"product-development"`.

The special character "+" represents the Boolean operator AND. The special character "-" represents the Boolean operator AND NOT.

Regular expression search with forward slashes

Lucene 4 supports regular expression searches matching a pattern between forward slashes "/". For example, to look for search results containing the words "moat" or "boat", use the search string `/[mb]oat/`.

If you are specifically looking for search results containing a forward slash "/" character, you must backslash-escape or quote-escape the forward slash character. For example, to look for search results containing "<opt/collabnet>", use the search string `<opt\/collabnet>`.

Excluded words

The following words are considered stop words and are not search-able on their own: `a, an, and, are, as, at, be, but, by, for, if, in, into, is, it, no, not, of, on, or, s, such, that, the, their, then, there, these, they, this, to, was, will, with`

Range searches

You can do a range-bound search using the `TO` operator. For example, the search entry, `[artf1100 TO artf1200]`, returns items containing values between artf1100 and artf1200, including artf1100 and artf1200. To exclude the upper and lower bounds from the search results, use curly brackets {} instead of square brackets [].

Install, Upgrade and Site Administration

TeamForge install/upgrade how-to

A system administrator provides the infrastructure that lets site administrators, project managers and members collaborate effectively.

Install TeamForge 8.2

When you finish the sequence of tasks given below, you will have a working TeamForge 8.2 site tailored to the specific requirements of your user base and environment.

Installing TeamForge is not difficult, but it can be complex. To succeed, you should be familiar with the essential Linux commands and terminology. Most of the commands are given explicitly, but sometimes you have to substitute the values corresponding to your own specific environment.

☞ **Note:** For the hardware and software required to run TeamForge, see *Platform specification for TeamForge 8.2* on page 747.

Plan your installation

Before you install TeamForge 8.2, let's take a look at the product from a system administrator's perspective, so that you know exactly what you are getting into.

Overview

A TeamForge site consists of a core TeamForge application and several tightly integrated services that support it. In addition, you can integrate TeamForge with other third party applications such Review Board, Nexus, Black Duck Code Sight, Git, and so on.

- The core TeamForge application provides the Web interface that users see, and the API that other applications can interact with. It also includes the file system where some user content is stored, such as wiki pages.
- The site database is where most of the user-created content is stored and accessed. Documents, discussion posts, tracker artifacts, project administration settings: all that sort of thing lives in the database.
- The source control server ties any number of Subversion or CVS repositories into the TeamForge site.
- The Extract Transform and Load (ETL) server pulls data from the site database and populates the datamart to generate charts and graphs about how people are using the site.
- The datamart (Reports DB) is an abstraction of the site database, optimized to support the reporting functionality.

Integrations

TeamForge supports integration with a wide array of third party applications such as Review Board, Git, Black Duck Code Sight, Nexus, Jira and so on. As a customer you may or may not always want (or have) all of TeamForge's supported integrated applications. It's also quite possible that some of the integrated applications may not always run on all the platforms supported by TeamForge.

To accommodate a wider audience, by default, TeamForge install and upgrade instructions include steps to integrate such third party applications with TeamForge. However, use your discretion to ignore and skip such steps if they are not relevant to your site. See *Software requirements for CollabNet TeamForge 8.2* on page 748 to understand what it takes to run TeamForge 8.2 with integrations.

☞ **Important:** To activate Black Duck Code Sight, a dedicated Black Duck Code Sight licensing key is required. To receive a Black Duck Code Sight license key, contact your CollabNet account manager, or send an email request to info@collab.net. (A commercial TeamForge license is required, i.e. TeamForge "free option license" does not qualify).

Plan your TeamForge installation setup

TeamForge supports multiple options for customizing and expanding your site to fit your organization's unique use patterns. TeamForge 8.2 can run on RHEL/CentOS 6.7 or 7.2.

In the default setup, all services run on the same server as the main TeamForge application. But in practice, only the TeamForge application *needs* to run on the TeamForge application server. The other services can share that server or run on other servers, in almost any combination. When you distribute your services on multiple servers, you must do some configuration to handle communication among the services.

Install considerations for RHEL/CentOS 6.7 and 7.2

Though TeamForge 8.2 can run on RHEL/CentOS 6.7, it is a best practice to install TeamForge 8.2 on the most recent version of RHEL/CentOS supported by TeamForge, which in this case is RHEL/CentOS 7.2. However, choosing between RHEL/CentOS 6.7 and 7.2 depends on whether or not you need Black Duck Code Sight.

As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, CollabNet recommends that you install Black Duck Code Sight on a separate RHEL/CentOS 6.7 server if you choose to install TeamForge 8.2 on a RHEL/CentOS 7.2 server. The following flowchart can help you decide how you can proceed.

In practice, it is always recommended to run Black Duck Code Sight on a separate server. Assess your own site's particular use patterns and resources to decide how to distribute your services, if at all. For example, if you anticipate heavy use of your site, you will want to consider running the site database, the source control service, or the reporting engine on separate hardware to help balance the load.

PostgreSQL or Oracle?

PostgreSQL 9.3.6 is installed automatically when you install TeamForge 8.2. If you intend to use Oracle, CollabNet recommends that you let the installer run its course, make sure things work normally, and then set up your Oracle database and switch over to it.

If you want to use Oracle as your database, consider the following points:

- TeamForge 8.2 supports Oracle server 12c and Oracle client 12c.
- Oracle express edition is not supported for both client and server.
- Black Duck Code Sight (BDCS) 2.1.3 was tested with PostgreSQL 9.3.6 only. BDCS with Oracle was not tested. If you want BDCS, you may install BDCS on the PostgreSQL database that is installed by default during TeamForge installation.
- Review Board 2.0.11 was tested with PostgreSQL 9.3.6 only. Review Board with Oracle was not tested. Note that Review Board must be installed on the TeamForge application server and therefore uses the PostgreSQL database that is installed by default during TeamForge installation.
- GIT integration works only with PostgreSQL. The Git integration will use PostgreSQL even if your TeamForge site uses Oracle.

The efficiency of your database can have an impact on your users' perception of the site's usability. If your site uses a PostgreSQL database (which is the default), you may want to consider tuning it to fit your specific circumstances. The default settings are intended for a small-to-medium site running on a single server. See *What are the right PostgreSQL settings for my site?* on page 909 for recommendations from CollabNet's performance team on optimizing PostgreSQL for different conditions.

Choose your hardware

TeamForge can run on a wide range of hardware configurations.

- For a small team, you can install it on any laptop that can run VMware Player.
- In a large organization, you may need multi-processor hardware with NFS storage and multiple layers of redundancy.

Most sites will need something in between. For the minimal requirements, see *Hardware requirements for CollabNet TeamForge 8.2* on page 747.

Set up networking for your TeamForge server

After installing the operating system, prepare the networking connections and configuration that your TeamForge 8.2 site will require.

☞ **Note:** You must have root access to all the hosts you will be setting up for your site.

1. Open the appropriate ports, and close all other ports.

   ⚠ **Caution:** Expose only the JBOSS and Tomcat ports that are required for integration with another application, and open them only to that specific host IP address, even within your internal network.

   For detailed port requirements, see *Which ports are required to be open for a TeamForge site?* on page 935

2. Use the `hostname` command to verify that the machine name is resolvable on the network.

   ```
   hostname
   bigbox.supervillain.org
   ```

3. Use the `nslookup` command to verify that your hostname maps to the right IP address.

   ```
   nslookup bigbox.supervillain.org
   Server: 204.16.107.137
   Address: 204.16.107.137#53
   ```

   ☞ **Tip:** If there is any doubt about what the system's real IP address is, use the `/sbin/ifconfig` command.

4. If you are installing behind a proxy, specify your proxy settings.

   ```
   export http_proxy=http://
   <PROXY_USERNAME>:<PROXY_PASSWD>@<PROXY_HOST>:<PROXY_PORT>
   export no_proxy=localhost,127.0.0.0/8,<hostname>
   ```

5. If any mail service is running on port 25, stop it and make sure it won't restart.

For example:

```
/sbin/service sendmail stop
/sbin/chkconfig sendmail off
```

6. Use a tool such as Nessus to scan your server for potential vulnerabilities.

(See *Which ports are required to be open for a TeamForge site?* on page 935 for detailed security recommendations.)

Install TeamForge 8.2 on RHEL/CentOS

You can install TeamForge in a dedicated mode. In dedicated installation mode, the server is fully dedicated to the TeamForge application. This makes the installation process very simple as the TeamForge installer takes care of configuring your Apache, Postgres, SELinux (if enabled) automatically without any manual intervention.

In a "Dedicated" install, it is advised not to install additional software applications other than TeamForge. It is also recommended not to run any unwanted services that are not used by TeamForge.

Install TeamForge 8.2: All services on a single RHEL/CentOS 7.2 server

The easiest way to install TeamForge is to install it on a single server, dedicated to TeamForge taking the default configuration settings.

☞ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.2, all the following services but Black Duck Code Sight run on a single RHEL/CentOS 7.2 server (we call this my.app.host). If required, you must install TeamForge 8.2 with Black Duck Code Sight on a separate RHEL/CentOS 6.7 server, see *Install TeamForge 8.2: Black Duck Code Sight on a separate server on RHEL/CentOS* on page 403.

*   TeamForge Application Server
*   Database Server (Operational DB and Reports DB)
*   ETL Server
*   GIT Integration Server
*   SCM Integration Server (Subversion and CVS)
*   Search Server (Indexer).

☞ **Remember:**

*   It is *critical* that you start with a fresh server, without any software installed. You must have root access to the server.
*   **Log on to the server as root user always.**

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

*   Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
*   The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
*   See *the Red Hat installation guide* for help.

☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```

Save the file and reboot the server.

3. Check your basic networking setup.

   See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

      ```
      getenforce
      ```

   b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

      ```
      setenforce 0
      ```

   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to /tmp.

   b) Install the repository package.

      ```
      yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
      ```

   c) Refresh your repository cache.

      ```
      yum clean all
      ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

      • Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

        ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

   b) Unpack the disconnected installation package.

      ```
      rpm -ivh <package-name>
      ```

   c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

      ```
      rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
      ```

   d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

   Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

   In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

      ```
      cd /media/
      mkdir cdrom
      mount /dev/cdrom ./cdrom/
      ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

   e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

      ```
      vi /etc/yum.repos.d/cdrom.repo
      ```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

6. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

b) GIT: To install the GIT packages run the following command.

```
yum install teamforge-git
```

7. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Configure the HOST token.

```
HOST_localhost=app database datamart etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

b) Add 'gerrit' to the HOST_localhost token if you are installing Git.

```
HOST_localhost=app database datamart etl indexer subversion cvs gerrit
```

c) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs
 gerrit binary
```

d) Configure the database and datamart settings.

☞ **Tip:** For more information about configuring variables, see *site-options.conf* on page 775

```
DATABASE_TYPE=postgresql
```

```
DATABASE_USERNAME=ctfuser
```

```
DATABASE_NAME=ctfdb
```

```
DATABASE_READ_ONLY_USER=ctfrouser
```

```
REPORTS_DATABASE_USERNAME=ctfrptuser
```

```
REPORTS_DATABASE_NAME=ctfrptdb
```

```
REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
```

```
REPORTS_DATABASE_MAX_POOL_SIZE=30
```

☞ **Note:** The database name and user name values are arbitrary alphanumeric strings.

e) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

f) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

g) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

h) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

    i)   Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

    ☞   **Important:** Remember to use the same key in the external SCM integration server also.

    j)   If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

        `ADMIN_EMAIL=root@{__APPLICATION_HOST__}`

    k) If you have LDAP set up for external authentication, you must set the *REQUIRE_USER_PASSWORD_CHANGE* site options token to false.

    l)   Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

    m) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

    n) Configure Black Duck Code Sight tokens if you are installing Black Duck Code Sight. See *Black Duck Code Sight site-option tokens*.

    o) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

    p) Ensure to set the token *SELINUX_SETUP=true*.

    q) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

    r) Save the `site-options.conf` file.

8. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

9. Set up the initial site data (bootstrap).

```
./bootstrap-data.sh
```

10. Start TeamForge.

```
/etc/init.d/collabnet start
```

  ☞   **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

11. ☞   **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

    Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

  ☞   **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

12. Restart the `collabnet` services.

```
/etc/init.d/collabnet restart
```

13. Apply some finishing touches and make sure everything is running smoothly.

    a) Reboot the server and make sure all services come up automatically at startup.

    b) Log into your site as the administrator.

       The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.

    c) Create a sample project.

See *Create a TeamForge project*.

d) Write a welcome message to your site's users.

See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Caching with Memcached**: *Install Memcached* on page 571

Install TeamForge 8.2: All services on a single RHEL/CentOS 6.7 server

The easiest way to install TeamForge is to install it on a single server, dedicated to TeamForge taking the default configuration settings. We call this a "dedicated" install.

☞ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

All the following services run on a single RHEL/CentOS 6.7 server (we call this my.app.host).

- TeamForge Application Server
- Black Duck Code Sight Server
- Database Server (Operational DB and Reports DB)
- ETL Server
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer).

☞ **Remember:**

- It is *critical* that you start with a fresh server, without any software installed. You must have root access to the server.
- **Log on to the server as root user always.**

1. Install Red Hat Enterprise Linux/CentOS 6.7 and log in as root.

   - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
   - See *the Red Hat installation guide* for help.

   ⚠ **Warning:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf,` and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

3. Check your basic networking setup.

   See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. ☞   **Important:** Do this only if you are upgrading TeamForge on RHEL/CentOS 6.7.

Delete the `python-crypto` package.

```
yum erase python-crypto
```

6. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.
   b) Install the repository package.

   ```
   yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
   ```
   c) Refresh your repository cache.

   ```
   yum clean all
   ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

   - Red Hat Enterprise Linux 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**
   - CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-182.el6.x86_64.rpm**

   b) Unpack the disconnected installation package.

   ```
   rpm -ivh <package-name>
   ```
   c) ☞   **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

   Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

   In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.
   d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here's a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```

   e) Verify your yum configuration files.

   ```
   yum list httpd
   yum list apr
   ```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

7. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

b) GIT: To install the GIT packages run the following command.

```
yum install teamforge-git
```

c) To install Black Duck Code Sight run the following command.

```
yum install teamforge-codesearch
```

8. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Configure the HOST token.

```
HOST_localhost=app database datamart etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

b) Add 'gerrit' to the HOST_localhost token if you are installing Git.

```
HOST_localhost=app database datamart etl indexer subversion cvs gerrit
```

c) Add 'codesearch' to the HOST_localhost token if you are installing Black Duck Code Sight.

```
HOST_localhost=app database datamart etl indexer subversion cvs
 gerrit codesearch
```

d) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs gerrit
 codesearch binary
```

e) Configure the database and datamart settings.

☞ **Tip:** For more information about configuring variables, see *site-options.conf* on page 775

```
DATABASE_TYPE=postgresql
```

```
DATABASE_USERNAME=ctfuser
```

```
DATABASE_NAME=ctfdb
```

```
DATABASE_READ_ONLY_USER=ctfrouser
```

```
REPORTS_DATABASE_USERNAME=ctfrptuser
```

```
REPORTS_DATABASE_NAME=ctfrptdb
```

```
REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
```

```
REPORTS_DATABASE_MAX_POOL_SIZE=30
```

☞ **Note:** The database name and user name values are arbitrary alphanumeric strings.

f) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

g) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

h) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

i) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

j) Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

☞ **Important:** Remember to use the same key in the external SCM integration server also.

k) If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

```
ADMIN_EMAIL=root@{__APPLICATION_HOST__}
```

l) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

m) Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

n) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

o) Configure Black Duck Code Sight tokens if you are installing Black Duck Code Sight. See *Black Duck Code Sight site-option tokens*.

p) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

q) Ensure to set the token *SELINUX_SETUP=true*.

r) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

s) Save the `site-options.conf` file.

9. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

10. Set up the initial site data (bootstrap).

```
./bootstrap-data.sh
```

11. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

12. ☞ **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

13. If you have installed Black Duck Code Sight, then install the license for Black Duck Code Sight. For more information, see *Install the Black Duck Code Sight license* on page 736.

14. Restart the `collabnet` services.

```
/etc/init.d/collabnet restart
```

15. Apply some finishing touches and make sure everything is running smoothly.

a) Reboot the server and make sure all services come up automatically at startup.

b) Log into your site as the administrator.

The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.

   c) Create a sample project.

     See *Create a TeamForge project*.

   d) Write a welcome message to your site's users.

     See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Caching with Memcached**: *Install Memcached* on page 571

Install TeamForge 8.2: Database and Datamart on a separate server

In this option, we install the Database (Operational Database) and Datamart (Reporting database) on a separate server and other services on the main application server. If you want Black Duck Code Sight, you must install it on a separate RHEL/CentOS 6.7 server.

☞   **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- ETL Server
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer)

The following service runs on the database server. (We call this my.db.host)

- Database Server (Operational DB and Reports DB)

We call this a "dedicated install" since both the servers are dedicated to TeamForge. The TeamForge installer takes care of the database configurations. In this example, we will specify a separate port for the reports database. By default, both the site database and the reporting database use port 5432, but when heavy traffic is expected, it can be a good idea to use port 5632 for the reporting database.

☞   **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do this on the main TeamForge application server. We'll call this my.app.host.

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

   - Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
   - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
   - See *the Red Hat installation guide* for help.

   ☞   **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf,` and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
```

```
sf-admin soft nofile 65536
```
Save the file and reboot the server.

3. Check your basic networking setup.

   See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```

   b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

   ```
   setenforce 0
   ```

   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.
   b) Install the repository package.

   ```
   yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
   ```

   c) Refresh your repository cache.

   ```
   yum clean all
   ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

   • Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

      ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

   b) Unpack the disconnected installation package.

   ```
   rpm -ivh <package-name>
   ```

   c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

   ```
   rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
   ```

   d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

   Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

   In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

   e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

6. Install the following application packages.

   a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

   b) GIT: To install the GIT packages run the following command.

```
yum install teamforge-git
```

7. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

   a) Identify the servers and services running on them.

```
HOST_localhost=app etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.db.domain.com=database datamart
```

   b) Add 'gerrit' to the HOST_localhost token if you are installing Git.

```
HOST_localhost=app etl indexer subversion cvs gerrit
```

   c) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app etl indexer subversion cvs gerrit binary
```

   d) Configure the following token if you are installing Black Duck Code Sight.

```
HOST_my.codesight.domain.com=codesearch
```

   e) Configure the database and datamart settings.

☞ **Note:** For more information about configuring variables, see *site-options.conf* on page 775

```
DATABASE_TYPE=postgresql
```

```
DATABASE_USERNAME=ctfuser
```

```
DATABASE_NAME=ctfdb
```

```
DATABASE_READ_ONLY_USER=ctfrouser
```

```
REPORTS_DATABASE_USERNAME=ctfrptuser
```

```
REPORTS_DATABASE_NAME=ctfrptdb
```

```
REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
```

```
REPORTS_DATABASE_MAX_POOL_SIZE=30
```

☞ **Note:** The database name and username values are arbitrary alphanumeric strings.

f) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

g) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"`` `

   • To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
   • To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

h) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

i) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

j) Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

☞ **Important:** Remember to use the same key in the external SCM integration server also.

k) If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

`ADMIN_EMAIL=root@{__APPLICATION_HOST__}`

l) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

m) Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

n) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

o) Configure Black Duck Code Sight tokens if you are installing Black Duck Code Sight. See *Black Duck Code Sight site-option tokens*.

p) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

q) Ensure to set the token *SELINUX_SETUP=true*.

r) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

s) Save the `site-options.conf` file.

8. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the database server - my.db.host

9. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

- Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
- The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
- See *the Red Hat installation guide* for help.

☞ **Important:** Don't customize your installation. Select only the default packages list.

10. Check your basic networking setup.

See *Set up networking for your TeamForge server* on page 372 for details.

11. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

12. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

13. Install the TeamForge database packages.

```
yum install teamforge-database
```

14. Copy the site-options.conf file from the my.app.host to the database server in the directory /opt/collabnet/teamforge-installer/8.2.0.0/conf

15. Modify the host token settings on the site-options.conf file.

☞ **Important:** If you choose not to use the application server's `site-options.conf` file, then don't forget to copy the value of AUTO_DATA token from the application server.

- ```
  HOST_my.db.host=database datamart
  ```

  ☞ **Note:** 'HOST_my.db.host' is just an example. As you are installing database on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

- ```
  DOMAIN_my.db.host=my.db.domain.com
  ```

- ```
  HOST_my.app.domain.com=app etl indexer subversion cvs gerrit
  ```

16. Recreate the runtime environment.

    ```
    cd /opt/collabnet/teamforge-installer/8.2.0.0
    ./install.sh -r -I -V
    ```

Do the following on the Black Duck Code Sight server - my.codesight.host.

17. Set up Black Duck Code Sight on a separate server. See *Install Black Duck Code Sight on a separate RHEL/ CentOS 6.7 server*.

Do the following on the application server - my.app.host

18. Set up the initial site data (bootstrap).

    ```
    cd /opt/collabnet/teamforge-installer/8.2.0.0
    ./bootstrap-data.sh
    ```

19. Start TeamForge.

    ```
    /etc/init.d/collabnet start
    ```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

20. ☞ **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

    Run the TeamForge post installation script. For more information, see *post-install.py*.

    ```
    /opt/collabnet/teamforge/runtime/scripts/post-install.py
    ```

    ☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

21. Revoke the super user permissions of database and datamart users.

    ```
    /opt/collabnet/teamforge/runtime/scripts/revoke-superuser-permission.py
    ```

22. Restart the `collabnet` services.

    ```
    /etc/init.d/collabnet restart
    ```

23. Apply some finishing touches and make sure everything is running smoothly.
    a) Reboot the server and make sure all services come up automatically at startup.
    b) Log into your site as the administrator.

       The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.
    c) Create a sample project.

       See *Create a TeamForge project*.
    d) Write a welcome message to your site's users.

       See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.

- **Caching with Memcached**: *Install Memcached* on page 571

Install TeamForge 8.2: Reporting services on a separate server

In this option, we install the datamart (Reporting database) and ETL on a separate server and other services on the main application server. If you want Black Duck Code Sight, you must install it on a separate RHEL/CentOS 6.7 server.

☞ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Database Server (Operational DB)
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer).

The following service runs on the database server. (We call this my.reports.host)

- Database Server (Reports DB)
- ETL Server

Log on to the server as root user always.

Do this on the main TeamForge application server. We'll call this my.app.host.

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

    - Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
    - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
    - See *the Red Hat installation guide* for help.

    ☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, /etc/security/limits.d/teamforge.conf, and make sure it has the following "nproc" and "nofile" settings.

    ```
    vim /etc/security/limits.d/teamforge.conf
    sf-admin soft nproc 10240
    sf-admin hard nproc 10240
    sf-admin hard nofile 65536
    sf-admin soft nofile 65536
    ```
    Save the file and reboot the server.

3. Check your basic networking setup.

    See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

    a) Verify if SELinux is running in enforcing mode.

    ```
    getenforce
    ```

    b) If the output of the getenforce command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

    ```
    setenforce 0
    ```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. Configure your TeamForge installation repository.

> **TeamForge installation repository configuration for sites with internet access**

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to /tmp.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

> **TeamForge installation repository configuration for sites without internet access**

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

  • Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

    ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

6. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

b) GIT: To install the GIT packages run the following command.

```
yum install teamforge-git
```

7. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Identify the servers and services running on them.

```
HOST_localhost=app database indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.reports.domain.com=datamart etl
```

b) Add 'gerrit' to the HOST_localhost token if you are installing Git.

```
HOST_localhost=app database indexer subversion cvs  gerrit
```

c) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database indexer subversion cvs gerrit binary
```

d) Configure the following token if you are installing Black Duck Code Sight.

```
HOST_my.codesight.domain.com=codesearch
```

e) Configure the database and datamart settings.

> ☞ **Note:** For more information about configuring variables, see *site-options.conf* on page 775

```
DATABASE_TYPE=postgresql
```

```
DATABASE_USERNAME=ctfuser
```

```
DATABASE_NAME=ctfdb
```

```
DATABASE_READ_ONLY_USER=ctfrouser
```

```
REPORTS_DATABASE_USERNAME=ctfrptuser
```

```
REPORTS_DATABASE_NAME=ctfrptdb
```

```
REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
```

```
REPORTS_DATABASE_MAX_POOL_SIZE=30
```

> ☞ **Note:** The database name and user name values are arbitrary alphanumeric strings.

f) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

g) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

h) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

i) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

j) Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

☞ **Important:** Remember to use the same key in the external SCM integration server also.

k) If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

```
ADMIN_EMAIL=root@{__APPLICATION_HOST__}
```

l) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

m) Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

n) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

o) Configure Black Duck Code Sight tokens if you are installing Black Duck Code Sight. See *Black Duck Code Sight site-option tokens*.

p) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

q) Ensure to set the token *SELINUX_SETUP=true*.

r) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

s) Save the `site-options.conf` file.

8. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the reporting server - my.reports.host

9. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

- Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
- The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
- See *the Red Hat installation guide* for help.

 ☞ **Important:** Don't customize your installation. Select only the default packages list.

10. Check your basic networking setup.

See *Set up networking for your TeamForge server* on page 372 for details.

11. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

12. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

13. Run the following command to install the Reporting packages.

```
yum install teamforge-database teamforge-etl
```

14. Copy the `site-options.conf` file from the application server to the reporting server in the directory `/opt/collabnet/teamforge-installer/8.2.0.0/conf`

15. Modify the host token settings on the `site-options.conf` file.

☞ **Important:** If you choose not to use the application server's `site-options.conf` file, then don't forget to copy the value of AUTO_DATA token from the application server.

```
HOST_my.reports.host=datamart etl
```

☞ **Note:** 'HOST_my.reports.host' is just an example. As you are installing reporting services on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.reports.host=my.reports.domain.com
```

```
HOST_my.app.domain.com=app database indexer subversion cvs gerrit
```

16. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do the following on the Black Duck Code Sight server - my.codesight.host.

17. Set up Black Duck Code Sight on a separate server. See *Install Black Duck Code Sight on a separate RHEL/CentOS 6.7 server*.

Do the following on the application server - my.app.host

18. Set up the initial site data (bootstrap).

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./bootstrap-data.sh
```

19. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

Do this on the reporting server - my.reports.host

20. Start TeamForge.

```
/etc/init.d/collabnet start
```

Do the following on the application server - my.app.host

21. ☞ **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

22. Revoke the super user permissions of database and datamart users.

```
/opt/collabnet/teamforge/runtime/scripts/revoke-superuser-permission.py
```

23. Restart the `collabnet` services.

```
/etc/init.d/collabnet restart
```

24. Apply some finishing touches and make sure everything is running smoothly.

   a) Reboot the server and make sure all services come up automatically at startup.

   b) Log into your site as the administrator.

   The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.

   c) Create a sample project.

   See *Create a TeamForge project*.

   d) Write a welcome message to your site's users.

   See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Caching with Memcached**: *Install Memcached* on page 571

Install TeamForge 8.2: Indexer on a separate server

In this option, we install the indexer on a separate server and all other services on the main application server. If you want Black Duck Code Sight, you must install it on a separate RHEL/CentOS 6.7 server.

☞ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Database Server (Operational DB and Reports DB)
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- ETL Server

The following service runs on the database server. (We call this my.indexer.host)

- Search Server (Indexer)

Log on to the server as root user always.

Do this on the main TeamForge application server. We'll call this my.app.host.

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

   - Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
   - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
   - See *the Red Hat installation guide* for help.

   ☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

3. Check your basic networking setup.

   See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

      ```
      getenforce
      ```
   b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

      ```
      setenforce 0
      ```

   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.
   b) Install the repository package.

      ```
      yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
      ```
   c) Refresh your repository cache.

      ```
      yum clean all
      ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

      - Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

        ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

   b) Unpack the disconnected installation package.

      ```
      rpm -ivh <package-name>
      ```
   c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

6. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

b) GIT: To install the GIT packages run the following command.

```
yum install teamforge-git
```

7. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Identify the servers and services running on them.

```
HOST_localhost=app database datamart etl subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.indexer.domain.com=indexer
```

b) Add 'gerrit' to the HOST_localhost token if you are installing Git.

```
HOST_localhost=app database datanart etl subversion cvs  gerrit
```

c) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl subversion cvs gerrit binary
```

d) Configure the following token if you are installing Black Duck Code Sight.

```
HOST_my.codesight.domain.com=codesearch
```

e) Configure the database and datamart settings.

☞ **Note:** For more information about configuring variables, see *site-options.conf* on page 775

```
DATABASE_TYPE=postgresql
```

```
DATABASE_USERNAME=ctfuser
```

```
DATABASE_NAME=ctfdb
```

```
DATABASE_READ_ONLY_USER=ctfrouser
```

```
REPORTS_DATABASE_USERNAME=ctfrptuser
```

```
REPORTS_DATABASE_NAME=ctfrptdb
```

```
REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
```

```
REPORTS_DATABASE_MAX_POOL_SIZE=30
```

☞ **Note:** The database name and user name values are arbitrary alphanumeric strings.

f) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

g) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

h) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

i) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes

or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

j) Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

> ☞ **Important:** Remember to use the same key in the external SCM integration server also.

k) If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

   `ADMIN_EMAIL=root@{__APPLICATION_HOST__}`

l) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

m) Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

n) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

o) Configure Black Duck Code Sight tokens if you are installing Black Duck Code Sight. See *Black Duck Code Sight site-option tokens*.

p) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

q) Ensure to set the token *SELINUX_SETUP=true*.

r) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

s) Save the `site-options.conf` file.

8. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the search server - my.indexer.host

9. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

- Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
- The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
- See *the Red Hat installation guide* for help.

> ☞ **Important:** Don't customize your installation. Select only the default packages list.

10. Check your basic networking setup.

See *Set up networking for your TeamForge server* on page 372 for details.

11. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

   `getenforce`

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

   `setenforce 0`

12. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

   • Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

   ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

13. Run the following command to install the indexer packages.

```
yum install teamforge-indexer
```

14. Copy the `site-options.conf` file from the application server to the reporting server in the directory `/opt/collabnet/teamforge-installer/8.2.0.0/conf`

15. Modify the host token settings on the `site-options.conf` file.

☞ **Important:** If you choose not to use the application server's `site-options.conf` file, then don't forget to copy the value of AUTO_DATA token from the application server.

```
HOST_my.indexer.host=indexer
```

☞ **Note:** 'HOST_my.reports.host' is just an example. As you are installing reporting services on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.indexer.host=my.indexer.domain.com
```

```
HOST_my.app.domain.com=app database datamart etl subversion cvs gerrit
 binary
```

16. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do the following on the Black Duck Code Sight server - my.codesight.host.

17. Set up Black Duck Code Sight on a separate server. See *Install Black Duck Code Sight on a separate RHEL/CentOS 6.7 server*.

Do the following on the application server - my.app.host

18. Set up the initial site data (bootstrap).

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./bootstrap-data.sh
```

19. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

Do this on the search server - my.indexer.host

20. Start TeamForge.

```
/etc/init.d/collabnet start
```

Do the following on the application server - my.app.host

21. ☞ **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

22. Restart the `collabnet` services.

```
/etc/init.d/collabnet restart
```

23. Apply some finishing touches and make sure everything is running smoothly.
   a) Reboot the server and make sure all services come up automatically at startup.
   b) Log into your site as the administrator.

   The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.
   c) Create a sample project.

   See *Create a TeamForge project*.
   d) Write a welcome message to your site's users.

See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Caching with Memcached**: *Install Memcached* on page 571

Install TeamForge 8.2: Black Duck Code Sight on a separate server on RHEL/CentOS

In this option, we install Black Duck Code Sight on a separate server and other services on the main application server.

☞ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Database Server (Operational DB and Reports DB)
- ETL Server
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer).

☞ **Important:** Black Duck Code Sight is not supported on RHEL/CentOS 7.2. CollabNet recommends that you install Black Duck Code Sight on a separate RHEL/CentOS 6.7 server.

The following service runs on the code sight server (we call this my.codesight.host).

- Black Duck Code Sight Server

Log on to the server as root user always.

Do this on the main TeamForge application server. We'll call this my.app.host.

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

   - Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
   - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
   - See *the Red Hat installation guide* for help.

   ☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

3. Check your basic networking setup.

   See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
```

```
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

6. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

b) GIT: To install the GIT packages run the following command.

```
yum install teamforge-git
```

7. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Configure the HOST token.

-
  ```
  HOST_localhost=app database datamart etl indexer subversion cvs
  ```

-
  ```
  DOMAIN_localhost=my.app.domain.com
  ```

-
  ```
  HOST_my.codesight.domain.com=codesearch
  ```

b) Add 'gerrit' to the HOST_localhost token if you are installing Git.

```
HOST_localhost=app database datamart etl indexer subversion cvs gerrit
```

c) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs
 gerrit binary
```

d) Configure the database and datamart settings.

> ☞ **Note:** For more information about configuring variables, see *site-options.conf* on page 775

-
  ```
  DATABASE_TYPE=postgresql
  ```

-
  ```
  DATABASE_USERNAME=ctfuser
  ```

-
  ```
  DATABASE_NAME=ctfdb
  ```

-
  ```
  DATABASE_READ_ONLY_USER=ctfrouser
  ```

-
  ```
  REPORTS_DATABASE_USERNAME=ctfrptuser
  ```

-
  ```
  REPORTS_DATABASE_NAME=ctfrptdb
  ```

-
  ```
  REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
  ```

-
  ```
  REPORTS_DATABASE_MAX_POOL_SIZE=30
  ```

> ☞ **Note:** The database name and username values are arbitrary alphanumeric strings.

e) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

f) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

g) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

h) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

i) Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

☞ **Important:** Remember to use the same key in the external SCM integration server also.

j) If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

```
ADMIN_EMAIL=root@{__APPLICATION_HOST__}
```

k) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

l) Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

m) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

n) Configure Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

o) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

p) Ensure to set the token *SELINUX_SETUP=true*.

q) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

r) Save the `site-options.conf` file.

8. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the Black Duck Code Sight Server. We'll call this my.codesight.host.

9. Install Red Hat Enterprise Linux/CentOS 6.7 and log in as root.

   - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
   - See *the Red Hat installation guide* for help.

   ☞ **Important:** Don't customize your installation. Select only the default packages list.

10. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

    a) Verify if SELinux is running in enforcing mode.

       ```
       getenforce
       ```
    b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.
    c) If not disabled, run the following command to disable SELinux.

       ```
       setenforce 0
       ```

11. Configure your TeamForge installation repository.

    TeamForge installation repository configuration for sites with internet access

    a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.
    b) Install the repository package.

       ```
       yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
       ```
    c) Refresh your repository cache.

       ```
       yum clean all
       ```

    TeamForge installation repository configuration for sites without internet access

    a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

       - Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**
    b) Unpack the disconnected installation package.

       ```
       rpm -ivh <package-name>
       ```
    c) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

       Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

       In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

       ```
       cd /media/
       mkdir cdrom
       mount /dev/cdrom ./cdrom/
       ```

       If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.
    d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

       ```
       vi /etc/yum.repos.d/cdrom.repo
       ```

       Here's a sample yum configuration file.

       ```
       [RHEL-CDROM]
       name=RHEL CDRom
       baseurl=file:///media/cdrom/Server/
       ```

```
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

12. Run the following command to install the Black Duck Code Sight packages.

```
yum install teamforge-codesearch
```

13. Copy the `site-options.conf` file from the application server to the `/opt/collabnet/teamforge-installer/8.2.0.0/conf` directory of the Black Duck Code Sight server.

14. Modify the host token settings on the `site-options.conf` file.

> ☞ **Important:** If you choose not to use the application server's `site-options.conf` file, then don't forget to copy the value of AUTO_DATA token from the application server.

-
  ```
  HOST_my.codesight.host=codesearch
  ```

-
  ```
  DOMAIN_my.codesight.host=my.codesight.domain.com
  ```

-
  ```
  HOST_my.app.domain.com=app database datamart etl indexer subversion cvs
   gerrit
  ```

15. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do the following on the application server - my.app.host

16. Set up the initial site data (bootstrap).

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./bootstrap-data.sh
```

17. Start TeamForge.

```
/etc/init.d/collabnet start
```

> ☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

18. ☞ **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

> ☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

19. Revoke the user permissions of the database and datamart users.

```
/opt/collabnet/teamforge/runtime/scripts/revoke-superuser-permission.py
```

Do this on my.codesight.host

20. Install the Black Duck Code Sight license on the server where Black Duck Code Sight is installed. For more information, see *Install the Black Duck Code Sight license* on page 736.

21. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

22. Restart the Black Duck Code Sight service.

```
/etc/init.d/collabnet restart tomcatcs
```

Do this on my.app.host

23. Restart the collabnet services.

```
/etc/init.d/collabnet restart
```

24. Apply some finishing touches and make sure everything is running smoothly.

   a) Reboot the server and make sure all services come up automatically at startup.

   b) Log into your site as the administrator.

      The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.

   c) Create a sample project.

      See *Create a TeamForge project*.

   d) Write a welcome message to your site's users.

      See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Caching with Memcached**: *Install Memcached* on page 571

Install TeamForge 8.2: SCM and Git integration on a separate server

In this option, we install SCM (Subversion, CVS) and GIT integrations on a separate server and other services on the main application server. If you want Black Duck Code Sight, you must install it on a separate RHEL/CentOS 6.7 server.

> ☛ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Database Server (Operational DB and Reports DB)
- ETL Server
- Search Server (Indexer).

The following service runs on the SCM server. (We call this my.scmandgit.host)

- SCM Integration Server (Subversion and CVS)
- GIT Integration Server

> ☛ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do this on the main TeamForge application server. We'll call this my.app.host.

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

   - Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
   - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.

- See *the Red Hat installation guide* for help.

☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```
Save the file and reboot the server.

3. Check your basic networking setup.

See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

    ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

6. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

7. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Identify the servers and services running on them.

```
HOST_localhost=app database datamart etl indexer
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.scmandgit.domain.com=subversion cvs gerrit
```

b) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer binary
```

c) Configure the following token if you are installing Black Duck Code Sight.

```
HOST_my.codesight.domain.com=codesearch
```

d) Configure the database and datamart settings.

☞ **Note:** For more information about configuring variables, see *site-options.conf* on page 775

- ```
  DATABASE_TYPE=postgresql
  ```

- ```
  DATABASE_USERNAME=ctfuser
  ```

- ```
  DATABASE_NAME=ctfdb
  ```

- ```
  DATABASE_READ_ONLY_USER=ctfrouser
  ```

- ```
  REPORTS_DATABASE_USERNAME=ctfrptuser
  ```

- ```
  REPORTS_DATABASE_NAME=ctfrptdb
  ```

- ```
  REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
  ```

- ```
  REPORTS_DATABASE_MAX_POOL_SIZE=30
  ```

☞ **Note:** The database name and username values are arbitrary alphanumeric strings.

e) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

f) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

g) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

h) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

i) Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

☞ **Important:** Remember to use the same key in the external SCM integration server also.

j) If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

```
ADMIN_EMAIL=root@{__APPLICATION_HOST__}
```

k) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

l) Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

m) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

n) Configure Black Duck Code Sight tokens if you are installing Black Duck Code Sight. See *Black Duck Code Sight site-option tokens*.

o) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

p) Ensure to set the token *SELINUX_SETUP=true*.

q) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

r) ☞ **Important:** This step is required if you want Git notification emails.

Update the *JAMES_ACCEPTED_RELAYS* site-options token with the Git server's IP address. See *JAMES_ACCEPTED_RELAYS* on page 794 for more information.

```
JAMES_ACCEPTED_RELAYS=127.0.0.1,{__CEE_DOMAIN__},<The IP address of the
  Git server>
```

s) Save the `site-options.conf` file.

8. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the SCM Server - my.scmandgit.host

9. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

- Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
- The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
- See *the Red Hat installation guide* for help.

☞ **Important:** Don't customize your installation. Select only the default packages list.

10. Check your basic networking setup.

See *Set up networking for your TeamForge server* on page 372 for details.

11. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

12. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

-   Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

    ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

13. Install the TeamForge SCM and Git packages.

```
yum install teamforge-scm teamforge-git
```

14. Copy the `site-options.conf` file from the application server to the SCM server in the directory `/opt/collabnet/teamforge-installer/8.2.0.0/conf`

15. Modify the host token settings on the `site-options.conf` file.

> ☞ **Important:** If you choose not to use the application server's `site-options.conf` file, then don't forget to copy the value of AUTO_DATA token from the application server.

```
HOST_my.scmandgit.host=subversion cvs gerrit
```

```
DOMAIN_my.scmandgit.host=my.scmandgit.domain.com
```

```
HOST_my.app.domain.com=app database datamart etl indexer
```

16. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do the following on the Black Duck Code Sight server - my.codesight.host.

17. Set up Black Duck Code Sight on a separate server. See *Install Black Duck Code Sight on a separate RHEL/CentOS 6.7 server*.

Do the following on the application server - my.app.host

18. Set up the initial site data (bootstrap).

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./bootstrap-data.sh
```

19. Start TeamForge.

```
/etc/init.d/collabnet start
```

20. ☞ **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

> ☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

21. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do this on the SCM server - my.scmandgit.host

22. Start TeamForge.

```
/etc/init.d/collabnet start
```

> ☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

23. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

24. ☞ **Important:** This step is required if you want Git notification emails.

Edit the `/opt/collabnet/gerrit/etc/gerrit.config` file and update the 'smtpServer' property with the TeamForge application server's host name.

For example:

```
[sendemail]
              smtpServer = <TeamForge application server host name>
```

a) Restart gerrit.

```
/etc/init.d/collabnet stop gerrit
/etc/init.d/collabnet start gerrit
```

Do the following on the application server - my.app.host

25. Restart the `collabnet` services.

```
/etc/init.d/collabnet restart
```

26. Apply some finishing touches and make sure everything is running smoothly.

a) Reboot the server and make sure all services come up automatically at startup.

b) Log into your site as the administrator.

The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.

c) Create a sample project.

See *Create a TeamForge project*.

d) Write a welcome message to your site's users.

See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Caching with Memcached**: *Install Memcached* on page 571

Install TeamForge 8.2: Database and SCM on separate servers

In this option, we install the Database (Operational Database) and Datamart (Reporting Database) on the same server; SCM (Subversion and CVS) and Git on the second server, and other services on the application server. If you want Black Duck Code Sight, you must install it on a separate RHEL/CentOS 6.7 server.

☞ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- ETL Server
- Search Server (Indexer).

The following service runs on the database server. (We call this my.db.host)

- Database Server (Operational DB and Reports DB)

The following services run on the SCM server. (We call this my.scmandgit.host)

- SCM Integration Server (Subversion and CVS)
- GIT Integration Server

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do this on the main TeamForge application server. We'll call this my.app.host.

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

   - Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
   - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
   - See *the Red Hat installation guide* for help.

   ☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

3. Check your basic networking setup.

   See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```
   b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

   ```
   setenforce 0
   ```
   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.
   b) Install the repository package.

   ```
   yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
   ```
   c) Refresh your repository cache.

   ```
   yum clean all
   ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

   - Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

     ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

   b) Unpack the disconnected installation package.

   ```
   rpm -ivh <package-name>
   ```
   c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

6. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

7. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Identify the servers and services running on them.

```
HOST_localhost=app etl indexer
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.db.domain.com=database datamart
```

```
HOST_my.scmandgit.domain.com=subversion cvs gerrit
```

b) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app etl indexer binary
```

c) Configure the following token if you are installing Black Duck Code Sight.

```
HOST_my.codesight.domain.com=codesearch
```

d) Configure the database and datamart settings.

☞ **Note:** For more information about configuring variables, see *site-options.conf* on page 775

```
DATABASE_TYPE=postgresql
```

```
DATABASE_USERNAME=ctfuser
```

```
DATABASE_NAME=ctfdb
```

```
DATABASE_READ_ONLY_USER=ctfrouser
```

```
REPORTS_DATABASE_USERNAME=ctfrptuser
```

```
REPORTS_DATABASE_NAME=ctfrptdb
```

```
REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
```

```
REPORTS_DATABASE_MAX_POOL_SIZE=30
```

☞ **Note:** The database name and username values are arbitrary alphanumeric strings.

e) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

f) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

g) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

h) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

i) Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

☞ **Important:** Remember to use the same key in the external SCM integration server also.

j) If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

`ADMIN_EMAIL=root@{__APPLICATION_HOST__}`

k) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

l) Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

m) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

n) Configure Black Duck Code Sight tokens if you are installing Black Duck Code Sight. See *Black Duck Code Sight site-option tokens*.

o) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

p) Ensure to set the token *SELINUX_SETUP=true*.

q) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

r) ☞ **Important:** This step is required if you want Git notification emails.

Update the *JAMES_ACCEPTED_RELAYS* site-options token with the Git server's IP address. See *JAMES_ACCEPTED_RELAYS* on page 794 for more information.

```
JAMES_ACCEPTED_RELAYS=127.0.0.1,{__CEE_DOMAIN__},<The IP address of the
 Git server>
```

s) Save the `site-options.conf` file.

8. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the database server - my.db.host

9. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

- Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
- The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
- See *the Red Hat installation guide* for help.

☞ **Important:** Don't customize your installation. Select only the default packages list.

10. Check your basic networking setup.

See *Set up networking for your TeamForge server* on page 372 for details.

11. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

12. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

13. Install the TeamForge database packages.

```
yum install teamforge-database
```

14. Copy the `site-options.conf` file from the application server to the database server in the directory `/opt/collabnet/teamforge-installer/8.2.0.0/conf`

15. Modify the host token settings on the `site-options.conf` file.

☞ **Important:** If you choose not to use the application server's `site-options.conf` file, then don't forget to copy the value of AUTO_DATA token from the application server.

```
HOST_my.db.host=database datamart
```

☞ **Note:** 'HOST_my.db.host' is just an example. As you are installing database on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.db.host=my.db.domain.com
```

```
HOST_my.app.domain.com=app etl indexer
```

```
HOST_my.scmandgit.domain.com=subversion cvs gerrit
```

16. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the SCM Server - my.scmandgit.host

17. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

- Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
- The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
- See *the Red Hat installation guide* for help.

☞ **Important:** Don't customize your installation. Select only the default packages list.

18. Check your basic networking setup.

See *Set up networking for your TeamForge server* on page 372 for details.

19. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

20. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

21. Install the TeamForge SCM and Git packages.

```
yum install teamforge-scm teamforge-git
```

22. Copy the `site-options.conf` file from the application server to the SCM server in the directory `/opt/collabnet/teamforge-installer/8.2.0.0/conf`

23. Modify the host token settings on the `site-options.conf` file.

☞ **Important:** If you choose not to use the application server's `site-options.conf` file, then don't forget to copy the value of AUTO_DATA token from the application server.

```
HOST_my.scmandgit.host=subversion cvs gerrit
```

☞ **Note:** 'HOST_my.scmandgit.host' is just an example. As you are installing SCM on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.scmandgit.host=my.scmandgit.domain.com
```

```
HOST_my.app.domain.com=app etl indexer
```

```
HOST_my.db.domain.com=database datamart
```

24. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

25. Set up the initial site data (bootstrap).

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./bootstrap-data.sh
```

Do the following on the Black Duck Code Sight server - my.codesight.host.

26. Set up Black Duck Code Sight on a separate server. See *Install Black Duck Code Sight on a separate RHEL/ CentOS 6.7 server*.

Do the following on the application server - my.app.host

27. Set up the initial site data (bootstrap).

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./bootstrap-data.sh
```

28. Start TeamForge.

```
/etc/init.d/collabnet start
```

29. ☞ **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

30. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do this on the SCM server - my.scmandgit.host

31. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

32. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

33. ☞ **Important:** This step is required if you want Git notification emails.

Edit the `/opt/collabnet/gerrit/etc/gerrit.config` file and update the 'smtpServer' property with the TeamForge application server's host name.

For example:

```
[sendemail]
                smtpServer = <TeamForge application server host name>
```

a) Restart gerrit.

```
/etc/init.d/collabnet stop gerrit
/etc/init.d/collabnet start gerrit
```

Do the following on the application server - my.app.host

34. Revoke the super user permissions of database and datamart users.

```
/opt/collabnet/teamforge/runtime/scripts/revoke-superuser-permission.py
```

35. Restart the `collabnet` services.

```
/etc/init.d/collabnet restart
```

36. Apply some finishing touches and make sure everything is running smoothly.

   a) Reboot the server and make sure all services come up automatically at startup.
   b) Log into your site as the administrator.

      The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.
   c) Create a sample project.

      See *Create a TeamForge project*.
   d) Write a welcome message to your site's users.

      See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Caching with Memcached**: *Install Memcached* on page 571

Install TeamForge 8.2: Oracle database on a separate server

In this option, we install the Oracle database (Operational database and Reports database) on a separate server and other services on the main application server. If you want Black Duck Code Sight, you must install it on a separate RHEL/CentOS 6.7 server.

☞ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- ETL Server
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer).

The following service runs on the database server. (We call this my.db.host)

- Database Server (Operational DB and Reports DB)

☞ **Note:** If either of the remote servers (the data server or the source code server) is not under your direct control, check with the Database Administrator to make sure that you can carry out these instructions on that server.

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do this on the TeamForge application server. We'll call this my.app.host.

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

   • Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
   • The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
   • See *the Red Hat installation guide* for help.

   ☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

3. Check your basic networking setup.

   See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

      ```
      getenforce
      ```
   b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

      ```
      setenforce 0
      ```

   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.
   b) Install the repository package.

      ```
      yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
      ```
   c) Refresh your repository cache.

      ```
      yum clean all
      ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

      • Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

6. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

b) GIT: To install the GIT packages run the following command.

```
yum install teamforge-git
```

7. Rename the sample site configuration file from the installation package.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
cp conf/site-options-oracle.conf conf/site-options.conf
```

☞ **Note:** The files `site-options-small.conf`, `site-options-medium.conf` and `site-options-large.conf` contain options to tune the performance of the TeamForge site. To tune your site's performance, you can look through these files for the load specifications they are intended for, and use the appropriate one for your site's requirements.

8. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Configure the HOST token.

```
HOST_localhost=app etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_<my.db.host>=database datamart
```

b) Add 'gerrit' to the HOST_localhost token if you are installing GIT.

```
HOST_localhost=app etl indexer subversion cvs gerrit
```

c) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app etl indexer subversion cvs gerrit binary
```

d) Configure the following token if you are installing Black Duck Code Sight.

```
HOST_my.codesight.domain.com=codesearch
```

e) Configure the database and datamart settings.

☞ **Note:** For more information about configuring variables, see *site-options.conf* on page 775

```
DATABASE_TYPE=oracle
```

```
DATABASE_USERNAME=sitedatabaseusername
```

```
DATABASE_PASSWORD=sitedatabasepwd
```

```
DATABASE_READ_ONLY_USER=sitedatabasereadonlyusername
```

```
DATABASE_READ_ONLY_PASSWORD=sitedatabasereadonlyuserpwd
```

```
DATABASE_NAME=sitedatabaseinstancename
```

```
REPORTS_DATABASE_USERNAME=reportingdatabaseusername
```

```
REPORTS_DATABASE_PASSWORD=reportingdatabasepwd
```

```
REPORTS_DATABASE_NAME=reportingdatabaseinstancename
```

```
REPORTS_DATABASE_READ_ONLY_USER=reportingreadonlyusername
```

```
REPORTS_DATABASE_READ_ONLY_PASSWORD=reportingreadonlyuserpwd
```

```
REPORTS_DATABASE_MAX_POOL_SIZE=30
```

```
DATABASE_SERVICE_NAME
```

```
REPORTS_DATABASE_SERVICE_NAME
```

☞ **Tip:** To find the value for the token *DATABASE_SERVICE_NAME* log in to your Oracle server and execute this command.

```
su - oracle
tnsping <database_name>
```

Find the value of the `SERVICE NAME` in the output and use this value for the *DATABASE_SERVICE_NAME* in the `site-options.conf` file.

f) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: $<>/\'" `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

g) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

h) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

i) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

j) Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

☞ **Important:** Remember to use the same key in the external SCM integration server also.

k) If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

```
ADMIN_EMAIL=root@{__APPLICATION_HOST__}
```

l) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

m) Ensure to set the token DEDICATED_INSTALL=true. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

n) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

o) Configure Black Duck Code Sight tokens if you are installing Black Duck Code Sight. See *Black Duck Code Sight site-option tokens*.

p) Ensure to set the token, *SELINUX_SETUP=true*.

q) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

r) Save the `site-options.conf` file.

9. Download the corresponding version of Oracle client from *http://www.oracle.com/technetwork/database/features/instant-client/index-097480.html* and run the following command:

```
yum localinstall <path to oracle client rpm>
```

Do this on the database server - my.db.host

10. Log on to the Oracle database server as a system administrator with 'SYSDG' privilege and run the following query.

```
alter system set parallel_threads_per_cpu=4;
```

Do this on the TeamForge application server - my.app.host

11. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

12. Copy the Oracle datamart setup script from `/opt/collabnet/teamforge/runtime/scripts` to the `/tmp` directory of `my.db.host`.

```
scp /opt/collabnet/teamforge/runtime/scripts/datamart-oracle-setup.sh
<username>@<my.db.host>:/tmp
```

Do this on the database server - my.db.host

13. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

- Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
- The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
- See *the Red Hat installation guide* for help.

☞ **Important:** Don't customize your installation. Select only the default packages list.

14. Install Oracle 12c.

☞ **Note:** Make sure your database uses UTF8 or AL32UTF8 encoding. This is needed to support users in Asian languages. See *this Oracle knowledge base article.*

15. Copy the Oracle datamart setup script.

```
mkdir /u1
cp /tmp/datamart-oracle-setup.sh /u1
```

16. Log in as Oracle user and create the site database user and permissions.

See *Set up an Oracle database* on page 567 for help.

17. Create the reporting user and schema.

☞ **Note:** Skip this step if you have already set up the datamart setup in the Oracle database. Your responses to the script's prompts must match the values of the equivalent variables in the `site-options.conf` file on `my.app.server`.

```
cd /u1
sh datamart-oracle-setup.sh
```

Do the following on the Black Duck Code Sight server - my.codesight.host.

18. Set up Black Duck Code Sight on a separate server. See *Install Black Duck Code Sight on a separate RHEL/ CentOS 6.7 server*.

Do this on the TeamForge Application Server - my.app.host

19. Set up the initial site data (bootstrap).

```
./bootstrap-data.sh
```

20. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

21. ☞ **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

Run the TeamForge post installation script. For more information, see `post-install.py`.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

22. Restart the `collabnet` services.

```
/etc/init.d/collabnet restart
```

23. Apply some finishing touches and make sure everything is running smoothly.
    a) Reboot the server and make sure all services come up automatically at startup.
    b) Log into your site as the administrator.

       The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.
    c) Create a sample project.

       See *Create a TeamForge project*.
    d) Write a welcome message to your site's users.

       See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Caching with Memcached**: *Install Memcached* on page 571
- **Git**: To install Git on a separate server, see *Install Git on a separate server* on page 431.

Install Git on a separate server

In this option, we install the GIT integration services on a separate server.

☞ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

In this option, the following service runs on the Git server (we call this my.git.host).

- GIT Integration Server

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do this on the Git server. We'll call this my.git.host.

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

    - Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
    - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
    - See *the Red Hat installation guide* for help.

    ☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Check your basic networking setup.

    See *Set up networking for your TeamForge server* on page 372 for details.

3. If the Git server has SELinux enabled, disable it temporarily while installing or upgrading Git.

    a) Verify if SELinux is running in enforcing mode.

        ```
        getenforce
        ```
    b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.
    c) If not disabled, run the following command to disable SELinux.

        ```
        setenforce 0
        ```

4. Configure your TeamForge installation repository.

    TeamForge installation repository configuration for sites with internet access

    a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.
    b) Install the repository package.

        ```
        yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
        ```
    c) Refresh your repository cache.

        ```
        yum clean all
        ```

    TeamForge installation repository configuration for sites without internet access

    a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

        - Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

            ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

    b) Unpack the disconnected installation package.

        ```
        rpm -ivh <package-name>
        ```
    c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

        ```
        rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
        ```
    d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

    Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

    In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

    ```
    cd /media/
    mkdir cdrom
    mount /dev/cdrom ./cdrom/
    ```

    If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

5. Install the Git packages.

```
yum install teamforge-git
```

6. Configure the token settings for Git in the `site-options.conf` file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Modify the host token settings.

☞ **Important:** If you choose not to use the application server's `site-options.conf` file, then don't forget to copy the value of AUTO_DATA token from the application server.

```
HOST_my.git.host=gerrit
```

```
DOMAIN_my.git.host=my.git.domain.com
```

```
HOST_my.app.domain.com=app database datamart etl indexer subversion cvs
 binary
```

b) Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

c) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

d) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

e) Remember to include the *SCM_DEFAULT_SHARED_SECRET* on page 810 token in the `site-options.conf` file and provide it with a value of 16-24 characters.

Remember to use the same key as used in the primary TeamForge application server.

    f) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

    g) Save the `site-options.conf` file.

7. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

8. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

Configuring James if Git is on a separate server

See *James configuration if Git is on a separate server* on page 572.

Installing TeamForge Orchestrate

To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.

Uninstall TeamForge 8.2

To remove TeamForge completely, use the YUM utility.

☞   **Important:** This procedure removes the TeamForge and all associated databases, including your site data. Be sure to back up any data you want to keep.

1. Stop TeamForge.

```
/etc/init.d/collabnet stop all
```

2. Run yum to remove TeamForge.

```
yum erase TeamForge-installer
```

For every box in a multi-box site, use the same steps to uninstall.

Verify your TeamForge 8.2 installation

Congratulations: you have just installed your TeamForge 8.2 site. Now you can apply some finishing touches and make sure everything is running smoothly.

1. Turn on SSL for your site by editing the relevant variables in the `site-options.conf` file.

See *Set up SSL for your TeamForge site* on page 581 for details.

☞   **Note:** If SSL is enabled for any box belonging to your site, it must be enabled for all of them.

2. Log into your site as the administrator.

The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.

3. Install your license key.

See *Supply your TeamForge license key from Teamforge user interface* on page 435.

4. To verify that the tracker initial load (triggered through `TrackerInitialJob`) completed successfully, run the following SQL from `[RUNTIME_DIR]/scripts/psql-reporting-wrapper` or by selecting the **Datamart** option on the **System Tools** > **Ad Hoc Database Query** page in the TeamForge web interface.

```
select * from etl_job where job_name ='tracker_initial_etl';
```

The status column should have a value of 1.

5. Install a project template.

TeamForge comes with a sample project template that showcases some of the platform's most interesting features. Site administrators and project managers can use this template to jump-start projects without a lot of manual setup steps. See *Install project templates manually* on page 566.

☞   **Note:** This procedure is recommended, but not required.

6. Create a sample project.

See *Create a TeamForge project*.

7. Write a welcome message to your site's users.

See *Create a site-wide broadcast*.

Now that you have successfully installed your TeamForge site in its basic configuration, you can use the instructions under *Maintain your TeamForge 8.2 site* on page 566 to help keep your site going.

Supply your TeamForge license key from Teamforge user interface

Your license key enables you to use CollabNet TeamForge for the period of your contract.

Your license key will only work for the IP address of the machine that your CollabNet TeamForge is running on, as specified in your order form.

☞ **Tip:** These steps are for installing your license key via the web interface. If you prefer, you can install it as a text file instead. See *Supply your CollabNet TeamForge license key as a text file* on page 435.

1. Locate the confirmation email you received from your CollabNet representative when you purchased your contract.

2. Log into your site as the site administrator.

☞ **Note:** The site administrator is different from the root user on the machine where the site is running.

3. Click **Admin** > **License Key**.
If you have entered a license before, the IP address and current licensed number of users on your site are listed on the **License Key** page. Verify that the IP address is the same as the one you entered in your order form.

4. Click **Enter License Key**.

5. Copy your new license key from the confirmation email and paste it into the **Enter License Key** field.
A license key string looks like this:
`25:supervillaininc:144.16.116.25.:302D02150080D7853DB3E5C6F67EABC65BD3AC17D4D35CB3Z002`

☞ **Tip:** Save this license key in case you need to reinstall CollabNet TeamForge.

6. Click **Save**.

7. Verify that the new value for **Licensed Number of Users** matches the total number of licensed users in your contract.

Supply your CollabNet TeamForge license key as a text file

Your license key enables you to use CollabNet TeamForge for the period of your contract.

Your license key will only work for the IP address of the machine that your CollabNet TeamForge is running on.

☞ **Important:** If you are upgrading from a site with a limited number of users to an enterprise-scale site, you must install your license key before starting CollabNet TeamForge . Otherwise, your site could be rendered inoperable.

1. Locate the confirmation email you received from your CollabNet representative when you purchased your contract.

2. Create a text file and copy-paste your license key from the confirmation email into it.

For example, if your organization has 80 users who will use only the source code management features and 100 users who need the full range of application lifecycle management features, your license key string may look like this:
`alm=100:SCM=80:supervillaininc:144.16.116.25.:302D02150080D7853DB3E5C6F67EABC65BD3AC17`

☞ **Tip:** Save this license key in case you need to reinstall CollabNet TeamForge.

3. Save the text file as `/opt/collabnet/teamforge/var/etc/sflicense.txt`

☞ **Tip:** Save your license key somewhere remote too, in case you need to reinstall CollabNet TeamForge and your `sflicense.txt` file is not accessible.

4. Make the license file usable by the application.

```
chmod 0664 /opt/collabnet/teamforge/var/etc/sflicense.txt
chown <APP_USER>:<APP_GROUP> /opt/collabnet/teamforge/var/etc/
sflicense.txt
```

☞ **Note:** Change the values of `<APP_USER>` and `<APP_GROUP>` with the values of *APP_USER* and *APP_GROUP* tokens respectively from the `/opt/collabnet/teamforge/runtime/conf/` `runtime-options.conf` file.

Install TeamForge on VMware Player or VMware ESXi

To get the functionality of CollabNet TeamForge with the ease of installation and maintenance that comes with VMware, run CollabNet TeamForge on VMware.

☞ **Note:** For the hardware and software required to run TeamForge 8.2, see *Hardware and software requirements for CollabNet TeamForge 8.2 on a virtual machine* on page 750.

Get TeamForge 8.2 for VMware Player or VMware ESXi

Download the CollabNet TeamForge installer from CollabNet and unzip it on the server that will host your TeamForge site.

Install TeamForge on VMware Player

The VMware installation is ideal for small number of TeamForge user installations due to its detailed installation process. You have to download the CollabNet TeamForge installer for VMware Player and unzip it on the machine that will host your TeamForge site.

☞ **Note:** Your VMware Player machine must have at least 4 GB RAM and a 2 GHz processor.

Installing the VMware Player

1. Before you start TeamForge installation, install the VMware Player.

- Download the VMware Player for Windows (64bit) from *vmware.com*.
- Double-click the `VMware-player.exe` file and follow the instructions.

2. Start the VMware Player.

- Click **Start** > **Programs** > **VMware** > **VMware Player**.

☞ **Note:** You don't have to update the VMware Player when you start the VMware Player for the first time.

Downloading the TeamForge 8.2 installer for VMware Player

3. Contact the *CollabNet Support* and download the TeamForge installer zip file.

4. Unzip the `TeamForge-8.2.0.0-182-VMware.zip` file.

Installing the TeamForge 8.2 on VMware Player

5. Click **Open a Virtual Machine**.

6. Browse and select the `TeamForge-82.ova` file from where you unzipped the `TeamForge-8.2.0.0-182-VMware.zip` file earlier.

7. Click **Import**.
Once successfully imported, you can see a new virtual machine named **"TeamForge-82"** on the VM host list.

8. Click **Edit Virtual machine Settings** and set the following parameters.

 a) Set the **No. of Processors**.
 b) Set the **Memory** to 4 GB. You may increase the memory for better performance.
 c) Set the **MAC ID** of the network adapter.
 d) Click **Save**.

9. Select the **TeamForge-82** virtual machine and click **Play Virtual Machine**.
It may take a few minutes to start the **TeamForge-82** virtual machine depending on your computer's processing speed.

Proceed with configuring your VMware Player installation by following the steps in *Configure CollabNet TeamForge on VMware Player*.

Install TeamForge on VMware ESXi

The VMware ESXi 5.1 or later image server is ideal for large TeamForge user installations due to its simplified installation process. You can download the CollabNet TeamForge installer from CollabNet and unzip it on the machine to host your TeamForge site.

- Install the VMware ESXi 5.1 or later on the server where Teamforge 8.2 would be hosted.
- Install the VMware vSphere Client 5.0 or later on a Windows (client) computer to connect to the VMware ESXi TeamForge server.

To install TeamForge 8.2 on a VMware ESXi server:

1. Contact the *CollabNet Support* and download the TeamForge installer zip file.
2. Unzip the `TeamForge-8.2.0.0-182-VMware.zip` file.
3. Run the VMware vSphere Client and select the VMware ESXi host on which you want to install TeamForge.
4. Select **File** > **Deploy OVF Template**.
   The **Deploy OVF Template** dialog box appears.
5. Click **Browse** and select the `TeamForge-82.ova` file from where you unzipped the `TeamForge-8.2.0.0-182-VMware.zip` file earlier.
6. Click **Next**.
   The **OVF Template Details** wizard appears.
7. Click **Next**.
8. Type a name, for example "TeamForge-8.2" and click **Next**.
9. Select the **Thin Provision** check box and click **Next**.
10. Clear the **Power On after Deployment** check box if it's selected by default.
11. Click **Finish**.
    Wait for the OVF template deployment to complete.
12. Click **Close** when the deployment is complete.
    Open the main VMware vSphere Client window and expand the VMware ESXi host to see the "TeamForge-8.2" virtual machine.
13. Select the "TeamForge-8.2" virtual machine.
14. Select the **Summary** tab from the work area.
15. Click **Edit Settings**.
16. Configure the hardware settings:
    a) Set the memory to 4096 MB. You may increase the memory for better performance.
    b) Set the CPU as 2 Cores.
    c) Set the hard disk memory as 100 GB.
    d) Click **OK**.
17. Start the TeamForge virtual machine: click **Power On** on the **Summary** tab.
    Select the **Console** tab to view the TeamForge virtual machine's console output.

Proceed with configuring your VMware ESXi installation by following the steps in *Configure CollabNet TeamForge on VMware Player*.

Configure CollabNet TeamForge on VMware Player

After you have installed the VMware Player, configure the TeamForge VMware image.

Only one user needs to configure TeamForge. This instance acts as the application server. To access CollabNet TeamForge, the CollabNet TeamForge application server must be running in VMware Player. Other users can access it via a Web browser without running VMware Player.

1. In the VMware Player, log in with the username `root` and the password `changeme`.
2. Enter and confirm a new Linux password.

☞ **Tip:** The system may warn that your password does not meet security standards. For example, it may be too short. This does not mean the password is rejected. If you confirm the same password, it will work.

3. When prompted to run the configuration tool, type `y`.

4. Read the product license agreement.

   Type `q` to close it.

   ☞ **Tip:** You can use the space bar to advance a screen at a time.

5. If you accept the license terms, type `y`.

6. In the CollabNet TeamForge configuration tool, choose **Dynamic Networking (DHCP)** or **Static Networking (Static IP)**:

   - Dynamic networking is useful for a one-person trial installation. It is quick and easy, but email integration with TeamForge will not work correctly.

     ☞ **Note:** If your IP address changes, you may also have to reconfigure source control.

   - Static networking is best if you are evaluating CollabNet TeamForge with a team, or if you already have a license and intend to use TeamForge to support your team.

     To configure static networking, you will need to get a static IP address and hostname from your network administrator, and specify your network settings when prompted.

     ☞ **Tip:** In this case, it's also a good idea to run TeamForge in VMware Player on a dedicated machine.

   The networking for TeamForge is restarted.

7. Specify your outgoing email (SMTP) server.

   - For a one-person evaluation, accept the default value.
   - If you have a CollabNet TeamForge license and intend to send email outside of your firewall, use the SMTP server settings provided by your network administrator.

     ☞ **Note:** Depending on your corporate email configuration, your system administrator may need to permit TeamForge to send mail to the corporate mail server.

8. Choose whether to run CollabNet TeamForge at startup.

   - Choose "Yes" to start CollabNet TeamForge automatically whenever you start the TeamForge VMware image.
   - Choose "No" to require a manual CollabNet TeamForge startup whenever you start the TeamForge VMware image.

9. At the prompt, click **Enter** to start your CollabNet TeamForge site.

   ☞ **Note:** Startup can take several minutes, depending on the speed of the host system. On some slower systems, you may get a false failure message from JBoss, like this: `jboss (app) (localhost:8080) ...........................failed to start in 600 seconds, giving up now. Please check the log: /opt/collabnet/teamforge/ log/apps/service.log FAILED`

   This can safely be ignored.

   ☞ **Tip:** From now on, you can stop and restart TeamForge using these commands:

   ```
   /etc/init.d/httpd stop
   /etc/init.d/collabnet stop
   /etc/init.d/postgresql-9.3 stop

   /etc/init.d/httpd start
   /etc/init.d/postgresql-9.3 start
   /etc/init.d/collabnet start
   ```

☞ **Note:** If you are configuring SSL for the site, after starting all `collabnet` services, execute`iptables -F` before accessing the site from the UI.

10. This step is valid if you are setting up a production environment.

   a) Stop TeamForge.

```
/etc/init.d/collabnet stop all
```

   b) Add a 'DOMAIN_localhost' token in the `site-options.conf` file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
DOMAIN_localhost=my.app.domain.com
```

   c) Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
./install.sh -r -n -I -V
```

   d) Start TeamForge and run the `post-install.py` script.

```
/etc/init.d/collabnet start
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

Integrate Binaries with VMWare

To access Binaries from a project, configure the TeamForge VMWare image in the following way:

11. Export the following environment variables:

```
export COLLAB_HOME=/opt/collabnet
export APP_DIR="/opt/collabnet/integrations/collabnet-binary-1.0.0.0"
export PATH=$APP_DIR/vendor/bundle/ruby/2.1.0/bin:$COLLAB_HOME/ruby/bin/:
$PATH
export GEM_HOME=$APP_DIR/vendor/bundle/ruby/2.1.0/
export ENVIRONMENT=production
export RAILS_ENV=production
export RAILS_RELATIVE_URL_ROOT='/iaf/'
```

12. Open `config/settings.yml`.

```
vi /opt/collabnet/integrations/collabnet-binary-1.0.0.0/config/
settings.yml
```

13. Configure the following in the production section:

☞ **Note:** Depending on how you have configured your VMware, (with the IP address or the domain name), the `ctf_base_url` and `iaf_host_url` values need to be specified.

```
ctf_base_url: '<IP address or domain_name of VMware instance>'
```

For example, if you have configured your VMware with the IP address, the value is: `ctf_base_url: 'http://10.2.0.196'`

```
iaf_host_url: '<IP address or domain_name of VMware instance>'
```

```
iaf_app_name: 'Binaries'
app_url_prefix: 'binary'
```

14. Restart Binaries.

```
/etc/init.d/collabnet restart binary
```

15. Run the following script from the location, `/opt/collabnet/integrations/collabnet-binary-1.0.0.0`:

```
bin/ctf_configure -u<site admin username> -p<site admin password> -e
```

For example, `bin/ctf_configure -uadmin -padmin -e`

16. Log into your new site with the user name `admin` and the password `admin`.

The URL for your site is the IP address or domain name provided in the Linux console at the end of the installation process.

☞ **Important:** You must change your administrator password when you first log in. After logging in, you must change the administrator email ID (**My Workspace** > **My Settings** > **Edit**) with a valid email ID of the format abc@xyz.com.

Configure the CLI Jobs Linked Application's URL

☞ **Important:** Configure the CLI Jobs linked application's URL only if the site option token *HOST_localhost* is configured as "localhost" in the site-options.conf file.

17. After logging on to TeamForge, click the **Projects** tab.

18. Click the **Look** project.

19. Click **PROJECT ADMIN** from the **Project Home** menu.

20. Click **Project Toolbar** from the **Project Admin Menu**.

21. Select the **Linked Applications** tab.

22. Select the **CLI Jobs** linked application's check box and click **Edit**.

23. Modify the URL: Replace the "localhost" with the hostname or IP address of the computer that hosts the TeamForge application.

24. Click **Save**.

☞ **Note:** To track the ETL job failures, configure the token SOAP_ANONYMOUS_SHARED_SECRET in site-options.conf and re-create runtime. This token sends notifications when the ETL job fails. For more details, refer to: *SOAP_ANONYMOUS_SHARED_SECRET*.

Supply your TeamForge license key

Your license key enables you to use CollabNet TeamForge for the period of your contract.

Your license key will only work for the IP address of the machine that your CollabNet TeamForge is running on, as specified in your order form.

☞ **Tip:** These steps are for installing your license key via the web interface. If you prefer, you can install it as a text file instead. See *Supply your CollabNet TeamForge license key as a text file* on page 435.

1. Locate the confirmation email you received from your CollabNet representative when you purchased your contract.

2. Log into your site as the site administrator.

☞ **Note:** The site administrator is different from the root user on the machine where the site is running.

3. Click **Admin** > **License Key**.
   If you have entered a license before, the IP address and current licensed number of users on your site are listed on the **License Key** page. Verify that the IP address is the same as the one you entered in your order form.

4. Click **Enter License Key**.

5. Copy your new license key from the confirmation email and paste it into the **Enter License Key** field.
   A license key string looks like this:
   25:supervillaininc:144.16.116.25.:302D02150080D7853DB3E5C6F67EABC65BD3AC17D4D35CB3Z002

   ☞ **Tip:** Save this license key in case you need to reinstall CollabNet TeamForge.

6. Click **Save**.

7. Verify that the new value for **Licensed Number of Users** matches the total number of licensed users in your contract.

Upgrade to TeamForge 8.2

When you upgrade to TeamForge 8.2 on RHEL/CentOS 7.2, you can have Black Duck Code Sight on a separate RHEL/CentOS 6.7 server. Choose the instructions that fit your case.

TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

Plan your upgrade

As the first step in upgrading to TeamForge 8.2, consider some key things that will affect how your new site works.

Overview

A TeamForge site consists of a core TeamForge application and several tightly integrated services that support it. In addition, you can integrate TeamForge with other third party applications such Review Board, Nexus, Black Duck Code Sight, Git, and so on.

- The core TeamForge application provides the Web interface that users see, and the API that other applications can interact with. It also includes the file system where some user content is stored, such as wiki pages.
- The site database is where most of the user-created content is stored and accessed. Documents, discussion posts, tracker artifacts, project administration settings: all that sort of thing lives in the database.
- The source control server ties any number of Subversion or CVS repositories into the TeamForge site.
- The Extract Transform and Load (ETL) server pulls data from the site database and populates the datamart to generate charts and graphs about how people are using the site.
- The datamart (Reports DB) is an abstraction of the site database, optimized to support the reporting functionality.

Integrations

TeamForge supports integration with a wide array of third party applications such as Review Board, Git, Black Duck Code Sight, Nexus, Jira and so on. As a customer you may or may not always want (or have) all of TeamForge's supported integrated applications. It's also quite possible that some of the integrated applications may not always run on all the platforms supported by TeamForge.

To accommodate a wider audience, by default, TeamForge install and upgrade instructions include steps to integrate such third party applications with TeamForge. However, use your discretion to ignore and skip such steps if they are not relevant to your site. See *Software requirements for CollabNet TeamForge 8.2* on page 748 to understand what it takes to run TeamForge 8.2 with integrations.

Upgrade path

TeamForge 8.2 upgrade instructions are for upgrading from TeamForge 8.1 (including patch releases, if any) to TeamForge 8.2. Contact *CollabNet Support* if you want to upgrade from TeamForge 8.0 or earlier versions to TeamForge 8.2.

One-hop upgrade compatibility for TeamForge 8.2

☞ **Note:**

> The TeamForge 8.2 installer supports one-hop upgrade from TeamForge 7.1 or later versions only.

During an upgrade, the TeamForge 8.2 migration script detects the TeamForge version you run, checks if it's TeamForge 7.1 or later, and if 'true', proceeds with the data migration.

The migration script aborts with an error message if it detects TeamForge 7.0 or earlier versions. You must upgrade your site to TeamForge 7.1 or later and then upgrade to TeamForge 8.2.

Services and hosts

TeamForge consists of several interrelated services (such as the TeamForge application, database, datamart, indexer and so on) and other integrated applications that can run on separate hardware or share one or more servers in various configurations. If you aren't the person who first installed your current TeamForge site (or maybe, even if you are), it's essential to catalog the hosts where your services are running and to know what configuration has been applied to them.

Branding changes

Every release of TeamForge can bring changes to the look and feel of the product. TeamForge 8.2 is no exception. If you have edited files in your site's branding repository (that's how you customize the look and feel of the product), you must download the new branding package and check into your branding repository the new versions of any files you have edited. See *Customize anything on your site* on page 290 for instructions.

PostgreSQL or Oracle?

PostgreSQL 9.3.6 is installed automatically when you install TeamForge 8.2. If you intend to use Oracle, CollabNet recommends that you let the installer run its course, make sure things work normally, and then set up your Oracle database and switch over to it.

If you want to use Oracle as your database, consider the following points:

- TeamForge 8.2 supports Oracle server 12c and Oracle client 12c.
- Oracle express edition is not supported for both client and server.
- Black Duck Code Sight (BDCS) 2.1.3 was tested with PostgreSQL 9.3.6 only. BDCS with Oracle was not tested. If you want BDCS, you may install BDCS on the PostgreSQL database that is installed by default during TeamForge installation.
- Review Board 2.0.11 was tested with PostgreSQL 9.3.6 only. Review Board with Oracle was not tested. Note that Review Board must be installed on the TeamForge application server and therefore uses the PostgreSQL database that is installed by default during TeamForge installation.
- GIT integration works only with PostgreSQL. The Git integration will use PostgreSQL even if your TeamForge site uses Oracle.

The efficiency of your database can have an impact on your users' perception of the site's usability. If your site uses a PostgreSQL database (which is the default), you may want to consider tuning it to fit your specific circumstances. The default settings are intended for a small-to-medium site running on a single server. See *What are the right PostgreSQL settings for my site?* on page 909 for recommendations from CollabNet's performance team on optimizing PostgreSQL for different conditions.

Upgrade considerations for RHEL/CentOS 6.7 and 7.x

Though TeamForge 8.2 can run on RHEL/CentOS 6.7, it is a best practice to upgrade to TeamForge 8.2 on the most recent version of RHEL/CentOS supported by TeamForge, which in this case is RHEL/CentOS 7.2. However, choosing between RHEL/CentOS 6.7 and 7.2 depends on whether or not you 'already have' Black Duck Code Sight.

As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, CollabNet recommends that you have Black Duck Code Sight run on a separate RHEL/CentOS 6.7 server if you choose upgrade to TeamForge 8.2 on a RHEL/CentOS 7.x server.

'Team' feature and its implications

☞ **Important:**

Ignore this if you are upgrading from TeamForge 8.0 to 8.1 or later.

With the 'Team' feature added to TeamForge 8.0, any field by the name 'team', if already existing in your trackers, might cause a conflict while running the `migrate.py` script. To overcome such a conflict, follow the steps in the following *Troubleshooting FAQ*.

Upgrade to TeamForge 8.2 on RHEL/CentOS

You can upgrade from TeamForge 8.1 to TeamForge 8.2. You can either upgrade on the same hardware where your TeamForge 8.1 site is running or you can take this opportunity to move your site to a new hardware. You can upgrade to TeamForge 8.2 either on RHEL/CentOS 6.7 or 7.2. Choose the instructions that fit your case.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Note:** The YUM installer installs the CollabNet TeamForge site in the default directory `"/opt/ collabnet/teamforge"`.

Upgrade to TeamForge 8.2 on new hardware: RHEL/CentOS 7.2

You can upgrade to TeamForge 8.2 on a new RHEL/CentOS 7.2 server if you have TeamForge 8.1 running on a RHEL/CentOS 6.7 or 7.0 server.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

Upgrade to TeamForge 8.2 on new hardware - Black Duck Code Sight on a separate server

To upgrade to TeamForge 8.2, set up a new hardware, then bring your old site's data and convert it. As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, you must install Black Duck Code Sight on a separate RHEL/CentOS 6.7 server and move your Black Duck Code Sight data to this server.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Database Server (Operational DB and Reports DB)
- ETL Server
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer).

Black Duck Code Sight Server runs on a separate RHEL/CentOS 6.7 server (we call this my.codesight.host).

Log on to the server as root user always.

Do the following on the new TeamForge Application Server - my.app.host

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

   - Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
   - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
   - See *the Red Hat installation guide* for help.

   ☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf,` and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

3. Check your basic networking setup.

   See *Set up networking for your TeamForge server* on page 372 for details.

Do the following on the existing TeamForge application server

4. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

5. Back up your site database.

☞ **Tip:** `/tmp/backup_dir` is just an example. You can use any directory or partition you prefer to store your backup files.

```
mkdir -p /tmp/backup_dir
cd /var/lib
tar -zcvf /tmp/backup_dir/pgsql.tgz pgsql/9.3
```

6. Back up the file system data.

☞ **Tip:** `/tmp/backup_dir` is just an example. You can use any directory or partition you prefer to store your backup files.

a) Make an archive file with the following data directories:

Directory	Contents
/opt/collabnet/teamforge/var	User-created data, such as artifact attachments
/opt/collabnet/reviewboard	Review Board data
/svnroot	Subversion source code repositories
/sf-svnroot	Subversion repository for branding data
/cvsroot	CVS source code repositories (not present on all sites)
/gitroot	GIT source code repositories

```
cp -Rpfv /svnroot /sf-svnroot /cvsroot /opt/collabnet/teamforge/var /
opt/collabnet/reviewboard /tmp/backup_dir
```

If GIT integration is enabled, do the following:

```
mkdir /tmp/backup_dir/gerrit
cp -Rpfv /gitroot /tmp/backup_dir
cp -Rpfv /opt/collabnet/gerrit/ /tmp/backup_dir/gerrit
```

Compress your backup data.

```
cd /tmp
tar czvf 81backup.tgz backup_dir
```

b) Back up your SSH keys, if any.
c) Back up your SSL certificates and keys, if any.

7. Copy the master configuration file from the old server to the `/tmp` directory on the new TeamForge and Black Duck Code Sight servers.

```
scp /opt/collabnet/teamforge-installer/8.1.0.x/conf/site-options.conf
username@newbox:/tmp
```

☞ **Note:** Replace "x" with the appropriate patch release number if applicable.

☞ **Tip:** scp is just an example. You can choose any file transfer method you prefer.

8. Copy the file system data to the new TeamForge and Black Duck Code Sight servers.

```
scp /tmp/81backup.tgz username@newbox:/tmp
```

Do the following on the new TeamForge Application Server - my.app.host

9. Unpack the file system data.

```
cd /tmp
tar xzvf 81backup.tgz
```

10. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

11. Upgrade the operating system packages.

```
yum upgrade
```

12. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```

Save the file and reboot the server.

13. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
```

```
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

14. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

15. Reload the PostgreSQL data.

```
cd /var/lib
mv pgsql pgsql_orig
tar -zxvf /tmp/backup_dir/pgsql.tgz
```

16. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

> ⚠️ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

b) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options.conf` file in the `/tmp` directory and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

c) Update the host name and domain name, if required.

```
HOST_localhost=app database datamart etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

Add 'gerrit' to the HOST_localhost token if you are installing Git.

```
HOST_localhost=app database datamart etl indexer subversion cvs gerrit
```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs
 gerrit binary
```

Configure the following token if you are installing Black Duck Code Sight.

```
HOST_my.codesight.domain.com=codesearch
```

d) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

e) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

f) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

g) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: $<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

h) Ensure to set the token *DEDICATED_INSTALL=true*.

i) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

j) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

k) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

l) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

m) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

n) Ensure to set the token *SELINUX_SETUP=true*.

o) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) Save the `site-options.conf` file.

17. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do the following on the Black Duck Code Sight server - my.codesight.host.

18. Set up Black Duck Code Sight on RHEL/CentOS 6.7. See *Install Black Duck Code Sight on a separate RHEL/CentOS 6.7 server*.

Do the following on the new TeamForge Application Server - my.app.host

19. Reload the `svnroot`, `sf-svnroot`, `cvsroot`, `gitroot` and `var` directories.

```
mv svnroot/ svnroot_orig
mv cvsroot/ cvsroot_orig
If you have Git integration: mv gitroot/ gitroot_orig
cp -Rpfv /tmp/backup_dir/svnroot /svnroot
cp -Rpfv /tmp/backup_dir/cvsroot /cvsroot
cp -Rpfv /tmp/backup_dir/sf-svnroot /sf-svnroot
cp -Rpfv /tmp/backup_dir/var /opt/collabnet/teamforge/var
```

If Git integration is enabled, do the following:

```
cp -Rpfv /tmp/backup_dir/gitroot /gitroot
cp -Rpfv /tmp/backup_dir/gerrit/gerrit/etc /opt/collabnet/gerrit
cp -Rpf /tmp/backup_dir/gerrit/gerrit/.ssh /opt/collabnet/gerrit
cp -Rpf /tmp/backup_dir/gerrit/gerrit/bin /opt/collabnet/gerrit
```

20. Recreate the runtime environment to set the database credentials.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

21. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

> ☞ **Note:** This process can take a long time for sites with a lot of data.

22. Convert your site data to work with TeamForge 8.2.

☞ **Tip:** Before you kick off the data migration, do the following:

- Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
- If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

23. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

24. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

25. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

26. Run the following query to get the repository IDs of the repositories for which the *Available in Search Results* option was enabled.

```
echo "select sr.id from folder f, project p,scm_repository sr where
f.id like 'reps%'and p.is_deleted='f' and p.id=f. project_id and
sr.search_enabled='t'and sr.id=f.id and f.is_deleted='f';" | /opt/
collabnet/teamforge/runtime/scripts/psql-wrapper -t -A >>/tmp/repository-
ids.txt
```

27. As this is a new hardware upgrade, run the `codesightmigration.sh` script to have the code search project repository URLs point to the new server.

☞ **Note:** Make sure TeamForge and Black Duck Code Sight licenses are installed before running the `codesightmigration.sh` script.

```
/opt/collabnet/teamforge/runtime/scripts/codesearch/codesightmigration.sh
--filename=/tmp/repository-ids.txt
```

Do the following on the Black Duck Code Sight server - my.codesight.host.

28. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do the following on the new TeamForge Application Server - my.app.host

29. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

30. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

* **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
* **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
* **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
* **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
* **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
* **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
* **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 on same hardware: RHEL/CentOS 6.7

If your TeamForge 8.1 site runs on a RHEL/CentOS 6.7 server, you can upgrade to TeamForge 8.2 on the same server.

While it's desirable to upgrade to the latest RHEL/CentOS version, one of the advantages of upgrading to TeamForge 8.2 on RHEL/CentOS 6.7 is the ability to run Black Duck Code Sight on RHEL/CentOS 6.7. Remember, Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

Upgrade to TeamForge 8.2 - All services on the same RHEL/CentOS 6.7 server

In this procedure, we'll assume that you are upgrading on the same RHEL/CentOS 6.7 server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, all the following services run on a single RHEL/CentOS 6.7 server (we call this my.app.host).

* TeamForge Application Server
* Black Duck Code Sight Server
* Database Server (Operational DB and Reports DB)
* ETL Server
* GIT Integration Server
* SCM Integration Server (Subversion and CVS)
* Search Server (Indexer)

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

2. Stop all TeamForge application services.

   ```
   /etc/init.d/collabnet stop all
   ```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

5. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```

   b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

   ```
   setenforce 0
   ```

   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

   b) Install the repository package.

   ```
   yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
   ```

   c) Refresh your repository cache.

   ```
   yum clean all
   ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

   • Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

   b) Unpack the disconnected installation package.

   ```
   rpm -Uvh <package-name>
   ```

c) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

8. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

👉 **Tip:** If you have GIT and Black Duck Code Sight already, they are automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-steps.

b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

c) Black Duck Code Sight: To install the Black Duck Code Sight packages run the following command:

```
yum install teamforge-codesearch
```

9. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

⚠ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app database datamart etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

Add "gerrit" to the HOST_localhost token if you are installing GIT.

```
HOST_localhost=app database datamart etl indexer subversion cvs gerrit
```

Add "codesearch" to the HOST_localhost token if you are installing Black Duck Code Sight.

```
HOST_localhost=app database datamart etl indexer subversion
 cvs codesearch
```

Add "binary" to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs
 codesearch binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

> ☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"``

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) Save the `site-options.conf` file.

10. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

11. Convert your site data to work with TeamForge 8.2.

> ☞ **Tip:** Before you kick off the data migration, do the following:
>
> - Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
> - If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

12. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

13. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

14. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

15. Install the Black Duck Code Sight license. For more information see *these instructions.*

16. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

17. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

18. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

TeamForge Avatar display issue on RHEL/CentOS 6.7	TeamForge Avatar image is not displayed properly post Review Board installation on RHEL/CentOS 6.7. Run the following commands to work around this issue:
	
	```yum erase python-imaging```
	```yum install teamforge```
	```/etc/init.d/httpd restart```

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

• **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.

• **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.

• **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.

• **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.

- **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
- **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - Database and Datamart on a separate server

In this procedure, we'll assume that you are upgrading on the same server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Black Duck Code Sight Server
- ETL Server
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer)

Both the operational and reports databases run on the database server (we call this my.db.host).

- Database Server (Operational DB and Reports DB)

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

2. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

5. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   ```

```
sf-admin soft nofile 65536
```
Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

8. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

> ☞ **Tip:** If you have GIT and Black Duck Code Sight already, they are automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-steps.

b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

c) Black Duck Code Sight: To install the Black Duck Code Sight packages run the following command:

```
yum install teamforge-codesearch
```

9. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

> ⚠ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.db.domain.com=database datamart
```

Add "gerrit" to `Host_localhost` if you are installing Git.

```
HOST_localhost=app etl indexer subversion cvs gerrit
```

Add "codesearch" to `Host_localhost` if you are installing Black Duck Code Sight.

```
HOST_localhost=app etl indexer subversion cvs codesearch
```

Add "binary" to `Host_localhost` if you are installing Nexus.

```
HOST_localhost=app etl indexer subversion cvs codesearch binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

> ☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

> ⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

> ☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: $<>/\'"`

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) Save the `site-options.conf` file.

10. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

11. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

> ☞ **Note:** This process can take a long time for sites with a lot of data.

Do this on the database server - my.db.host

12. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
yum erase subversion-devel -y
```

b) Upgrade the operating system packages.

```
yum upgrade
```

13. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

14. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

- Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

15. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

16. Install the TeamForge database packages.

```
yum install teamforge-database
```

17. Copy the site-options.conf file from **my.app.host** and modify the token settings.

👉 **Note:** If you choose to use the old site-options.conf file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.db.host=database datamart
```

👉 **Note:** 'HOST_my.db.host' is just an example. As you are installing database on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.db.host=my.db.domain.com
```

```
HOST_my.app.domain.com=app etl indexer subversion cvs
```

b) Save the site-options.conf file.

18. ☞ **Important:** This step is required if and only if you are upgrading from TeamForge 8.0 (or earlier versions) to TeamForge 8.2.

As JDK is upgraded to 1.8.0_45, move the existing runtime directory before create-runtime.

```
cd /opt/collabnet/teamforge/
mv runtime/ runtime_old
```

19. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the application server - my.app.host

20. Convert your site data to work with TeamForge 8.2.

☞ **Tip:** Before you kick off the data migration, do the following:

- Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
- If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

21. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

22. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

23. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

24. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

25. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.
b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

26. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

TeamForge Avatar display issue on RHEL/CentOS 6.7	TeamForge Avatar image is not displayed properly post Review Board installation on RHEL/CentOS 6.7. Run the following commands to work around this issue:
	``` yum erase python-imaging yum install teamforge /etc/init.d/httpd restart ```

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

- **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
- **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
- **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
- **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - Reporting services on a separate server

In this procedure, we'll assume that you are upgrading on the same RHEL/CentOS 6.7 server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Black Duck Code Sight Server
- Database Server (Operational DB)
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer)

The following service runs on the database server (we call this my.reports.host).

- Database Server (Reports DB)
- ETL Server

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

```
cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
custom-install --uninstall
yum erase teamforge_cli_server
```

2. Stop TeamForge.

```
/etc/init.d/collabnet stop all
```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

```
yum upgrade
```

5. Create a file, /etc/security/limits.d/teamforge.conf, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```

   Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

   b) If the output of the getenforce command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to /tmp.

   b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

   c) Refresh your repository cache.

```
yum clean all
```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

   • Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

   b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

   c) **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

   Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

8. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

9. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

☞ **Tip:** If you have GIT and Black Duck Code Sight already, they are automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-steps.

b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

c) Black Duck Code Sight: To install the Black Duck Code Sight packages run the following command:

```
yum install teamforge-codesearch
```

10. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

⚠ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app database indexer subversion cvs
```

```
HOST_my.reports.domain.com=datamart etl
```

```
DOMAIN_localhost=my.app.domain.com
```

Add "gerrit" to `Host_localhost` if you are installing Git.

```
HOST_localhost=app database indexer subversion cvs gerrit
```

Add 'codesearch' to the HOST_localhost token if you are installing Black Duck Code Sight.

```
HOST_localhost=app database indexer subversion cvs codesearch
```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database indexer subversion cvs codesearch binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

> ☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

> ⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

> ☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) Save the `site-options.conf` file.

11. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

12. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

> ☞ **Note:** This process can take a long time for sites with a lot of data.

Do this on the reports server - my.reports.host

13. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
```

```
yum erase subversion-devel -y
```
b) Upgrade the operating system packages.

```
yum upgrade
```

14. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```
b) If the output of the getenforce command is either "Disabled" or "Permissive", SELinux is already disabled.
c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

15. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to /tmp.
b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```
c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

- Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

16. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

17. Install the TeamForge database packages.

```
yum install teamforge-database teamforge-etl
```

18. Copy the `site-options.conf` file from **my.app.host** and modify the token settings.

> ☞ **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.reports.host=datamart etl
```

> ☞ **Note:** 'HOST_my.reports.host' is just an example. As you are installing reporting services on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.reports.host=my.reports.domain.com
```

```
HOST_my.app.domain.com=app database indexer subversion cvs
```

b) Save the `site-options.conf` file.

19. ☞ **Important:** This step is required if and only if you are upgrading from TeamForge 8.0 (or earlier versions) to TeamForge 8.2.

As JDK is upgraded to 1.8.0_45, move the existing runtime directory before create-runtime.

```
cd /opt/collabnet/teamforge/
mv runtime/ runtime_old
```

20. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the application server - my.app.host

21. Convert your site data to work with TeamForge 8.2.

> ☞ **Tip:** Before you kick off the data migration, do the following:
> - Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
> - If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

22. Start TeamForge.

```
/etc/init.d/collabnet start
```

> ☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

23. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

24. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

25. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

26. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.
b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

27. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

TeamForge Avatar display issue on RHEL/CentOS 6.7	TeamForge Avatar image is not displayed properly post Review Board installation on RHEL/CentOS 6.7. Run the following commands to work around this issue:
	```yum erase python-imaging``` ```yum install teamforge``` ```/etc/init.d/httpd restart```

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

- **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
- **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
- **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - Black Duck Code Sight on a separate server
In this procedure, we'll assume that you are upgrading on the same server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

👉 **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- ETL Server
- Database Server (Operational DB and Reports DB)
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer)

The following service runs on the code sight server (we call this my.codesight.host).

- Black Duck Code Sight Server

👉 **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   👉 **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

2. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

5. Create a file, `/etc/security/limits.d/teamforge.conf,` and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```
   b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

   ```
   setenforce 0
   ```
   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to /tmp.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

   • Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

8. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

9. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

☞ **Tip:** If you have GIT already, it is automatically upgraded when you run yum install teamforge. In such case, you can skip the following sub-step.

b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

10. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

⚠️ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app database datamart etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.codesight.domain.com=codesearch
```

Add 'gerrit' to the HOST_localhost token if you are installing Git.

```
HOST_localhost=app database datamart etl indexer subversion cvs gerrit
```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs
 gerrit binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"``

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.

When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) Save the `site-options.conf` file.

11. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

12. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

☞ **Note:** This process can take a long time for sites with a lot of data.

Do this on the Black Duck Code Sight Server - my.codesight.host

13. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
yum erase subversion-devel -y
```

b) Upgrade the operating system packages.

```
yum upgrade
```

14. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

15. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

16. Stop the Black Duck Code Sight service.

```
/etc/init.d/collabnet stop tomcatcs
```

17. Install Black Duck Code Sight.

```
yum install teamforge-codesearch
```

18. Copy the master `site-options.conf` file from my.app.host and modify the token settings.

☞ **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.codesight.host=codesearch
```

```
DOMAIN_my.codesight.host=my.codesight.domain.com
```

```
Host_my.app.domain.com=app database datamart etl indexer subversion cvs
```

b) Save the `site-options.conf` file.

19. ☞ **Important:** This step is required if and only if you are upgrading from TeamForge 8.0 (or earlier versions) to TeamForge 8.2.

As JDK is upgraded to 1.8.0_45, move the existing runtime directory before create-runtime.

```
cd /opt/collabnet/teamforge/
mv runtime/ runtime_old
```

20. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

21. Start the Black Duck Code Sight service.

```
/etc/init.d/collabnet start tomcatcs
```

22. To install the license for Black Duck Code Sight follow *these instructions.*

Do this on the application server - my.app.host

23. Convert your site data to work with TeamForge 8.2.

☞ **Tip:** Before you kick off the data migration, do the following:

- Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
- If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

24. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

25. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

26. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

Do this on the Black Duck Code Sight server - my.codesight.host

27. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

28. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do this on the application server - my.app.host

29. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.
b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

30. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

TeamForge Avatar display issue on RHEL/CentOS 6.7	TeamForge Avatar image is not displayed properly post Review Board installation on RHEL/CentOS 6.7. Run the following commands to work around this issue:

```
yum erase python-imaging
yum install teamforge
/etc/init.d/httpd restart
```

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

* **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
* **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
* **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
* **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
* **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
* **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
* **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - GIT on a separate server

In this procedure, we'll assume that you are upgrading on the same server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

* TeamForge Application Server
* Black Duck Code Sight Server
* ETL Server
* Database Server (Operational DB and Reports DB)
* SCM Integration Server (Subversion and CVS)
* Search Server (Indexer)

The following service runs on the GIT integration server (we call this my.git.host).

* GIT Integration Server

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

2. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

```
yum upgrade
```

5. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```

Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

8. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

9. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

☞ **Tip:** If you have Black Duck Code Sight already, it is automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-step.

b) Black Duck Code Sight: To install the Black Duck Code Sight packages run the following command:

```
yum install teamforge-codesearch
```

10. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

⚠ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-
options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app database datamart etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.git.domain.com=gerrit
```

Add 'codesearch' to the HOST_localhost token if you are installing Black Duck Code Sight.

```
HOST_my.localhost= app database datamart etl indexer subversion
 cvs codesearch
```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_my.localhost= app database datamart etl indexer subversion cvs
 codesearch binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: $<>/\'"`

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) ☞ **Important:** This step is required if you want Git notification emails.

Update the *JAMES_ACCEPTED_RELAYS* site-options token with the Git server's IP address. See *JAMES_ACCEPTED_RELAYS* on page 794 for more information.

```
JAMES_ACCEPTED_RELAYS=127.0.0.1,{__CEE_DOMAIN__},<The IP address of the
  Git server>
```

r) Save the `site-options.conf` file.

11. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

12. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

> ☞ **Note:** This process can take a long time for sites with a lot of data.

13. Convert your site data to work with TeamForge 8.2.

> ☞ **Tip:** Before you kick off the data migration, do the following:
>
> • Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
> • If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

14. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

15. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

16. Run the TeamForge post installation script. For more information, see *post-install.py*.

   a) Use the following command to run the `post-install.py` script.

   ```
   /opt/collabnet/teamforge/runtime/scripts/post-install.py
   ```

   ☞ **Note:** Enter the 'gitadmin' user name and password when prompted.

   ☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

17. Install the Black Duck Code Sight license on the server where Black Duck Code Sight is installed. For more information see *these instructions.*

18. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do this on the Git integration server - my.git.host

19. Back up the Git file system data.

   a) Make an archive file with the following data directories.

   ☞ **Tip:** `/tmp` is just an example. You can use any directory or partition that you prefer.

Directory	Contents
/gitroot	Git source code repositories

   ```
   cp -Rpfv /gitroot /tmp/gitbackup_dir
   cp -Rpfv /opt/collabnet/gerrit/ /tmp/gitbackup_dir/gerrit
   ```

   b) Back up your SSH keys, if any.

20. Upgrade the operating system packages.

```
yum upgrade
```

21. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```

   b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

   c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

22. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

   - Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

23. Install the Git packages.

```
yum install teamforge-git
```

24. Copy the `site-options.conf` file from my.app.host and modify the token settings.

☞ **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.git.host=gerrit
```

```
DOMAIN_my.git.host=my.git.domain.com
```

```
HOST_my.app.domain.com=app database datamart etl indexer subversion cvs
```

b) Save the `site-options.conf` file.

25. ☞   **Important:** This step is required if and only if you are upgrading from TeamForge 8.0 (or earlier versions) to TeamForge 8.2.

As JDK is upgraded to 1.8.0_45, move the existing runtime directory before create-runtime.

```
cd /opt/collabnet/teamforge/
mv runtime/ runtime_old
```

26. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

27. Run the TeamForge post installation script. For more information, see *post-install.py*.

a) Use the following command to run the `post-install.py` script.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞   **Note:** Enter the 'gitadmin' user name and password when prompted.

☞   **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

28. ☞   **Important:** This step is required if you want Git notification emails.

Edit the `/opt/collabnet/gerrit/etc/gerrit.config` file and update the 'smtpServer' property with the TeamForge application server's host name.

For example:

```
[sendemail]
            smtpServer = <TeamForge application server host name>
```

a) Restart gerrit.

```
/etc/init.d/collabnet stop gerrit
/etc/init.d/collabnet start gerrit
```

Do the following on the application server - my.app.host

29. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞   **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

30. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

TeamForge Avatar display issue on RHEL/CentOS 6.7	TeamForge Avatar image is not displayed properly post Review Board installation on RHEL/CentOS 6.7. Run the following commands to work around this issue: ``` yum erase python-imaging yum install teamforge /etc/init.d/httpd restart ```

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

- **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
- **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
- **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to ALLOW-FROM http:// forge.collab.net overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
- **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - Database and SCM on separate servers

In this procedure, we'll assume that you are upgrading on the same server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Black Duck Code Sight Server
- ETL Server
- Search Server (Indexer).

The following service runs on the database server (we call this my.db.host).

- Database Server (Operational DB and Reports DB)

The following services run of the SCM server (we call this my.scmandgit.host).

- SCM Integration Server (Subversion and CVS)
- GIT Integration Server

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   ```

```
custom-install --uninstall
yum erase teamforge_cli_server
```

2. Stop TeamForge.

```
/etc/init.d/collabnet stop all
```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

```
yum upgrade
```

5. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```
Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

• Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

8. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

☞ **Tip:** If you have Black Duck Code Sight already, it is automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-step.

b) Black Duck Code Sight: To install the Black Duck Code Sight packages run the following command:

```
yum install teamforge-codesearch
```

9. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

⚠ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-
options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app etl indexer
```

```
HOST_my.db.domain.com=database datamart
```

```
HOST_my.scmandgit.domain.com=subversion cvs gerrit
```

```
DOMAIN_localhost=my.app.domain.com
```

Add 'codesearch' to the HOST_localhost token if you are installing Black Duck Code Sight.

```
HOST_localhost=app etl indexer codesearch
```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app etl indexer codesearch binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

> ☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

> ⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

> ☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: $<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) ☞ **Important:** This step is required if you want Git notification emails.

Update the *JAMES_ACCEPTED_RELAYS* site-options token with the Git server's IP address. See *JAMES_ACCEPTED_RELAYS* on page 794 for more information.

```
JAMES_ACCEPTED_RELAYS=127.0.0.1,{__CEE_DOMAIN__},<The IP address of the
 Git server>
```

r) Save the `site-options.conf` file.

10. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

11. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

> ☞ **Note:** This process can take a long time for sites with a lot of data.

Do the following on the database server - my.db.host

12. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
yum erase subversion-devel -y
```

b) Upgrade the operating system packages.

```
yum upgrade
```

13. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

   b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

   c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

14. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

   b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

   c) Refresh your repository cache.

```
yum clean all
```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

   - Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

   b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

   c) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

   Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

   In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

   d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

   Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

   e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

15. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

16. Install the TeamForge database packages.

```
yum install teamforge-database
```

17. Copy the `site-options.conf` file from my.app.host and modify the token settings.

> 👉 **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.db.host=database datamart
```

> 👉 **Note:** 'HOST_my.db.host' is just an example. As you are installing database on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.db.host=my.db.domain.com
```

```
HOST_my.app.domain.com=app etl indexer
```

```
HOST_my.scmandgit.domain.com=subversion cvs gerrit
```

b) Save the `site-options.conf` file.

18. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the application server - my.app.host

19. Convert your site data to work with TeamForge 8.2.

> 👉 **Tip:** Before you kick off the data migration, do the following:
> - Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
> - If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

20. Start TeamForge.

```
/etc/init.d/collabnet start
```

21. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

22. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

> 👉 **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

23. Install the Black Duck Code Sight license on the server where Black Duck Code Sight is installed. For more information see *these instructions.*

24. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do the following on the SCM server - my.scmandgit.host

If your TeamForge setup includes source control running on its own server, you'll have to upgrade that server as well as the main TeamForge application server.

25. Stop TeamForge.

```
/etc/init.d/httpd stop
/etc/init.d/collabnet stop tomcat
```

26. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

27. Run the following commands to upgrade Red Hat/CentOS to the latest version.

  a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
yum erase subversion-devel -y
```

  b) Upgrade the operating system packages.

```
yum upgrade
```

28. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

  a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

  b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

  c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

29. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

  a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

  b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

  c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

  a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

  • Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

  b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

  c) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

30. If Git is enabled and the PostgreSQL database is running locally:

a) Stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.2 stop
```

31. Install the following application packages.

a) Install the source code component.

```
yum install teamforge-scm
```

b) To install the Git packages, run the following command.

```
yum install teamforge-git
```

32. Copy the master `site-options.conf` file from my.app.host and modify the host token settings in the `site-options.conf` file.

> ☞ **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

```
HOST_my.scmandgit.host=subversion cvs gerrit
```

```
DOMAIN_my.scmandgit.host=my.scmandgit.domain.com
```

```
HOST_my.app.domain.com=app etl indexer codesearch
```

```
HOST_my.db.domain.com=database datamart
```

33. Save the `site-options.conf`file.

34. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

35. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

> ☞ **Note:** This process can take a long time for sites with a lot of data.

36. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

37. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

38. ☞ **Important:** This step is required if you want Git notification emails.

Edit the `/opt/collabnet/gerrit/etc/gerrit.config` file and update the 'smtpServer' property with the TeamForge application server's host name.

For example:

```
[sendemail]
                smtpServer = <TeamForge application server host name>
```

a) Restart gerrit.

```
/etc/init.d/collabnet stop gerrit
/etc/init.d/collabnet start gerrit
```

Do this on the application server - my.app.host

39. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.
b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.
c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

Do this on the database server - my.db.host

40. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

TeamForge Avatar display issue on RHEL/CentOS 6.7	TeamForge Avatar image is not displayed properly post Review Board installation on RHEL/CentOS 6.7. Run the following commands to work around this issue:

	`yum erase python-imaging`
	`yum install teamforge`
	`/etc/init.d/httpd restart`

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

- **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
- **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
- **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an

iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.

- **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - Oracle database services on a separate server

In this procedure, we assume that you are upgrading on the same server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Black Duck Code Sight Server
- ETL Server
- GIT Integration Server
- Search Server (Indexer).
- SCM Integration Server (Subversion and CVS)

The following service runs on the database server (Oracle) (we call this my.db.host).

- Database Server (Operational DB and Reports DB)

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do this on the Oracle Database Server - my.db.host

1. Make a dump file of your site database.

   To back up the Oracle database, follow the *Oracle backup procedure*.

Do the following on the application server - my.app.host

2. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

3. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

4. Back up the file system data.

   ☞ **Tip:** /tmp in the following step is just an example. You can use any directory or partition that you prefer.

   a) Make an archive file with the following data directories:

Directory	Contents
/opt/collabnet/teamforge/var	User-created data, such as artifact attachments
/svnroot	Subversion source code repositories

Directory	Contents
/sf-svnroot	Subversion repository for branding data
/cvsroot	CVS source code repositories (not present on all sites)

```
mkdir -p /tmp/backup_dir
cp -Rpfv /svnroot /sf-svnroot /cvsroot /opt/collabnet/teamforge/var /
tmp/backup_dir
```

b) If GIT integration is enabled, do the following:

```
mkdir /tmp/backup_dir/gerrit
cp -Rpfv /gitroot /tmp/backup_dir
cp -Rpfv /opt/collabnet/gerrit/ /tmp/backup_dir/gerrit
```

c) Back up your SSH keys, if any.

d) Back up your SSL certificates and keys, if any.

5. If GIT is enabled, create a gerrit database dump file, for PostgreSQL upgrade.

```
su - postgres
/usr/bin/pg_dumpall > /var/lib/pgsql/9.3/backups/
teamforge_gerrit_data_backup.dmp
exit
```

6. Upgrade the operating system packages.

```
yum upgrade
```

7. Create a file, /etc/security/limits.d/teamforge.conf, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```

Save the file and reboot the server.

8. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the getenforce command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

9. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to /tmp.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

10. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

👉 **Tip:** If you have GIT and Black Duck Code Sight already, they are automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-steps.

b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

c) Black Duck Code Sight: To install the Black Duck Code Sight packages run the following command:

```
yum install teamforge-codesearch
```

11. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

> ⚠️ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app etl indexer subversion cvs
```

```
HOST_my.db.domain.com=database datamart
```

```
DOMAIN_localhost=my.app.domain.com
```

e) Add "gerrit" to the HOST_localhost token if you are installing GIT.

```
HOST_localhost=app etl indexer subversion cvs gerrit
```

f) Add "`codesearch`" to the HOST_localhost token if you are installing Black Duck Code Sight.

```
HOST_localhost=app etl indexer subversion cvs gerrit codesearch
```

g) Add "`binary`" to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app etl indexer subversion cvs gerrit codesearch binary
```

h) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

i) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

> ☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

j) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

> ⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

k) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

> ☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

   • To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
   • To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

l) Ensure to set the token *DEDICATED_INSTALL=true*.

m) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

n) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

o) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

p) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

q) Ensure to set the token *SELINUX_SETUP=true*.

r) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.

When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

s) Save the `site-options.conf` file.

12. Download the corresponding version of the Oracle client from *http://www.oracle.com/technetwork/database/features/instant-client/index-097480.html*

```
yum localinstall <path to oracle client rpm>
```

13. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

14. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

☞ **Note:** This process can take a long time for sites with a lot of data.

15. Convert your site data to work with TeamForge 8.2.

☞ **Tip:** Before you kick off the data migration, do the following:

- Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
- If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

16. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

17. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

18. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

19. If you have installed Black Duck Code Sight, then install the license for Black Duck Code Sight. For more information, see *Install the Black Duck Code Sight license* on page 736.

20. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

21. Restart the collabnet services.

```
/etc/init.d/collabnet restart
```

22. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

> ☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

  See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

  See *Communicate with all site users at once* on page 271.

TeamForge Avatar display issue on RHEL/CentOS 6.7	TeamForge Avatar image is not displayed properly post Review Board installation on RHEL/CentOS 6.7. Run the following commands to work around this issue:

```
yum erase python-imaging
yum install teamforge
/etc/init.d/httpd restart
```

• **ETL initial load jobs**: Run the ETL initial load jobs. See *ETL initial load jobs*.

Upgrade from TeamForge 7.1 to 8.2 - All services on the same server

In this procedure, we'll assume that you are upgrading on the same server where your existing TeamForge 7.1 site is running.

> ☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

> ☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

• TeamForge Application Server
• Black Duck Code Sight Server
• Database Server (Operational DB and Reports DB)
• ETL Server
• GIT Integration Server
• SCM Integration Server (Subversion and CVS)
• Search Server (Indexer)

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

> ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

```
cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
custom-install --uninstall
yum erase teamforge_cli_server
```

2. Stop TeamForge.

```
/etc/init.d/collabnet stop all
```

3. Migrate your PostgreSQL to the latest version supported by TeamForge 8.2.

☞ **Important:** You must have RHEL/CentOS 6.5 or later to run PGTurant. Upgrade the operating system packages (`yum upgrade`) before running PGTurant on sites running on RHEL/CentOS versions 6.4 or earlier.

a) Install PGTurant. See *Upgrade PostgreSQL using PGTurant* on page 573 for more information.

☞ **Important:** If your TeamForge site has no internet access, contact the *CollabNet Support*, get the **pgturant-8.0.0.0-7.noarch.rpm** package and unpack the RPM.

```
rpm -ivh pgturant-8.0.0.0-7.noarch.rpm
```

```
yum install pgturant -y
```

b) Upgrade TeamForge PostgreSQL data directory to PostgreSQL 9.3.

```
cd /opt/collabnet/pgturant/bin/
./pgturant -s /var/lib/pgsql/9.2/data -d /var/lib/pgsql/9.3 -u 9.3 -m
```

4. ☞ **Important:** This step is required only if you ran PGTurant. You can skip this step otherwise.

Start the PostgreSQL service, run the `analyze_new_cluster.sh` script and stop PostgreSQL.

```
/etc/init.d/postgresql-9.3 start
su - postgres -c "/var/lib/pgsql/9.3/analyze_new_cluster.sh"
/etc/init.d/postgresql-9.3 stop
```

5. *Click here* only if your reporting database is running on a separate port.

6. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

7. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package.

```
yum erase neon-devel -y
```

b) Upgrade the operating system packages.

```
yum upgrade
```

8. Create a file, `/etc/security/limits.d/teamforge.conf,` and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```

Save the file and reboot the server.

9. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

10. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.2 stop
```

11. 👉   **Important:** Do this only if you are upgrading TeamForge on RHEL/CentOS 6.7.

Delete the `python-crypto` package.

```
yum erase python-crypto
```

12. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) 👉   **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

13. Uninstall the PostgreSQL RPMs.

👉 **Note:** When prompted, you must select the option to remove all the dependent packages.

```
yum erase postgresql92-libs postgresql92-docs postgresql92-server
postgresql92
```

14. Install the following application packages.

   a) TeamForge: To install the TeamForge application packages run the following command:

   ```
   yum install teamforge
   ```

   Run the following command to install the Binary application packages.

   ```
   yum install collabnet-binary
   ```

   👉 **Tip:** If you have GIT and Black Duck Code Sight already, they are automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-steps.

   b) GIT: To install the GIT packages run the following command:

   ```
   yum install teamforge-git
   ```

   c) Black Duck Code Sight: To install the Black Duck Code Sight packages run the following command:

   ```
   yum install teamforge-codesearch
   ```

15. Set up your site's master configuration file.

   Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

   ⚠️ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

   a) Back up your existing `site-options.conf` as `site-options-old.conf`.

   ```
   cd /opt/collabnet/teamforge-installer/8.2.0.0/
   mv conf/site-options.conf conf/site-options-old.conf
   ```

   b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

   ```
   cd /opt/collabnet/teamforge-installer/8.2.0.0/
   cp conf/site-options-dedicated.conf conf/site-options.conf
   ```

   c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

   ```
   sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
   ```

   d) Update the host name and domain name, if required.

   ```
   HOST_localhost=app database datamart etl indexer subversion cvs
   ```

   ```
   DOMAIN_localhost=my.app.domain.com
   ```

   Add 'gerrit' to the HOST_localhost token if you are installing Git.

   ```
   HOST_localhost=app database datamart etl indexer subversion cvs gerrit
   ```

Add 'codesearch' to the HOST_localhost token if you are installing Black Duck Code Sight.

```
HOST_localhost=app database datamart etl indexer subversion
  cvs codesearch
```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs
  codesearch binary
```

e) Add the following database read-only tokens.

-
  ```
  DATABASE_READ_ONLY_USER
  ```

-
  ```
  DATABSE_READ_ONLY_PASSWORD
  ```

f) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

> ☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

g) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

> ⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

h) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

i) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

> ☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: $<>/\'"`

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

j) Ensure to set the token *DEDICATED_INSTALL=true*.

k) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

l) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

m) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.
n) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.
o) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.
p) Ensure to set the token *SELINUX_SETUP=true*.
q) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

r) Save the `site-options.conf` file.

16. As JDK is upgraded to 1.8.0_45, move the existing runtime directory before create-runtime.

```
cd /opt/collabnet/teamforge/
mv runtime/ runtime_old
```

17. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

18. Convert your site data to work with TeamForge 8.2.

> ☞ **Tip:** Before you kick off the data migration, do the following:
> - Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
> - If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

19. Run the following script to upgrade the Subversion working copies.

```
/opt/collabnet/teamforge/runtime/scripts/svn-upgrade-working-copies.sh
```

20. Start TeamForge.

```
/etc/init.d/collabnet start
```

> ☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

21. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

22. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the post-install.py script, see *Post install fails for GIT. What should I do?* on page 921.

23. Install the Black Duck Code Sight license on the server where Black Duck Code Sight is installed. For more information see *these instructions.*

24. Run the svn_cache.sh script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where myint.box is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this svn_cache.sh script, (after creating the new integration server), on the TeamForge application server.

25. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the teamforge-installer/8.2.0.0 directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

26. Remove the backup files, if any, after the TeamForge site is up and running as expected.

TeamForge Avatar display issue on RHEL/CentOS 6.7	TeamForge Avatar image is not displayed properly post Review Board installation on RHEL/CentOS 6.7. Run the following commands to work around this issue: ```yum erase python-imaging\nyum install teamforge\n/etc/init.d/httpd restart```

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

• **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
• **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
• **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
• **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
• **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to ALLOW-FROM http:// forge.collab.net overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
• **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
• **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 on same hardware: RHEL/CentOS 7.2

If your TeamForge 8.1 site runs on a RHEL/CentOS 7.0 server, you can upgrade to TeamForge 8.2 on the same server, which you can upgrade to RHEL/CentOS 7.2 during the upgrade process.

☞ **Remember:** As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, CollabNet recommends you *upgrade Black Duck Code Sight on a separate RHEL/CentOS 6.7 server* (my.codesight.host).

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

Upgrade to TeamForge 8.2 - All services on the same RHEL/CentOS 7.2 server

In this procedure, we'll assume that you are upgrading on the same RHEL/CentOS 7.2 server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

- TeamForge Application Server
- Database Server (Operational DB and Reports DB)
- ETL Server
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer)

☞ **Remember:** As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, CollabNet recommends you *upgrade Black Duck Code Sight on a separate RHEL/CentOS 6.7 server* (my.codesight.host).

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

2. Stop all TeamForge application services.

   ```
   /etc/init.d/collabnet stop all
   ```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

5. Create a file, /etc/security/limits.d/teamforge.conf, and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

8. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

☞ **Tip:** If you have GIT already, it is automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-step.

b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

9. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

⚠ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app database datamart etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.codesight.domain.com=codesearch
```

Add "gerrit" to the HOST_localhost token if you are installing GIT.

```
HOST_localhost=app database datamart etl indexer subversion cvs gerrit
```

Add "binary" to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs
 gerrit binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞　**Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠　**Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞　**Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) Save the `site-options.conf` file.

10. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

11. Convert your site data to work with TeamForge 8.2.

> ☞ **Tip:** Before you kick off the data migration, do the following:
>
> • Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
> • If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

12. Start TeamForge.

```
/etc/init.d/collabnet start
```

> ☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

13. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

14. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

> ☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

Do the following on the Black Duck Code Sight server - my.codesight.host.

15. Install the Black Duck Code Sight license. For more information see *these instructions.*

16. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do the following on the TeamForge Application Server - my.app.host

17. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

18. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

* **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
* **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
* **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
* **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
* **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
* **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
* **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - Database and Datamart on a separate server

In this procedure, we'll assume that you are upgrading on the same RHEL/CentOS 7.2 server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

* TeamForge Application Server
* ETL Server
* GIT Integration Server
* SCM Integration Server (Subversion and CVS)
* Search Server (Indexer)

Both the operational and reports databases run on the database server (we call this my.db.host).

* Database Server (Operational DB and Reports DB)

☞ **Remember:** As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, CollabNet recommends you *upgrade Black Duck Code Sight on a separate RHEL/CentOS 6.7 server* (my.codesight.host).

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

```
cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
custom-install --uninstall
yum erase teamforge_cli_server
```

2. Stop TeamForge.

```
/etc/init.d/collabnet stop all
```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

```
yum upgrade
```

5. Create a file, /etc/security/limits.d/teamforge.conf, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```

Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```

   b) If the output of the getenforce command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

   ```
   setenforce 0
   ```

   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to /tmp.

   b) Install the repository package.

   ```
   yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
   ```

   c) Refresh your repository cache.

   ```
   yum clean all
   ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

   • Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

   ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

   b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

8. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

👉 **Tip:** If you have GIT already, it is automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-step.

b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

9. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

⚠️ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
```

```
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's site-options-old.conf file and make sure that the TeamForge 8.2 site-options.conf file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.db.domain.com=database datamart
```

```
HOST_my.codesight.domain.com=codesearch
```

Add "gerrit" to Host_localhost if you are installing Git.

```
HOST_localhost=app etl indexer subversion cvs gerrit
```

Add "binary" to Host_localhost if you are installing Nexus.

```
HOST_localhost=app etl indexer subversion cvs gerrit binary
```

e) Turn on the SSL for your site by editing the relevant variables in the site-options.conf file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.

When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) Save the `site-options.conf` file.

10. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

11. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

☞ **Note:** This process can take a long time for sites with a lot of data.

Do this on the database server - my.db.host

12. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
yum erase subversion-devel -y
```

b) Upgrade the operating system packages.

```
yum upgrade
```

13. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

14. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

15. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

16. Install the TeamForge database packages.

```
yum install teamforge-database
```

17. Copy the `site-options.conf` file from **my.app.host** and modify the token settings.

> ☞ **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.db.host=database datamart
```

> ☞ **Note:** 'HOST_my.db.host' is just an example. As you are installing database on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.db.host=my.db.domain.com
```

```
HOST_my.app.domain.com=app etl indexer subversion cvs
```

b) Save the `site-options.conf` file.

18. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the application server - my.app.host

19. Convert your site data to work with TeamForge 8.2.

> ☞ **Tip:** Before you kick off the data migration, do the following:
>
> • Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
> • If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

20. Start TeamForge.

```
/etc/init.d/collabnet start
```

> ☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

21. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

22. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

> ☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

Do the following on the Black Duck Code Sight server - my.codesight.host.

23. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do the following on the TeamForge Application Server - my.app.host

24. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

25. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

- **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
- **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
- **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
- **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - Reporting services on a separate server

In this procedure, we'll assume that you are upgrading on the same RHEL/CentOS 7.2 server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- Database Server (Operational DB)
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer)

The following service runs on the database server (we call this my.reports.host).

- Database Server (Reports DB)
- ETL Server

☞ **Remember:** As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, CollabNet recommends you *upgrade Black Duck Code Sight on a separate RHEL/CentOS 6.7 server* (my.codesight.host).

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

2. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

5. Create a file, /etc/security/limits.d/teamforge.conf, and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```
   b) If the output of the getenforce command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

   ```
   setenforce 0
   ```
   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to /tmp.
   b) Install the repository package.

   ```
   yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
   ```
   c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

8. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

9. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

☞ **Tip:** If you have GIT already, it is automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-step.

b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

10. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

⚠️ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app database indexer subversion cvs
```

```
HOST_my.reports.domain.com=datamart etl
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.codesight.domain.com=codesearch
```

Add "gerrit" to `Host_localhost` if you are installing Git.

```
HOST_localhost=app database indexer subversion cvs gerrit
```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database indexer subversion cvs gerrit binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

> ☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

> ⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

> ☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) Save the `site-options.conf` file.

11. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

12. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

☞ **Note:** This process can take a long time for sites with a lot of data.

Do this on the reports server - my.reports.host

13. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
yum erase subversion-devel -y
```

b) Upgrade the operating system packages.

```
yum upgrade
```

14. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

15. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
```

```
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

16. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

17. Install the TeamForge database packages.

```
yum install teamforge-database teamforge-etl
```

18. Copy the `site-options.conf` file from **my.app.host** and modify the token settings.

☞ **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.reports.host=datamart etl
```

☞ **Note:** 'HOST_my.reports.host' is just an example. As you are installing reporting services on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.reports.host=my.reports.domain.com
```

```
HOST_my.app.domain.com=app database indexer subversion cvs
```

b) Save the `site-options.conf` file.

19. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the application server - my.app.host

20. Convert your site data to work with TeamForge 8.2.

☞ **Tip:** Before you kick off the data migration, do the following:
  • Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
  • If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

21. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

22. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

23. Run the TeamForge post installation script. For more information, see *`post-install.py`*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

Do the following on the Black Duck Code Sight server - my.codesight.host.

24. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do the following on the TeamForge Application Server - my.app.host

25. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

   a) Log into your site as the administrator.
   b) If your site has custom branding, verify that your branding changes still work as intended.

   See *Customize anything on your site* on page 290.
   c) Let your site's users know they've been upgraded.

   See *Communicate with all site users at once* on page 271.

26. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

• **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
• **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
• **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
• **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
• **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
• **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
• **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - Black Duck Code Sight on a separate server
In this procedure, we'll assume that you are upgrading on the same RHEL/CentOS 7.2 server where your existing
TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2*
on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you
choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- ETL Server
- Database Server (Operational DB and Reports DB)
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer)

The following service runs on the RHEl/CentOS 6.7 server (we call this my.codesight.host).

- Black Duck Code Sight Server

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have
the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

2. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back
   up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database
   and file system.

4. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

5. Create a file, /etc/security/limits.d/teamforge.conf, and make sure it has the following "nproc"
   and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or
   upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

      ```
      getenforce
      ```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

8. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

9. Install the following application packages.

   a) TeamForge: To install the TeamForge application packages run the following command:

   ```
   yum install teamforge
   ```

   Run the following command to install the Binary application packages.

   ```
   yum install collabnet-binary
   ```

   ☞ **Tip:** If you have GIT already, it is automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-step.

   b) GIT: To install the GIT packages run the following command:

   ```
   yum install teamforge-git
   ```

10. Set up your site's master configuration file.

    Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

    ⚠ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

    a) Back up your existing `site-options.conf` as `site-options-old.conf`.

    ```
    cd /opt/collabnet/teamforge-installer/8.2.0.0/
    mv conf/site-options.conf conf/site-options-old.conf
    ```

    b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

    ```
    cd /opt/collabnet/teamforge-installer/8.2.0.0/
    cp conf/site-options-dedicated.conf conf/site-options.conf
    ```

    c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

    ```
    sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
    ```

    d) Update the host name and domain name, if required.

    ```
    HOST_localhost=app database datamart etl indexer subversion cvs
    ```

    ```
    DOMAIN_localhost=my.app.domain.com
    ```

    ```
    HOST_my.codesight.domain.com=codesearch
    ```

    Add 'gerrit' to the HOST_localhost token if you are installing Git.

    ```
    HOST_localhost=app database datamart etl indexer subversion cvs gerrit
    ```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs
 gerrit binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `` ` ``

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) Save the `site-options.conf` file.

11. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

12. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

> ☞ **Note:** This process can take a long time for sites with a lot of data.

Do this on the Black Duck Code Sight Server - my.codesight.host

13. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
yum erase subversion-devel -y
```

b) Upgrade the operating system packages.

```
yum upgrade
```

14. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

15. ☞ **Important:** Do this only if you are upgrading TeamForge on RHEL/CentOS 6.7.

Delete the `python-crypto` package.

```
yum erase python-crypto
```

16. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

- Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, CodeSearch-installer-2.1.3-1.noarch.rpm from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the /tmp directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

17. Stop the Black Duck Code Sight service.

```
/etc/init.d/collabnet stop tomcatcs
```

18. Install Black Duck Code Sight.

```
yum install teamforge-codesearch
```

19. Copy the master site-options.conf file from my.app.host and modify the token settings.

☞ **Note:** If you choose to use the old site-options.conf file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.codesight.host=codesearch
```

```
DOMAIN_my.codesight.host=my.codesight.domain.com
```

```
Host_my.app.domain.com=app database datamart etl indexer subversion cvs
```

b) Save the `site-options.conf` file.

20. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

21. Start the Black Duck Code Sight service.

```
/etc/init.d/collabnet start tomcatcs
```

22. To install the license for Black Duck Code Sight follow *these instructions.*

Do this on the application server - my.app.host

23. Convert your site data to work with TeamForge 8.2.

☞ **Tip:** Before you kick off the data migration, do the following:

- Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
- If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

24. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

25. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

26. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

Do this on the Black Duck Code Sight server - my.codesight.host

27. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

28. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do this on the application server - my.app.host

29. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

30. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

- **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
- **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
- **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
- **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - GIT on a separate server

In this procedure, we'll assume that you are upgrading on the same RHEL/CentOS 7.2 server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- ETL Server
- Database Server (Operational DB and Reports DB)
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer)

The following service runs on the GIT integration server (we call this my.git.host).

- GIT Integration Server

☞ **Remember:** As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, CollabNet recommends you *upgrade Black Duck Code Sight on a separate RHEL/CentOS 6.7 server* (my.codesight.host).

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

2. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

5. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

      ```
      getenforce
      ```
   b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

      ```
      setenforce 0
      ```

   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.
   b) Install the repository package.

      ```
      yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
      ```
   c) Refresh your repository cache.

      ```
      yum clean all
      ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

8. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

9. To install the TeamForge application packages run the following command.

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

10. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

⚠ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app database datamart etl indexer subversion cvs
```

```
DOMAIN_localhost=my.app.domain.com
```

```
HOST_my.git.domain.com=gerrit
```

```
HOST_my.codesight.domain.com=codesearch
```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost= app database datamart etl indexer subversion cvs binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"`` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.

When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) ☞ **Important:** This step is required if you want Git notification emails.

Update the *JAMES_ACCEPTED_RELAYS* site-options token with the Git server's IP address. See *JAMES_ACCEPTED_RELAYS* on page 794 for more information.

```
JAMES_ACCEPTED_RELAYS=127.0.0.1,{__CEE_DOMAIN__},<The IP address of the
 Git server>
```

r) Save the `site-options.conf` file.

11. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

12. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

☞ **Note:** This process can take a long time for sites with a lot of data.

13. Convert your site data to work with TeamForge 8.2.

☞ **Tip:** Before you kick off the data migration, do the following:

- Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
- If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

14. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

15. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

16. Run the TeamForge post installation script. For more information, see *post-install.py*.

a) Use the following command to run the `post-install.py` script.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** Enter the 'gitadmin' user name and password when prompted.

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

Do the following on the Black Duck Code Sight server - my.codesight.host.

17. Install the Black Duck Code Sight license on the server where Black Duck Code Sight is installed. For more information see *these instructions.*

18. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do this on the Git integration server - my.git.host

19. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

20. Upgrade the operating system packages.

```
yum upgrade
```

21. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

22. Configure your TeamForge installation repository.

> **TeamForge installation repository configuration for sites with internet access**

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

> **TeamForge installation repository configuration for sites without internet access**

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

   • Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

   > ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

23. Install the Git packages.

```
yum install teamforge-git
```

24. Copy the `site-options.conf` file from my.app.host and modify the token settings.

☞ **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.git.host=gerrit
```

```
DOMAIN_my.git.host=my.git.domain.com
```

```
HOST_my.app.domain.com=app database datamart etl indexer subversion cvs
```

b) Save the `site-options.conf` file.

25. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

26. Run the TeamForge post installation script. For more information, see *post-install.py*.

a) Use the following command to run the `post-install.py` script.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** Enter the 'gitadmin' user name and password when prompted.

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

27. ☞ **Important:** This step is required if you want Git notification emails.

Edit the `/opt/collabnet/gerrit/etc/gerrit.config` file and update the 'smtpServer' property with the TeamForge application server's host name.

For example:

```
[sendemail]
               smtpServer = <TeamForge application server host name>
```

a) Restart gerrit.

```
/etc/init.d/collabnet stop gerrit
/etc/init.d/collabnet start gerrit
```

Do the following on the application server - my.app.host

28. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

29. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

☞ **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

• **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.

• **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.

- **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://` `forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
- **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 - Database and SCM on separate servers

In this procedure, we'll assume that you are upgrading on the same RHEL/CentOS 7.2 server where your existing TeamForge 8.1 site is running.

☞ **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- ETL Server
- Search Server (Indexer).

The following service runs on the database server (we call this my.db.host).

- Database Server (Operational DB and Reports DB)

The following services run of the SCM server (we call this my.scmandgit.host).

- SCM Integration Server (Subversion and CVS)
- GIT Integration Server

☞ **Remember:** As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, CollabNet recommends you *upgrade Black Duck Code Sight on a separate RHEL/CentOS 6.7 server* (my.codesight.host).

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do the following on the application server - my.app.host

1. Uninstall the TeamForge CLI add-on.

   ☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

   ```
   cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
   custom-install --uninstall
   yum erase teamforge_cli_server
   ```

2. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

3. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

4. Upgrade the operating system packages.

   ```
   yum upgrade
   ```

5. Create a file, `/etc/security/limits.d/teamforge.conf`, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```

Save the file and reboot the server.

6. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

7. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

8. To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

9. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

> ⚠️ **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

d) Update the host name and domain name, if required.

```
HOST_localhost=app etl indexer
```

```
HOST_my.db.domain.com=database datamart
```

```
HOST_my.scmandgit.domain.com=subversion cvs gerrit
```

```
HOST_my.codesight.domain.com=codesearch
```

```
DOMAIN_localhost=my.app.domain.com
```

Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app etl indexer binary
```

e) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

> ☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

f) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

> ⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

g) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

h) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

> ☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: `$<>/\'"` `

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

i) Ensure to set the token *DEDICATED_INSTALL=true*.

j) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

k) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

l) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

m) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

n) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

o) Ensure to set the token *SELINUX_SETUP=true*.

p) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

> ☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.
>
> When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

q) ☞ **Important:** This step is required if you want Git notification emails.

Update the *JAMES_ACCEPTED_RELAYS* site-options token with the Git server's IP address. See *JAMES_ACCEPTED_RELAYS* on page 794 for more information.

```
JAMES_ACCEPTED_RELAYS=127.0.0.1,{__CEE_DOMAIN__},<The IP address of the
 Git server>
```

r) Save the `site-options.conf` file.

10. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

11. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

> ☞ **Note:** This process can take a long time for sites with a lot of data.

Do the following on the database server - my.db.host

12. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
yum erase subversion-devel -y
```

b) Upgrade the operating system packages.

```
yum upgrade
```

13. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

14. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

15. If the PostgreSQL database is running locally, stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.3 stop
```

16. Install the TeamForge database packages.

```
yum install teamforge-database
```

17. Copy the `site-options.conf` file from my.app.host and modify the token settings.

☞ **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

a) Set the following tokens.

```
HOST_my.db.host=database datamart
```

☞ **Note:** 'HOST_my.db.host' is just an example. As you are installing database on a separate server, do not use 'HOST_localhost'. Use 'HOST_<valid host name>' instead.

```
DOMAIN_my.db.host=my.db.domain.com
```

```
HOST_my.app.domain.com=app etl indexer
```

```
HOST_my.scmandgit.domain.com=subversion cvs gerrit
```

```
HOST_my.codesight.domain.com=codesearch
```

b) Save the `site-options.conf` file.

18. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the application server - my.app.host

19. Convert your site data to work with TeamForge 8.2.

☞ **Tip:** Before you kick off the data migration, do the following:

- Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
- If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

20. Start TeamForge.

```
/etc/init.d/collabnet start
```

21. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

22. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

Do the following on the Black Duck Code Sight server - my.codesight.host.

23. Install the Black Duck Code Sight license on the server where Black Duck Code Sight is installed. For more information see *these instructions.*

24. Run the svn_cache.sh script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where myint.box is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this svn_cache.sh script, (after creating the new integration server), on the TeamForge application server.

Do the following on the SCM server - my.scmandgit.host

If your TeamForge setup includes source control running on its own server, you'll have to upgrade that server as well as the main TeamForge application server.

25. Stop TeamForge.

```
/etc/init.d/httpd stop
/etc/init.d/collabnet stop tomcat
```

26. No backup is required for same hardware upgrades. However, you can create a backup as a precaution. See *Back up and restore CollabNet TeamForge data* on page 562 to learn more about backing up TeamForge database and file system.

27. Run the following commands to upgrade Red Hat/CentOS to the latest version.

a) Remove the neon-devel package if you are upgrading from TeamForge 6.2.

```
yum erase neon-devel -y
yum erase subversion-devel -y
```

b) Upgrade the operating system packages.

```
yum upgrade
```

28. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the getenforce command is either "Disabled" or "Permissive", SELinux is already disabled.

c) If not disabled, run the following command to disable SELinux.

```
setenforce 0
```

29. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to /tmp.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

• Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

> ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

30. If Git is enabled and the PostgreSQL database is running locally:

a) Stop the PostgreSQL service.

```
/etc/init.d/postgresql-9.2 stop
```

31. Install the following application packages.

a) Install the source code component.

```
yum install teamforge-scm
```

b) To install the Git packages, run the following command.

```
yum install teamforge-git
```

32. Copy the master `site-options.conf` file from my.app.host and modify the host token settings in the `site-options.conf` file.

☞ **Note:** If you choose to use the old `site-options.conf` file, don't forget to copy the *AUTO_DATA* token from the application server.

```
HOST_my.scmandgit.host=subversion cvs gerrit
```

```
DOMAIN_my.scmandgit.host=my.scmandgit.domain.com
```

```
HOST_my.app.domain.com=app etl indexer
```

```
HOST_my.db.domain.com=database datamart
```

```
HOST_my.codesight.domain.com=codesearch
```

33. Save the `site-options.conf`file.

34. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

35. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

☞ **Note:** This process can take a long time for sites with a lot of data.

36. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

37. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

38. ☞ **Important:** This step is required if you want Git notification emails.

Edit the `/opt/collabnet/gerrit/etc/gerrit.config` file and update the 'smtpServer' property with the TeamForge application server's host name.

For example:

```
[sendemail]
               smtpServer = <TeamForge application server host name>
```

a) Restart gerrit.

```
/etc/init.d/collabnet stop gerrit
/etc/init.d/collabnet start gerrit
```

Do this on the application server - my.app.host

39. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

   c) Let your site's users know they've been upgraded.

     See *Communicate with all site users at once* on page 271.

Do this on the database server - my.db.host

40. Remove the backup files, if any, after the TeamForge site is up and running as expected. Remove the repository and the file system backup from the `/tmp/backup_dir` directory.

☞   **Note:** After upgrade, it takes some time for the publishing repositories to get created for projects imported from other TeamForge sites.

- **ETL initial load jobs**: To run ETL initial load jobs, see *ETL initial load jobs*.
- **Review Board**: To upgrade Review Board, see *Upgrade Review Board* on page 722.
- **SELinux**: To run TeamForge in SElinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Configure X-Frame-Options**: Setting the **X-Frame-Options** header to `ALLOW-FROM http://forge.collab.net` overrides the default Jenkins configuration header settings and displays the page in an iframe on the specified origin. For more information, see *Unable to embed Jenkins page into an iframe. How to fix it?* on page 928.
- **Internal code browser**: To turn on TeamForge code browser for a Subversion or Git repository, see *Internal code browser*.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade to TeamForge 8.2 with Oracle database services on a separate server

In this procedure, we assume that you are upgrading on the same RHEL/CentOS 7.2 server where your existing TeamForge 8.1 site is running.

☞   **Important:** See *Plan your upgrade* on page 441 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞   **Important:** TeamForge 8.2 supports RHEL/CentOS 7.2. However, it also supports RHEL/CentOS 6.7 if you choose to upgrade on the same server that runs RHEL/CentOS 6.7.

In this option, the following services run on the application server (we call this my.app.host).

- TeamForge Application Server
- ETL Server
- GIT Integration Server
- Search Server (Indexer).
- SCM Integration Server (Subversion and CVS)

The following service runs on the database server (Oracle) (we call this my.db.host).

- Database Server (Operational DB and Reports DB)

☞   **Remember:** As Black Duck Code Sight 2.1.3 has no support for RHEL/CentOS 7.x, CollabNet recommends you *upgrade Black Duck Code Sight on a separate RHEL/CentOS 6.7 server* (my.codesight.host).

☞   **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

Log on to the server as root user always.

Do this on the Oracle Database Server - my.db.host

1. Make a dump file of your site database.

To back up the Oracle database, follow the *Oracle backup procedure*.

Do the following on the application server - my.app.host

2. Uninstall the TeamForge CLI add-on.

☞ **Important:** You must have no tasks in progress while uninstalling the CLI add-ons.

```
cd /opt/collabnet/teamforge/add-ons/teamforge_cli_server
custom-install --uninstall
yum erase teamforge_cli_server
```

3. Stop TeamForge.

```
/etc/init.d/collabnet stop all
```

4. Back up the file system data.

☞ **Tip:** /tmp in the following step is just an example. You can use any directory or partition that you prefer.

a) Make an archive file with the following data directories:

Directory	Contents
/opt/collabnet/teamforge/var	User-created data, such as artifact attachments
/svnroot	Subversion source code repositories.
/sf-svnroot	Subversion repository for branding data
/cvsroot	CVS source code repositories (not present on all sites)

```
mkdir -p /tmp/backup_dir
cp -Rpfv /svnroot /sf-svnroot /cvsroot /opt/collabnet/teamforge/var /
tmp/backup_dir
```

b) If GIT integration is enabled, do the following:

```
mkdir /tmp/backup_dir/gerrit
cp -Rpfv /gitroot /tmp/backup_dir
cp -Rpfv /opt/collabnet/gerrit/ /tmp/backup_dir/gerrit
```

c) Back up your SSH keys, if any.
d) Back up your SSL certificates and keys, if any.

5. If GIT is enabled, create a gerrit database dump file, for PostgreSQL upgrade.

```
su - postgres
/usr/bin/pg_dumpall > /var/lib/pgsql/9.3/backups/
teamforge_gerrit_data_backup.dmp
exit
```

6. Upgrade the operating system packages.

```
yum upgrade
```

7. Create a file, /etc/security/limits.d/teamforge.conf, and make sure it has the following "nproc" and "nofile" settings.

```
vim /etc/security/limits.d/teamforge.conf
sf-admin soft nproc 10240
sf-admin hard nproc 10240
sf-admin hard nofile 65536
sf-admin soft nofile 65536
```
Save the file and reboot the server.

8. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

a) Verify if SELinux is running in enforcing mode.

```
getenforce
```

b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

```
setenforce 0
```

See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

9. Configure your TeamForge installation repository.

TeamForge installation repository configuration for sites with internet access

a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

c) Refresh your repository cache.

```
yum clean all
```

TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

    f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

10. Install the following application packages.

    a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

Run the following command to install the Binary application packages.

```
yum install collabnet-binary
```

> ☞   **Tip:** If you have GIT already, it is automatically upgraded when you run `yum install teamforge`. In such case, you can skip the following sub-step.

    b) GIT: To install the GIT packages run the following command:

```
yum install teamforge-git
```

11. Set up your site's master configuration file.

Always use the `site-options-dedicated.conf` file that's packaged with the TeamForge version you are upgrading to. Refer to the TeamForge 8.1 `site-options.conf` file, understand your site settings and set up your TeamForge 8.2 `site-options.conf` file.

> ⚠   **Caution:** You run the risk of missing out on newly added site option tokens if you keep reusing your `site-options.conf` file from an older TeamForge version on newer versions. See the *New features in CollabNet TeamForge 8.2* on page 1028 section of the Release Notes to know what's new with `site-options.conf` tokens.

    a) Back up your existing `site-options.conf` as `site-options-old.conf`.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
mv conf/site-options.conf conf/site-options-old.conf
```

    b) Copy the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file to the TeamForge installer directory.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0/
cp conf/site-options-dedicated.conf conf/site-options.conf
```

    c) **Set up your site options**: In addition to the new site options you may set up for TeamForge 8.2, refer to the TeamForge 8.1's `site-options-old.conf` file and make sure that the TeamForge 8.2 `site-options.conf` file **inherits** all the required site options.

```
sudo vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

    d) Update the host name and domain name, if required.

```
HOST_localhost=app etl indexer subversion cvs
```

```
HOST_my.db.domain.com=database datamart
```

```
HOST_my.codesight.domain.com=codesearch
```

```
DOMAIN_localhost=my.app.domain.com
```

    e) Add "gerrit" to the HOST_localhost token if you are installing GIT.

```
HOST_localhost=app etl indexer subversion cvs gerrit
```

f) Add **"binary"** to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app etl indexer subversion cvs gerrit binary
```

g) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

h) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

i) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

j) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: $<>/\'"`

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

k) Ensure to set the token *DEDICATED_INSTALL=true*.

l) Include the list of non-expiring TeamForge user accounts (password never expires).

```
USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin
```

m) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

n) Configure the Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

o) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

p) Ensure to set the token *SELINUX_SETUP=true*.

q) Configure the JBOSS_JAVA_OPTS `site-options.conf` token. See *JBOSS_JAVA_OPTS* on page 795.

☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other

parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.

When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

r) Save the `site-options.conf` file.

12. Download the corresponding version of the Oracle client from *http://www.oracle.com/technetwork/database/features/instant-client/index-097480.html*

```
yum localinstall <path to oracle client rpm>
```

13. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

14. Update the file permissions on your site's data.

```
/opt/collabnet/teamforge/runtime/scripts/fix_data_permissions.sh
```

☞ **Note:** This process can take a long time for sites with a lot of data.

15. Convert your site data to work with TeamForge 8.2.

☞ **Tip:** Before you kick off the data migration, do the following:

- Use the `/etc/init.d/collabnet status` command to make sure the JBoss, Tomcat and Tomcatcs services are stopped.
- If any of the trackers in your project already contains a field with the name `team`, conflict may arise while running the `migrate.py` script. This is because of the inclusion of the Team feature in TeamForge 8.0. To prevent this conflict, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

The `migrate.py` script locates the existing site data and modifies it as needed.

This includes configuration data for LDAP and the James mail server. Any modifications that you have applied to these components on your old site are reproduced on your upgraded TeamForge 8.2 site.

16. Start TeamForge.

```
/etc/init.d/collabnet start
```

☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

17. If you have CVS integrations, synchronize permissions post upgrade. See, *Synchronize TeamForge source control integrations* on page 572.

18. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

Do the following on the Black Duck Code Sight server - my.codesight.host.

19. If you have installed Black Duck Code Sight, then install the license for Black Duck Code Sight. For more information, see *Install the Black Duck Code Sight license* on page 736.

20. Run the `svn_cache.sh` script.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./svn_cache.sh <Repository Base URL Path of the SCM Integration Server>
```

Provide a repository base URL path of the SCM integration server, for example, "http://myint.box.net/svn/repos", where `myint.box` is the server with the SCM integration server.

In addition, if you add a new integration server at some point later, you must run this `svn_cache.sh` script, (after creating the new integration server), on the TeamForge application server.

Do the following on the application server - my.app.host

21. Restart the collabnet services.

```
/etc/init.d/collabnet restart
```

22. Apply the finishing touches and make sure everything is running smoothly after upgrading to TeamForge 8.2.

☞ **Important:** Do not delete the `teamforge-installer/8.2.0.0` directory. You will need it for future maintenance and upgrades.

a) Log into your site as the administrator.

b) If your site has custom branding, verify that your branding changes still work as intended.

See *Customize anything on your site* on page 290.

c) Let your site's users know they've been upgraded.

See *Communicate with all site users at once* on page 271.

• **ETL initial load jobs**: Run the ETL initial load jobs. See *ETL initial load jobs*.

Upgrade to CollabNet TeamForge 8.2 on a virtual machine

You can upgrade to CollabNet TeamForge 8.2 from a VMware installation of CollabNet TeamForge 8.1.

☞ **Note:**

• To upgrade from any version earlier than 8.1 (including CollabNet SourceForge Enterprise 5.x or any version of SourceForge Enterprise Edition), first *upgrade your site to TeamForge 8.1* and start it up. Then follow these steps for upgrading to CollabNet TeamForge 8.2

• When you upgrade to TeamForge 8.2, the VMware updater automatically upgrades your site to the latest patch release, if available.

For basic installation and configuration on VMware, see these instructions:

• *Get TeamForge 8.2 for VMware Player or VMware ESXi*

• *Configure CollabNet TeamForge on VMware Player*

1. On the computer where your previous version of TeamForge is running, open a command prompt and log onto the machine as an administrator.

☞ **Note:** Type `N` when prompted to start the TeamForge configuration tool.

2. Get the VMware updater for TeamForge 8.2 from your CollabNet Support representative and unzip it in the `/tmp` directory.

```
unzip updater-teamforge-vmware-8.2.0.0.zip
```

3. Back up your TeamForge 8.1 site.

See *Back up CollabNet TeamForge data* on page 562 for instructions.

4. Review the key configuration variables for your TeamForge site.

The configuration file is at `/opt/collabnet/teamforge/runtime/conf/runtime-options.conf`. Note these key variables and their values:

```
HOST_<host_name>
DOMAIN_<host_name>
SYSTEM_EMAIL
ADMIN_EMAIL
JAMES_POSTMASTER_EMAIL
JAMES_GATEWAY_HOST
JAMES_GATEWAY_PORT
```

5. Run the upgrade utility.

```
cd updater-teamforge-vmware-8.2.0.0
./updater-8_1-8_2.sh
```

The upgrade can take several minutes to complete, depending on system resources.

☞ **Important:** Type Y when prompted.

Configure the CLI Jobs Linked Application's URL

☞ **Important:** Configure the CLI Jobs linked application's URL only if the site option token *HOST_localhost* is configured as "localhost" in the `site-options.conf` file.

6. After logging on to TeamForge, click the **View All Projects** option under **My Workspace**.

7. Click the **Look** project.

8. Click **PROJECT ADMIN** from the **Project Home** menu.

9. Click **Tools** from the **Project Admin Menu**.

10. Click the **CLI Jobs** linked application from the **Project Tools** list..

11. On the **Edit Tool** page, modify the URL: Replace the "localhost" with the hostname or IP address of the computer that hosts the TeamForge application.

12. Click **Save**.

• **ETL initial load jobs**: To run ETL initial load jobs, see*ETL initial load jobs*.

Troubleshooting: VMware installer failed to install new TeamForge version
When your TeamForge 8.2 upgrade process on VMware fails during the installation phase, something may be blocking the installer or the data migration utility from running TeamForge from your VMware Player. Try these manual steps to recover.

Execute these steps from a command line terminal inside the VMware Player.

1. Ensure that the collabnet-8.2.0.0.repo file is available in the /etc/yum.repos.d

2. Install the TeamForge application packages.

```
yum install teamforge
```

3. Create the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -n -V -d /opt/collabnet/teamforge -F
```

4. Run the data migration utility.

☞ **Tip:** Before you kick off the data migration, you must identify if any of the trackers in your project already contains a field with the name team because if it does, conflict may arise while running the migrate.py script. This is because of the inclusion of the Team feature in TeamForge 8.0. To identify and modify the existing team name, follow the steps given in *FAQ on troubleshooting*.

```
/opt/collabnet/teamforge/runtime/scripts/migrate.py
```

5. Start TeamForge.

```
/etc/init.d/collabnet start
```

Troubleshooting: VMware installer failed to finish TeamForge 8.2 installation
When your TeamForge 8.2 upgrade process on VMware fails during the post-install phase, it may have been unable to configure the installation properly. Try these manual steps to recover.

Execute these steps from a command line terminal inside the VMware player.

1. Set the permissions for the branding repository.

```
chown -R apache:apache /opt/collabnet/teamforge/var/overrides
chown -R apache:apache /sf-svnroot
```

2. Replace the automatic configuration script.

```
cd $UPDATE_DIR (UPDATE_DIR=location where vmware updater is extracted)
cp -f configure-teamforge.py /root/configure-teamforge.py
```

3. Replace the static HTML content.

```
cd $UPDATE_DIR
'cp' $UPDATE_DIR/static-content/index.html /var/www/html
'cp' $UPDATE_DIR/static-content/logo.gif /var/www/html/static_files/
'cp' $UPDATE_DIR/static-content/static.htm /var/www/html
'cp' $UPDATE_DIR/static-content/c.css /var/www/html/static_files/
'cp' $UPDATE_DIR/static-content/initial_tasks.png /var/www/html/_i/
'cp' $UPDATE_DIR/static-content/launch.gif /var/www/html/_i
```

Troubleshooting: Problem with importing TeamForge .ova file on VMware Player running on Linux

If you face trouble while importing the TeamForge .ova file on VMware Player running on Linux, it is recommended to import using the VMware Open Virtualization Format Tool.

1. Download the VMware Open Virtualization Format Tool from the following link. https://my.vmware.com/web/vmware/details?downloadGroup=OVFTOOL352&productId=353

2. Install the VMware Open Virtualization Format Tool.

3. Use the following commands to import the TeamForge .ova file.

```
mkdir <teamforge_vm_path>
```

For example, `mkdir /root/teamforge-82`

```
ovftool --lax <path to teamforge-82.ova> <teamforge_vm_path>/
teamforge-82.vmx
```

4. Open the `teamforge-82.vmx` file using VMware Player.

Back up and restore CollabNet TeamForge data

CollabNet TeamForge stores data in the database and on the file system. Back up all data comprehensively so that it can be restored in the event of unrecoverable failures.

☞ **Note:** The items listed in this section address only the data that is either created by or a part of CollabNet TeamForge. Data that is not specific to TeamForge, such as operating system-based content, configuration files, and other third-party applications, will also require a backup and restoration routine to ensure that the entire server can be restored in the event of a catastrophic failure. Contact your application or operating system vendor for specific guidance on backup strategies for their products.

Back up CollabNet TeamForge data

Save a copy of your TeamForge site's data to a location from where you can quickly retrieve it to your TeamForge 8.2 site.

☞ **Important:** The following instructions, though applicable in general to any PostgreSQL version, assume that you run a TeamForge version that runs on PostgreSQL 9.2, which you want to back up. Replace 9.2 with the PostgreSQL version you run, if required.

- If you are upgrading by installing TeamForge 8.2 on new hardware, then you'll need the backed-up site data to complete the upgrade.
- If you are upgrading your site on the same hardware, then you won't need to back up but you should create a backup anyway, as a precaution.

1. Stop the TeamForge application server and the Apache server, if they are running.

RHEL/CentOS: `/etc/init.d/httpd stop`
SUSE: `/etc/init.d/apache2 stop`
`/etc/init.d/collabnet stop`

2. Back up your site data.

   a) Make a dump file of your site database.

   (Do this on your database server in case you have your DB running on a separate server.)

   You have to do a PostgreSQL dump if you are upgrading the PostgreSQL application as part of your TeamForge upgrade.

   ☞ **Note:** These commands are for a PostgreSQL database, which is the default. If your site uses an Oracle database, follow the *Oracle backup procedure* instead.

   ```
   su - postgres
   /usr/bin/pg_dumpall > /var/lib/pgsql/9.2/backups/
   teamforge_data_backup.dmp
   exit
   mkdir /tmp/backup_dir
   cp /var/lib/pgsql/9.2/backups/teamforge_data_backup.dmp /tmp/
   backup_dir/
   ```

   ☞ **Note:** If your reporting database is running on a separate port, back up your reporting database too:

   ```
   /usr/bin/pg_dumpall -p <reports_database_port> > /var/lib/
   pgsql/9.2/backups/teamforge_reporting_data_backup.dmp
   ```

   b) Make an archive file with the following data directories:

Directory	Contents
/opt/collabnet/teamforge/var	User-created data, such as artifact attachments
/opt/collabnet/reviewboard	Review Board data
/opt/collabnet/codesearch	Black Duck Code Sight (BDCS) data, in case you have BDCS installed
/svnroot	Subversion source code repositories
/sf-svnroot	Subversion repository for branding data
/cvsroot	CVS source code repositories (not present on all sites)

   ```
   cp -Rpf /svnroot /sf-svnroot /cvsroot /opt/collabnet/teamforge/var /
   opt/collabnet/reviewboard /opt/collabnet/codesearch /tmp/backup_dir
   cd /tmp
   tar czvf backup.tgz backup_dir
   ```

   If GIT integration is enabled, do the following:

   ```
   cp -Rpfv /gitroot /tmp/backup_dir
   cp -Rpfv /opt/collabnet/gerrit/ /tmp/backup_dir/gerrit
   ```

   Compress your backup data.

   ```
   cd /tmp
   tar czvf backup.tgz backup_dir
   ```

   c) Back up your SSH keys, if any.

   d) Back up your SSL certificates and keys, if any.

Back up CollabNet TeamForge data using `backup-data.py`

Use the `backup-data.py` utility to compress a copy of your site data to a location where you can quickly retrieve it.

This backup method requires shutting down your site briefly. If you cannot tolerate a shutdown, you might consider another backup/restore method, such as the NetApp `Snapshot` utility.

1. Stop TeamForge.

   ☞   **Note:** If TeamForge is running on multiple servers, stop all the servers.

   ```
   /etc/init.d/collabnet stop
   ```

2. Run the backup script.

   ```
   cd /opt/collabnet/teamforge/runtime/scripts
   ./backup-data.py --destination=<directory name>
   ```

   CollabNet TeamForge creates the directory and stores the following data in it, in compressed format:

   • Subversion repositories
   • CVS repositories
   • The data directory (`/var`)
   • The CollabNet TeamForge database.

3. Start TeamForge.

   ```
   /etc/init.d/collabnet start
   ```

Restore backed-up CollabNet TeamForge data
Use the `restore-data.py` utility to bring back site data that has been backed up.

1. Stop the CollabNet TeamForge application server.

   ```
   /etc/init.d/collabnet stop
   ```

2. Run the restore script.

   ```
   cd /opt/collabnet/teamforge/runtime/scripts
   ./restore-data.py --source=<directory name>
   ```

   Where, `<directory-name>` is the directory to which you backed up the data with the `backup-data.py` script.

   CollabNet TeamForge unpacks the backed-up Subversion and CVS repositories, the data directory, and the CollabNet TeamForge database.

3. Restart CollabNet TeamForge .

   ```
   /etc/init.d/collabnet start
   ```

Back up a PostgreSQL database
To back up a PostgreSQL database, use the `db.py` script or the `pg_dump` command depending on your setup.

   ☞   **Note:** You can use `db.py` to back up PostgreSQL data in all setups except the dedicated database server setup where `ctf` (TeamForge database) is not installed. In a dedicated database setup, you can back up the PostgreSQL database safely while it is online by using the native `pg_dump` command.

Using `db.py`:
• To back up the entire PostgreSQL database, run this command.

   ```
   ./db.py -a dump -f /tmp/backup/
   ```
• To back up only the `ctf` database, run this command.

   ```
   ./db.py -a dump -t ctf -f /tmp/backup/
   ```
• To back up only the datamart, run this command.

   ```
   ./db.py -a dump -t reporting -f /tmp/backup/
   ```

Using `pg_dump`:
• In a dedicated database setup, you can run one of these `pg_dump` commands:

In this example, the database is dumped into a GNU tar formatted file or a `.dmp` file.

- `pg_dump -Ft -b -o ctfdb > ctfdb.tar`
- `pg_dump -Fc <dbname> -f ctfdb.dmp`
- `pg_dump ctfdb > /tmp/backup_dir/teamforge_data_backup.dmp`

☞ **Note:** For this example, the name of the CollabNet TeamForge database is assumed to be ctfdb.

☞ **Tip:** See the PostgreSQL `pg_dump` man page for more information and examples.

Restore a PostgreSQL database

You can restore a PostgreSQL database with the native `pg_restore` or `psql` commands, or by using the `db.py` script.

☞ **Note:** You can use the `db.py` script to restore PostgreSQL data in all setups except the dedicated database server setup where the TeamForge database is not installed. In a dedicated database setup, you can restore the PostgreSQL database using the `pg_restore` or `psql` command.

Using **db.py**:

- To restore the entire PostgreSQL database, run this command.

      ./db.py -a restore -f /tmp/backup/

- To restore only the `ctf` database, run this command.

      ./db.py -a restore -t ctf -f /tmp/backup/

- To restore only the datamart, run this command.

      ./db.py -a restore -t reporting -f /tmp/backup/

Using **pg_restore** or **psql:**

- In a dedicated database setup, you can use `pg_restore` or `psql`:

  a) Locate the dump file you created when backing up the PostgreSQL database.
  b) Shut down CollabNet TeamForge.
  c) Create a database and user with the names used for CollabNet TeamForge .

      createuser -U $CTFUSER createdb -E UNICODE -U $CTFUSER ctfdb

  d) Restore the database using either the `pg_restore` or `psql` utility. Run one of these commands:

    - `pg_restore -d ctfdb ctfdb.tar`
    - `pg_restore -d <dbname> ctfdb.dmp`
    - `psql ctfdb < /tmp/backup_dir/teamforge_data_backup.dmp`

    ☞ **Note:** This example assumes that the name of the TeamForge database is `ctfdb`.

    ☞ **Tip:** It may also be necessary to restore ownership of the restored tables to the `ctfuser` database user. The following will work again (assuming the database is called `ctfdb`):

        for i in `echo "\d" | psql ctfdb | awk {'print $3'}` do echo
          "ALTER TABLE $i OWNER TO $SFUSER;" | psql ctfdb done

Back up CollabNet TeamForge data with NetApp

If your data is stored on a NetApp volume, you can use the NetApp `snap` utility to establish a known good data point.

☞ **Remember:** If your database is also on the NetApp device, the database files will be part of this snapshot.

1. Shut down the PostgreSQL database.
2. Stop the CollabNet TeamForge application server.

```
/etc/init.d/collabnet stop all
```

3. Provide a name to the snapshot.

The data up to the given time will be snapped and stored.

4. Restart CollabNet TeamForge .

```
/etc/init.d/collabnet start all
```

Restore CollabNet TeamForge data with NetApp

Use the Network Appliance snap utility to "roll back" to a known good data point on a NetApp volume.

☞ **Note:** You cannot snap and restore a qtree. You must snap and restore the entire NetApp volume.

⚠ **Caution:** When you snap and restore to a particular snapshot, all data between the present time and that snapshot is lost.

1. Shut down the PostgreSQL database.

2. Stop the CollabNet TeamForge application server.

```
/etc/init.d/collabnet stop all
```

3. Using the NetApp snap utility, select the snap to be restored.

4. Restart the site.

```
/etc/init.d/httpd start
/etc/init.d/postgresql-9.3 start
/etc/init.d/collabnet start
```

Uninstall TeamForge 8.2

To remove TeamForge completely, use the YUM utility.

☞ **Important:** This procedure removes the TeamForge and all associated databases, including your site data. Be sure to back up any data you want to keep.

1. Stop TeamForge.

```
/etc/init.d/collabnet stop all
```

2. Run yum to remove TeamForge.

```
yum erase TeamForge-installer
```

For every box in a multi-box site, use the same steps to uninstall.

Maintain your TeamForge 8.2 site

You've got CollabNet TeamForge installed, up to date, and operating on the appropriate scale. Now you're in day-to-day maintenance mode. While you're waiting around for something to go wrong, there's a lot you can do to support users and project managers, monitor the site's performance, and generally keep things running smoothly.

Installation and upgrade

You can follow these useful best practices for your TeamForge install and upgrade procedures.

Install project templates manually

Provide sample projects to help users get started quickly.

TeamForge comes with a sample template useful for agile development projects. Site administrators and project managers can use this template to jump-start a project without a lot of manual setup steps.

In the TeamForge installation directory, run the install-project-templates.py script.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
```

```
./install-project-templates.py -V
```

Use a site administrator user name and password. For a new site, these are `admin` and `admin`.

> ☞ **Tip:** On some servers, it may take a few seconds for the SOAP server to be ready after installation. If `install-project-templates.py` returns an error, try waiting briefly and then running it again.

Install project templates using the installer

With TeamForge 8.2, you can install project templates automatically.

TeamForge comes with a sample template useful for agile development projects. Site administrators and project managers can use this template to jump-start a project without a lot of manual setup steps.

* To install project templates automatically when the TeamForge startup script is executed, you must set these tokens:

  *
    ```
    INSTALL_TEMPLATES = true
    ```

  *
    ```
    REQUIRE_USER_PASSWORD_CHANGE = false
    ```

> ☞ **Note:**
>
> * The TeamForge startup script installs the sample TeamForge project templates by default.
> * If the project templates are already installed, you cannot re-install them using the TeamForge startup script.
> * You may choose to delete the sample project templates. After deleting the sample project templates, you must set the *INSTALL_TEMPLATES* site options token to false. Otherwise, the project templates, if not found in the database, are installed automatically every time you restart the CollabNet services.

Set up an Oracle database

To use an Oracle database for your TeamForge data, set up the Oracle database and tell the installer how to handle it.

TeamForge database setup

1. Make sure your database uses UTF8 or AL32UTF8 encoding.

   This is needed to support users in Asian languages.

   For information about discovering and changing the database encoding, see *this Oracle knowledge base article*.

2. Connect to your Oracle database.

   ```
   SQL> connect <adminusername>@<db_name>/<adminpassword> as sysdba
   ```

3. Create the database user and password you will use to connect from CollabNet TeamForge to Oracle.

   ```
   SQL> create user <sf user> identified by <sf passwd> default tablespace
   <your tablespace> temporary tablespace <temporary tablespace>;

   User created.
   ```

4. Grant permissions to the user that you just created.

   ```
   SQL> grant unlimited tablespace to <sf user>;
   SQL> grant create snapshot to <sf user>;
   SQL> grant create cluster to <sf user>;
   SQL> grant create database link to <sf user>;
   SQL> grant create procedure to <sf user>;
   SQL> grant create sequence to <sf user>;
   SQL> grant create trigger to <sf user>;
   SQL> grant create type to <sf user>;
   SQL> grant create view to <sf user>;
   SQL> grant query rewrite to <sf user>;
   SQL> grant alter session to <sf user>;
   ```

```
SQL> grant create table to <sf user>;
SQL> grant create session to <sf user>;
SQL> grant create any synonym to <sf user>;
SQL> exit
```

5. Create the database read-only user that you will use to connect from CollabNet TeamForge.

```
SQL> create user <database_readonly_user> identified by
<database_readonly_password> default tablespace <your tablespace>
temporary tablespace <temporary tablespace>;
```

6. Grant the required permissions to the read-only user that you just created.

```
SQL> grant create session to <database_readonly_user>;
SQL> exit
```

Datamart setup

7. Make sure your database uses UTF8 or AL32UTF8 encoding.

   This is needed to support users in Asian languages.

   For information about discovering and changing the database encoding, see *this Oracle knowledge base article*.

8. Connect to your Oracle database.

```
SQL> connect <adminusername>@<db_name>/<adminpassword> as sysdba
```

9. Create the datamart user you will use to connect from CollabNet TeamForge.

```
SQL> create user <reports_database_user> identified by
<reports_database_password> default tablespace <your tablespace> temporary
tablespace <temporary tablespace>;
```

10. Grant permissions to the user that you just created.

```
SQL> grant unlimited tablespace to <reports_database_user>;
SQL> grant create snapshot to <reports_database_user>;
SQL> grant create cluster to <reports_database_user>;
SQL> grant create database link to <sreports_database_user>;
SQL> grant create procedure to <reports_database_user>;
SQL> grant create sequence to <reports_database_user>;
SQL> grant create trigger to <reports_database_user>;
SQL> grant create type to <reports_database_user>;
SQL> grant create view to <reports_database_user>;
SQL> grant query rewrite to <reports_database_user>;
SQL> grant alter session to <reports_database_user>;
SQL> grant create table to <reports_database_user>;
SQL> grant create session to <reports_database_user>;
SQL> grant create any synonym to <reports_database_user>;
SQL> exit
```

☞ **Note:** Replace <reports_database_user> with the datamart username specified in the `site-options.conf` and <reports_database_password< with the database password specified in site-options.conf.

11. Create the datamart read-only user that you will use to connect from CollabNet TeamForge.

```
SQL> create user <reports_readonly_user> identified by
<reports_readonly_password> default tablespace <your tablespace> temporary
tablespace <temporary tablespace>;
```

12. Grant the required permissions to the read-only user that you just created.

```
SQL> grant create session to <reports_readonly_user>; SQL> exit
```

☞ **Note:** The CollabNet TeamForge installer creates the tables and default values for you.

Rebuild runtime without the install directory

You should keep your `teamforge-installer` directory around after installing TeamForge in case you need it later. However, if you delete or lose the directory you can still rebuild the application runtime.

1. Make a copy of the `runtime-options.conf` to use as the new `site-options.conf` file.

   ```
   cp /opt/collabnet/teamforge/runtime/conf/runtime-options.conf /var/tmp/
   site-options.conf
   ```

2. Rebuild the application runtime using the `create-runtime.py` script.

   ```
   /opt/collabnet/teamforge/dist/scripts/create-runtime.py -d /opt/collabnet/
   teamforge/ -f /var/tmp/site-options.conf
   ```

Change the scmviewer password

It is recommended to change the scmviewer password after installing TeamForge.

Follow these steps to change the scmviewer password:

1. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop
   ```

2. Create an encrypted password using the *password_util.sh* on page 762

   ```
   /opt/collabnet/teamforge/runtime/scripts/password_util.sh -encrypt
   '<new_password_text>'
   ```

3. Set the encrypted password to the *SCM_USER_ENCRYPTED_PASSWORD* on page 811 token in the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site_options.conf` file.

4. Recreate the runtime environment.

   ```
   cd /opt/collabnet/teamforge-installer/8.2.0.0
   ./install.sh -r
   ```

5. Start TeamForge.

   ```
   /etc/init.d/collabnet start
   ```

6. Run the Black Duck Code Sight `post-install.sh` script.

   ☞ **Important:** If the Black Duck Code Sight is running on the same server, run the `post-install.sh` script. If the Black Duck Code Sight is running on a separate server, follow steps one to four and execute the `post-install.sh` script in the Black Duck Code Sight server.

   ```
   /opt/collabnet/teamforge/runtime/scripts/codesearch/post-install.sh
   ```

7. Run the following command to set the new password for Review Board.

   ```
   cd /opt/collabnet/RBInstaller-1.8.0.0.12
   python ./install.py --auth-scmuser
   ```

Install Indexer on a separate server

In this task, we install the TeamForge Indexer on a separate server.

Pre-requisite: Ensure that there is no firewall restriction between the TeamForge application server and the new Indexer server.

Log on to the server as root user always.

Do this on the TeamForge application server.

1. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop
   ```

2. Move the Indexer directory [`/opt/collabnet/teamforge/var/searchIndex`] to the new Indexer server.

> ☞ **Note:** You can either use tar or NFS to back up and restore the searchIndex directory. Copy or transfer the searchIndex backup to the Indexer server. See the example below.

```
cd /opt/collabnet/teamforge/var
tar -zcvf /tmp/searchIndex.tgz searchIndex
scp /tmp/searchIndex.tgz user@indexer.box:/tmp/
```

3. Edit the `site-options.conf` file.

```
HOST_my.host.name=app etl database datamart subversion cvs gerrit
```

```
HOST_<domain/hostname of indexer box>=indexer
```

4. Run the installer.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -d /opt/collabnet/teamforge
```

5. Start TeamForge.

```
/etc/init.d/collabnet start all
```

Do this on the TeamForge Indexer server.

6. Set up the server with an operating system supported by TeamForge 8.2.

7. Configure the installation repository package for the OS of your choice.

   a) Configure your TeamForge 8.2 installation repository. See *TeamForge installation repository configuration for Red Hat/CentOS*.

   b) Configure your TeamForge 8.2 installation repository. See *TeamForge installation repository configuration for SUSE*.

8. Run the following command to install the TeamForge Indexer packages.

```
yum install teamforge-indexer
```

9. Copy the master `site-options.conf` file from the application server and modify these tokens:

```
HOST_my.host.name=indexer
```

```
DOMAIN_my.host.name=<myindexerbox.domain.com>
```

```
HOST_<myappboxdomain.com>=app etl database datamart subversion cvs gerrit
```

> ☞ **Important:**
> - It is recommended to use a separate netapp volume for the Indexer server.
> - The Indexer server should have the required permissions to access the NFS volume of the application server.

10. Create the Indexer and filestorage directories.

```
mkdir /opt/collabnet/teamforge/var/searchIndex -p
mkdir /opt/collabnet/teamforge/var/filestorage -p
```

11. Restore the Indexer data.

   Run the following command to restore the backup index data from the application server.

```
cd /opt/collabnet/teamforge/var/
tar -zxvf /tmp/searchIndex.tgz
```

12. Run the installer.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -d /opt/collabnet/teamforge
```

13. Start the TeamForge Indexer.

```
/etc/init.d/collabnet start
```

a) Run the following command and ensure that app filestorage directory is exported to indexer server.

```
mount -l
```

b) Run the following command and ensure that the Indexer services are up and running.

```
/etc/init.d/collabnet status
```

Install Memcached

Memcached caches Subversion (SVN) authentication and authorization information and serves the `mod_authnz_ctf` module's authentication and authorization requests thereby reducing the number of SOAP calls, which in turn results in less load on the TeamForge Application Server.

- See this *wiki page* for more information about Memcached.
- Memcached can run on the TeamForge Application Server or on a separate server. This document assumes that you install Memcached on the TeamForge Application Server.
- Caching is disabled by default in TeamForge. Enable it by setting up the following `site-options.conf` tokens.

  - *ENABLE_CACHING_WITH_MEMCACHED*
  - *MEMCACHED_SERVER_HOST*
  - *MEMCACHED_SERVER_PORT*
  - *MEMCACHED_SERVER_TTL*

To set up caching with Memcached:

1. Install Memcached.
2. Configure Memcached settings.
3. Set up the TeamForge site options.
4. Recreate the TeamForge runtime environment.

Do this on the TeamForge Application Server.

1. Install Memcached.

```
yum clean all
yum -y install memcached
```

2. Configure the OPTIONS key in the Memcached configuration file (`/etc/sysconfig/memcached`) and start Memcached.

a) The OPTIONS key in the `memcached` configuration file is used to set additional options during Memcached startup. Add the -l <ip-addr> flag to have Memcached listen to <ip_addr>. This is an important option to consider as there is no other way to secure the installation. Binding to an internal or firewalled network interface is recommended.

```
vi /etc/sysconfig/memcached
```

> ☞ **Important:** Remove the -l flag from the OPTIONS key to have Memcached listen to the server's default IP address or host name, including the 'localhost'.

b) Start Memcached.

> **Start Memcached during system boot up**: `sudo systemctl enable memcached`
> **Start Memcached normally**: `sudo systemctl start memcached`

3. Set up the following tokens in the TeamForge `site-options.conf` file.

See:

- *ENABLE_CACHING_WITH_MEMCACHED*
- *MEMCACHED_SERVER_HOST*
- *MEMCACHED_SERVER_PORT*
- *MEMCACHED_SERVER_TTL*

4. Recreate the TeamForge runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -n -V -d /opt/collabnet/teamforge
```

James configuration if Git is on a separate server

You must set the JAMES_ACCEPTED_RELAYS token with the Git server's IP address if you want Git notification emails to be sent to the TeamForge Application Server.

See *JAMES_ACCEPTED_RELAYS* on page 794 for more information.

1. Log on to the TeamForge Application Server and do this.
   a) Stop the collabnet service.

      ```
      /etc/init.d/collabnet stop
      ```

   b) Edit the `site-options.conf` file. Update the *JAMES_ACCEPTED_RELAYS* site-options token with the Git server's IP address.

      ```
      vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
      ```

      ```
      JAMES_ACCEPTED_RELAYS=127.0.0.1,{__CEE_DOMAIN__},<The IP address of the
        Git server>
      ```

   c) Recreate the runtime environment.

      ```
      cd /opt/collabnet/teamforge-installer/8.2.0.0
      ./install.sh -r -I -V
      ```

   d) Start the collabnet service.

      ```
      /etc/init.d/collabnet start
      ```

2. Log on to the Git server and do this.
   a) Copy the `site-options.conf` file from the TeamForge application server to the Git server.
   b) Recreate the runtime environment.

      ```
      cd /opt/collabnet/teamforge-installer/8.2.0.0
      ./install.sh -r -I -V
      ```

   c) Edit the `/opt/collabnet/gerrit/etc/gerrit.config` file and update the 'smtpServer' property with the TeamForge application server's host name.

      For example:

      ```
      [sendemail]
                    smtpServer = <TeamForge Application Server host name>
      ```

   d) Restart the gerrit service.

      ```
      /etc/init.d/collabnet stop gerrit
      /etc/init.d/collabnet start gerrit
      ```

Synchronize TeamForge source control integrations

Any time you upgrade your TeamForge site or a source control application, you must ensure that your users can still access their source code.

1. Click **Admin** in the CollabNet TeamForge navigation bar.
2. On the site administration navigation bar, click **Integrations**.
3. For each source control service you are supporting, verify that the right paths are specified.

- **SOAP service host** should be `localhost` or the host name of the server on which you just installed TeamForge.
- **Repository base URL** should be the URL for the top level of your source code server (which may be the same as your application server). For example, `http://<myscmbox>/svn/repos`
- **SCM Viewer URL** should be the URL for the ViewVC application on your source control server. For example, `http://<myscmbox>/integration/viewvc/viewvc.cgi`

  ☞ **Note:** If you want to turn on TeamForge code browser, specify the appropriate URL.

4. Select all your CVS integrations and click **Synchronize Permissions**.

   This updates the permissions on your code repositories so that users can access them from the new site.

☞ **Note:** By default, the *DISABLE_CREATE_INTEGRATION_SERVERS* flag in the `site-options.conf` file is set to false, which allows users to create new external integrations. To suppress the ability to add integrations, change this setting to true and recreate the runtime environment before making the site available to users.

Upgrade PostgreSQL using PGTurant

PGTurant, a wrapper for PostgreSQL's pg_upgrade (formerly known as pg_migrator), upgrades data stored in older versions of pgsql data files to a later pgsql major version, without data dump and reload, which otherwise is required for upgrades to major pgsql versions.

PGTurant is a multi-version PostgreSQL database system migrator that is capable of upgrading the PostgreSQL database file system to higher versions. As all the required PostgreSQL server, client and contrib RPMs are self-contained, customers need not worry about the required dependency packages while upgrading to higher PostgreSQL versions.

Though this tool was primarily intended for use by TeamForge applications during upgrades, it can also be used for any general purpose PostgreSQL upgrades provided the following prerequisites are met.

- Supported OS: 64-bit RHEL and CentOS (versions 6.x and 7.1)

  ☞ **Important:** You must have RHEL/CentOS 6.5 or later to run PGTurant. Upgrade the operating system packages (`yum upgrade`) before running PGTurant on sites running on RHEL/CentOS versions 6.4 or earlier.
- Pre-configured OS and CollabNet yum repository
- Python 2.6 or later
- A shell with 'root' access

The following table lists the supported PGSQL version for various TeamForge versions.

TeamForge version	Supported PostgreSQL version
6.2 (including 6.2 Patch 1)	9.0.13
7.0 (including 7.0 Patch 1)	9.0.13
7.1	9.2.4
7.2	9.2.4
8.0	9.3.5
8.0 Patch 1	9.3.6
8.1	9.3.6
8.2	9.3.6

The following table shows the possible upgrade options.

If you are in PostgreSQL 9.0	You can upgrade to PostgreSQL 9.2 or 9.3

If you are in PostgreSQL 9.2	You can upgrade to PostgreSQL 9.3

1. Starting TeamForge 6.2, PGTurant RPMs are distributed via CollabNet TeamForge's yum repository. Check your yum repository for PGTurant RPMs.

   a) Log on to your server as root user.

   b) Check for PGTurant package information.

   ```
   yum clean all
   yum info pgturant
   ```

   If you see PGTurant package information, your yum repository is fine. Otherwise, contact *CollabNet Support* for resolution.

2. Install PGTurant RPM.

   ```
   yum install pgturant -y
   ```

   PGTurant files are installed in the `/opt/collabnet/pgturant/` directory.

PGTurant command line options and usage examples

Option	Description
-D|--debug	Use this option to enable debug mode. Enabling the debug mode displays all the back-end debug messages on the console.
-s|--srcdir	Use this option to pass the PostgreSQL source database directory as a command line parameter. Suppose you want to upgrade your PostgreSQL data from v9.0 to v9.3 and that your v9.0 data directory is `/var/lib/pgsql/9.0/data`. Syntax for this option would be: `-s /var/lib/pgsql/9.0/data`.
-d|--destdir	Use this option to pass the PostgreSQL destination database directory as a command line parameter. Suppose you want to upgrade your PostgreSQL data from v9.0 to v9.3. Syntax for this option would be: `-d /var/lib/pgsql/9.3`. By default, the destination directory is assumed as `/var/lib/pgsql/<upgrade version>`, where <upgrade version> is specified in -u option.
-u|--upgrade	Use this option to pass the destination PostgreSQL version as a command line parameter so that the database files are upgraded to that specific PostgreSQL version supported by PGTurant.
-S|--setup	Use this option to install the required setup environment for PGTurant to be able to upgrade data to the desired PostgreSQL version.
-c|--check	Use this option to just check the database clusters and not change the database content. You can think of it as the dry-run mode.

Option	Description
-m|--migrate	Use this option to do the actual migration of database system to new PostgreSQL version as specified in the -u option.
-h|--help	Use this option to display PGTurant help information.

Example 1: Upgrade PostgreSQL database files from v9.0 to v9.3 - dry-run mode

```
cd /opt/collabnet/pgturant/bin
./pgturant -s /var/lib/pgsql/9.0/data -d /var/lib/pgsql/9.3 -u
9.3 -c
```

Example 2: Upgrade PostgreSQL database files from v9.2 to v9.3 - upgrade database

```
cd /opt/collabnet/pgturant/bin
./pgturant -s /var/lib/pgsql/9.2/data -d /var/lib/pgsql/9.3 -u
9.3 -m
```

Example 3: Upgrade PostgreSQL database files from v9.0 to v9.2 - dry-run and upgrade database

```
cd /opt/collabnet/pgturant/bin
./pgturant -s /var/lib/pgsql/9.0/data -d /var/lib/pgsql/9.2 -u
9.2 -c -m
```

Upgrade CLI reports to the latest version on Red Hat or CentOS

Use these instructions to upgrade the CLI reports to the latest version.

Though life cycle metric reports are released as part of regular TeamForge product releases (including patch releases), new reports are, at times, made available to customers as and when they are ready. When a new report is available, you can upgrade your CLI reports to the latest version.

1. Log on to TeamForge application server as root.
2. Install the latest `ctf_clireports`.

   ```
   yum install ctf_clireports
   ```
3. Run the `post-install.py` script.

   ```
   /opt/collabnet/teamforge/runtime/scripts/post-install.py
   ```
4. Run the following command and verify the CLI reports installation. Observe the version and release information.

   ```
   rpm -qi ctf_clireports
   ```

Upgrade Subversion on RHEL or CentOS

Use the yum package manager to upgrade to the latest supported Subversion release.

TeamForge 8.2 supports Subversion 1.8.14.

1. If you are doing this upgrade without Internet access, contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

   RHEL/CentOS 6.7

   • 32 bit RPM package: **CTFSVN-Disconnected-media-1.8.14-1.el6.i686.rpm**
   • 64 bit RPM package: **CTFSVN-Disconnected-media-1.8.14-1.el6.x86_64.rpm**

   RHEL/CentOS 7.0

   • 64 bit RPM package: **CTFSVN-Disconnected-media-1.8.14-1.el7.x86_64.rpm**

2. Set up the yum repository for upgrading Subversion.

   a) Log in as `root`.

   b) Install the disconnected-media package.

   ```
   rpm -ivh package_name
   ```

   package_name is the path of the media that you downloaded, for example, "/tmp/CTFSVN-Disconnected-media-1.8.14-1.el7.x86_64.rpm"

3. Log in as `root` and stop all `collabnet` services.

4. Check that Subversion 1.8.14 is available for upgrade:

   ```
   yum list subversion
   ```

5. Upgrade Subversion.

   ```
   yum update subversion subversion-perl subversion-python mod_dav_svn
   ```

6. Verify the Subversion upgrade.

   ```
   rpm -qa | grep subversion
   svn --version | grep " version"
   ```

7. 👉 **Note:** In the case of a distributed setup, where Subversion runs on a separate server, skip this step and go directly to the last step to restart `collabnet` services.

   Upgrade and set permissions only if you are upgrading Subversion on the `application` server.

   a) Change to the `runtime/scripts` directory.

   ```
   cd /opt/collabnet/teamforge/runtime/scripts
   ```

   b) Run the Subversion upgrade script.

   ```
   ./svn-upgrade-working-copies.sh
   ```

   c) Set the permissions for branding and `publish` directories.

   ```
   eval `./options-to-shell-variables.py`
   chown $HTTPD_USER:$HTTPD_GROUP $OVERRIDES_DIR $PUBLISH_DIR -R
   ```

8. Start all `collabnet` services.

   ```
   /etc/init.d/collabnet start all
   ```

Authenticate users with LDAP

Use LDAP to facilitate managing users and groups in TeamForge.

Set up LDAP integration for the TeamForge site

Follow these steps to convert your TeamForge installation to authenticate against your corporate OpenLDAP server.

👉 **Note:** Refer to the *Software requirements for TeamForge 8.2* topic for the supported OpenLDAP versions.

1. Stop TeamForge.

   ```
   /etc/init.d/collabnet stop all
   ```

2. Edit the `site-options.conf` file.

   a) Enable TeamForge to use LDAP authentication by editing the `<installation_source>/conf/site-options.conf` file, for example, edit /opt/collabnet/teamforge-installer/8.0.0.1/conf/site-options.conf file.

   Under "External User Authentication," uncomment this line:

   ```
   USE_EXTERNAL_USER_AUTHENTICATION=false
   ```

   and change its value to true.

   b) Configure the site-options tokens.

☞ **Note:** The values specified for the following tokens are only for illustration purpose.

```
EXTERNAL_AUTHENTICATION_TYPE=ldap
LDAP_DN_PREFIX=cn=
LDAP_DN_SUFFIX=,cn=Users,dc=testldap,dc=qa,dc=collab,dc=net
LDAP_SERVER_URL=ldap://testldap.qa.collab.net:3268
```

3. Recreate the runtime environment.

```
./install.sh -V -r -d /opt/collabnet/teamforge
```

4. Start TeamForge.

```
/etc/init.d/collabnet start all
```

Turn off LDAP authentication

During some maintenance operations, such as upgrades, you may need to turn off LDAP authentication.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. In the `site-options.conf` file, comment out these variables:

```
USE_EXTERNAL_USER_AUTHENTICATION
```

```
LOGIN_CONFIG_XML_FILE
```

```
MINIMUM_PASSWORD_LENGTH
```

3. Recreate the runtime environment.

```
./install.sh -V -r -d /opt/collabnet/teamforge
```

4. Review the variables you have changed, then save the `site-options.conf` file.

Maintain database

Follow these recommendations for maintaining your TeamForge database.

Context for the current task

Schedule data extraction for reporting

Set the interval at which you want yourTeamForge site's data extracted to the datamart from which reports are generated.

Each extract-transform-load (ETL) job consists of extracting the data from the production database, transforming it to support reporting, and loading it into the datamart.

By default, this is done every night at 2:30 a.m., by the host's local clock.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Set the *ETL_JOB_TRIGGER_TIME* variable to the interval at which you want ETL jobs to run.
For example, a value of `0 0/15 * * * ?` will run an ETL job every 15 minutes.

3. Review the variables you have changed, then save the `site-options.conf` file.

Raise the logging visibility of selected database requests

For easier troubleshooting, you can dictate that certain database requests get logged in a handy central log file.

For example, database requests that run longer than 10 seconds are likely candidates for troubleshooting. You can have such requests automatically logged in the `vamessages.log` file for your inspection. The exact length of time after which a request becomes problematic depends on your environment.

How it works:

- All database queries are logged at DEBUG level by default.
- By default, the `vamessages.log` file is configured to include all events logged at the INFO level or higher.
- Database queries that run over a configurable time limit are logged at INFO rather than DEBUG, which causes them to appear in `vamessages.log`.

1. Stop TeamForge.

    Stop PostgreSQL on RHEL/Centos:
    Stop PostgreSQL on SUSE:
    Stop TeamForge: `/etc/init.d/collabnet stop`

2. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

    ```
    vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
    ```

    ☞ **Note:** `vi` is an example. Any *nix text editor will work.

3. In the `site-options.conf` file, change the value of the *LOG_QUERY_TIME_THRESHOLD* variable to a value, in milliseconds, that makes sense for your environment.

4. Run the installer.

    ```
    cd /opt/collabnet/teamforge-installer/8.2.0.0
    ./install.sh -r -I -V
    ```

5. Start TeamForge.

    ```
    /etc/init.d/collabnet start
    ```

Optimize PostgreSQL with vacuum

To optimize your PostgreSQL database, run a built-in utility called "vacuum."

Normal use of database software often creates data overhead that needs to be cleaned periodically in order to ensure optimal speed and stability. This overhead is usually the result of temporary files and indexes that the database creates (analogous to a fragmented hard disk.)

The vacuum utility runs on a live database and, like the backup command, can be scripted to run nightly or at minimal server load times.

1. To vacuum the CollabNet TeamForge database, run the `vacuum` command as the PostgreSQL user.

    ```
    vacuumdb -h `hostname` -U <DATABASE_USERNAME> -z <DATABASE_NAME>
    ```

    For example, using the default values in the `site-options.conf` file:

    ```
    vacuumdb -h `hostname` -U ctfuser -z ctfdb
    ```

2. To set up automatic vacuuming of the database based on activity statistics, set up auto-vacuuming according to these instructions:

    http://www.postgresql.org/docs/8.2/interactive/routine-vacuuming.html#AUTOVACUUM.

Move the datamart (dedicated database server mode)

The setup with all services on one server includes having the datamart in the same PostgreSQL instance as TeamForge as well as running it in a separate instance. In either case, you can move the PostgreSQL datamart from its own instance to a separate box in the dedicated database server mode.

1. Stop TeamForge on the `app` server.

If you have a distributed setup, stop TeamForge on all other servers as well.

```
/etc/init.d/collabnet stop
```

2. Create a dump of the PostgreSQL datamart.

```
su - postgres -c 'pg_dump -C -p <database-port > <reports-database-name >
<path-to-dump-file>
```

3. Create a new datamart instance using the `datamart-pgsql-setup.sh` or use the following commands on the database server.

```
su - postgres
initdb -D /var/lib/pgsql/9.0/reports
```

4. Set the *REPORTS_DATABASE_PORT* in `site-options.conf` file.

☞ **Note:** The port should use the same value as specified in `postgresql.conf` as specified in the previous step. The recommended value is 5632.

5. Recreate the [RUNTIME_DIR] on all the servers.

```
install.sh -r -d /opt/collabnet/teamforge
```

6. Restore the datamart into the new instance.

```
[RUNTIME_DIR]/scripts/db.py -a restore -t reporting -f <dump-location>
```

7. Copy the `postgresql-9.3` script from runtime scripts and replace `/etc/init.d/postgresql-9.3`

8. Restore the datamart from the database server.

```
su - postgres -c 'psql -p <reports-database-port> <path-to-dump-file>
```

☞ **Note:** Restart the Postgres service. If any warning messages are displayed, kill the service and start again.

9. Start the database in the database box.

```
/etc/init.d/postgresql-9.3 start
```

10. Start the services on all servers.

```
/etc/init.d/collabnet start all
```

11. Check your old reports for data accuracy.

12. Permanently remove the old datamart from the TeamForge instance.

```
su - postgres -c 'dropdb <datamart-name> -p <database-port>'
```

Protect your CollabNet TeamForge site

You can take various measures to maximize the security of your CollabNet TeamForge users.

Encrypt communication between the application and database servers

To prevent your data from being exposed in a readable format on the network, use Secure Socket Layer (SSL) to encrypt the network traffic between the application and the database servers.

If you have a dedicated server for your database (operational database or datamart database), encrypt the data traffic between the application and database servers and between the ETL and datamart servers.

☞ **Important:** The following steps are applicable only in a multi-box installation setup.

Log onto the server as root user always

1. Stop TeamForge in all the servers.

```
/etc/init.d/collabnet stop all
```

2. Add the following site option tokens in all the TeamForge servers.
   a) If the operational database is running on a separate server, include the token *DATABASE_SSL=on*.
   b) If the datamart database is running on a separate server, include the token *REPORTS_DATABASE_SSL=on*

☞ **Note:** It is mandatory to include the tokens specified above in all the servers.

3. Recreate the runtime environment in all the TeamForge servers.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0

./install.sh -r -n -V -d /opt/collabnet/teamforge
```

4. Start TeamForge in all the servers.

```
/etc/init.d/collabnet start all
```

5. Verify that your PostgreSQL database is running in the SSL mode.
   a) Log in to the database server.
   b) Run the following command:

```
grep "ssl = " var/lib/pgsql/9.3/data/postgresql.conf
```

   Observe:"ssl = on"

Set up SELinux

If SELinux is active on the server that runs your CollabNet TeamForge site, modify it to allow the services that TeamForge requires.

Pre-requisites to set up the SELinux environment:

- TeamForge SELinux can be configured only for the TeamForge application server.
- TeamForge SELinux supports RHEL/CentOS 7.2 or later versions. RHEL/CentOS 6.7 is also supported.
- In case of same server upgrade using RHEL/CentOS, it is recommended to upgrade the OS to RHEL/CentOS 7.2 or later versions.
- **Log on to the server as root user always.**

1. Stop TeamForge.

```
/etc/init.d/collabnet stop all
```

2. Set SELinux to run in enforcing mode on the TeamForge application server.
   a) Edit the file `/etc/sysconfig/selinux` and set `SELINUX=enforcing`.
   b) Turn off TeamForge startup on boot.

```
chkconfig collabnet off
```

   c) Reboot the server.
   d) Ensure that SELinux is running in enforcing mode.

```
getenforce
```

3. Make sure that the SELINUX_SETUP site options token is set to true in the `site-options.conf` file of the TeamForge application server.

```
SELINUX_SETUP=true
```

4. Run the following commands if Review Board is integrated with TeamForge.

```
semanage fcontext -a -t httpd_sys_rw_content_t "/opt/collabnet/
reviewboard/data(/.*)?"
restorecon -R -v /opt/collabnet/reviewboard/data
semanage fcontext -a -t httpd_sys_rw_content_t "/u1/reviewboard(/.*)?"
restorecon -R -v /u1/reviewboard
semanage fcontext -a -t httpd_sys_rw_content_t "/opt/collabnet/teamforge/
var/home/apache(/.*)?"
restorecon -R -v /opt/collabnet/teamforge/var/home/apache
```

5. Before recreating runtime, remove `pdk-root` and `pdk-sf-admin` directories by running the following commands:

```
rm -rf /tmp/pdk-root
rm -rf /tmp/pdk-sf-admin
```

6. Recreate the runtime environment.

> ☞ **Important:** Do this only if you have modified the `site-options.conf` file.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

7. Start TeamForge.

```
/etc/init.d/collabnet start
```

8. Fix the SELinux data permissions.

```
cd /opt/collabnet/teamforge/runtime/scripts
./fix_data_selinux_permissions.sh
```

> ☞ **Note:** If your data directory is on a NetApp volume, you may observe some warning messages which can be ignored safely.

Protect your TeamForge site with SSL

Use Secure Socket Layer (SSL) to run your Web server securely.

Set up SSL for your TeamForge site

To force all TeamForge traffic to use SSL encryption (HTTPS), state that preference in your configuration file.

1. Back up your existing `/etc/httpd/conf/httpd.conf` file.

2. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

> ☞ **Note:** `vi` is an example. Any *nix text editor will work.

3. Set the options to enable SSL for the site.

   a) Set the *SSL* variable to on.

   b) Set the *SSL_CERT_FILE* variable to the location of the file that contains your site's SSL certificates.

   ```
   SSL_CERT_FILE=www.example.com.crt
   ```

   c) Set the *SSL_KEY_FILE* variable to the location of the file that contains your site's RSA private keys.

   ```
   SSL_KEY_FILE=www.example.com.key
   ```

   > ☞ **Important:** Select a location for your cert file and your key file that is permanent across restarts. Don't use a temp directory that can be wiped out.

4. In the `site-options.conf` file, make sure the value of the *DOMAIN_localhost* variable matches that of your SSL certificate.

5. Rename the `ssl.conf` file (RHEL/CentOS: `/etc/httpd/conf.d/ssl.conf`) as `ssl.conf.old`, if it exists.

6. If you are converting an existing site to use SSL (that is, if your site already has had users accessing it via HTTP and not HTTPS), you must update your site's publishing repository to use the new SSL settings.

To do this, ask your CollabNet support representative for the `fix-publishing-repos-to-ssl.py` script.

7. Stop all TeamForge services.

```
/etc/init.d/collabnet stop all
```

8. Run the installer.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

9. Start TeamForge

```
/etc/init.d/collabnet start all
```

A new Apache configuration file is created with the information you provided in the `site-options.conf` file. The new file is named `httpd.conf.cn_new`. It contains `VirtualHost` sections for port 80 and port 443. All port 80 requests are redirected to port 443.

When you point your browser at CollabNet TeamForge, it should now automatically redirect to HTTPS.

As your site is configured to run in the SSL mode (http to https), there is a change in the URI scheme. Run the following post installation scripts that are applicable to make your TeamForge integration components to function seamlessly.

Run the TeamForge `post-install.py` script.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

Generate SSL certificates

To use https for web traffic, you will need to obtain a valid Apache SSL certificate.

When generating an Apache (mod_ssl) SSL certificate, you have two options:

- Purchase a SSL certificate from a certificate authority (CA). Searching the Web for "certificate authority" will present several choices.
- Generate a self-signed certificate. This option costs nothing and provides the same level of encryption as a certificate purchased from a certificate authority (CA). However, this option can be a mild annoyance to some users, because Internet Explorer (IE) issues a harmless warning each time a user visits a site that uses a self-signed certificate.

Regardless of which option you select, the process is almost identical.

1. Know the fully qualified domain name (FQDN) of the website for which you want to request a certificate.

   If you want to access your site through `https://www.example.com`, then the FQDN of your website is `www.example.com`.

   ☞ **Note:** This is also known as your common name.

2. Generate the key with the SSL `genrsa` command.

   ```
   openssl genrsa -out www.example.com.key 1024
   ```

   This command generates a 1024 bit RSA private key and stores it in the file `www.example.com.key`.

   ☞ **Tip:** Back up your `www.example.com.key` file, because without this file your SSL certificate will not be valid.

3. Generate the CSR with SSL `req` command.

   ```
   openssl req -new -key www.example.com.key -out www.example.com.csr
   ```

   This command will prompt you for the X.509 attributes of your certificate. Give the fully qualified domain name, such as `www.example.com`, when prompted for `Common Name`.

   ☞ **Note:** Do not enter your personal name here. It is requesting a certificate for a webserver, so the Common Name has to match the FQDN of your website.

4. Generate a self-signed certificate.

   ```
   openssl x509 -req -days 370 -in www.example.com.csr -signkey
   www.example.com.key -out www.example.com.crt
   ```

   This command will generate a self-signed certificate in `www.example.com.crt`.

You will now have an RSA private key in `www.example.com.key`, a Certificate Signing Request in `www.example.com.csr`, and an SSL certificate in `www.example.com.crt`. The self-signed SSL certificate that you generated will be valid for 370 days.

Prevent HTTPS cracking

To reduce the risk of HTTPS ciphers being cracked, allow only the strongest ciphers available.

Deploying an Apache SSL certificate and forcing https ensures that all data is encrypted. It does not, however, ensure that the encryption methods (also known as ciphers) that are used are strong. With the ever-increasing power of computers, many older or weaker ciphers can be cracked in a matter of days or even hours by a determined person with malicious intentions.

1. In the `/etc/httpd/conf.d/ssl.conf` file, find the headings *SSLProtocol* and *SSLCipherSuite*.

   ☞ **Note:** If they do not exist, add them below the *SSLEngine* line.

2. In each section, add the following two lines:

```
SSLProtocol all -SSLv2 SSLCipherSuite
RSA:!EXP:!NULL:+HIGH:+MEDIUM:-LOW
```

3. Save the file and restart Apache.

```
apachectl restart
```

Protect integrations with SSL

If you have registered Secure Socket Layer (SSL) certificates, your site's users can use SSL when they set up an SCM integration server.

If you use certificates that are generated in-house, self-signed, or signed by a non-established Certificate Authority, they must be registered with each client system that will connect to the TeamForge server. Registration consists of importing custom certificates into the Java runtime's global keystore on each server.

☞ **Important:** This will affect any other Java applications on the server that use the same Java runtime.

1. Collect server certificates from all servers.

   On RHEL, CentOS and other RedHat-based distributions, these are contained in `/etc/httpd/conf/ssl.crt/server.crt`.

   ☞ **Tip:** Be sure to use exactly this path, as there are other files with similar names, plus server certificates are not really secret, but some other files are. So, files must be copied (e.g., via scp) to the same directory, and renamed if necessary to avoid clashes. We recommend that you use the short server name of the corresponding server for this.

2. Locate the Java keystore.

   This is `PATH_TO_JAVA/jre/lib/security/cacerts`.

   For example, this may be `/usr/local/j2sdk1.4.2_10/jre/lib/security/cacerts`.

3. Locate the Java `keytool` utility.

   This is `PATH_TO_JAVA/bin/keytool`

   For example, `/usr/local/j2sdk1.4.2_10/bin/keytool`.

4. Import each server certificate into the keystore.

```
PATH_TO_JAVA/bin/keytool -import -keystore PATH_TO_JAVA/jre/lib/security/
cacerts -file <server>.crt -alias <server>
```

   ☞ **Note:** Any value is accepted for server in -alias <server>.

5. At the password prompt, use `changeit`.

   Confirm that you trust the certificate by typing `yes`.

6. Verify that all your certificates are added.

```
PATH_TO_JAVA/bin/keytool -list -keystore PATH_TO_JAVA/jre/lib/security/
cacerts |less
```

   ☞ **Note:** The list will contain many more certificates. These are top-level CA certificates, provided with Java.

7. If you are running more than one separate server, repeat these steps for each server.

8. Restart TeamForge

Now you can check the **Use SSL** checkbox when creating an SCM integration.

Logs

Here is some additional information about maintaining your log files.

Context for the current task

Change the location of a log file
To change where log files are written to, edit the `site-options.conf` file and restart the runtime environment.

1. Stop the site.

   ```
   /etc/init.d/collabnet stop all
   ```
2. In the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf` file,
   change the value of the *LOG_DIR* variable to reflect the location where you want the log files to be written.
3. Recreate the runtime environment.

   ```
   ./install.sh -V -r -d /opt/collabnet/teamforge
   ```
4. Start the site.

   ```
   /etc/init.d/httpd start
   /etc/init.d/postgresql-9.3 start
   /etc/init.d/collabnet start
   ```

All future Apache logs, mail logs, database logs, java logs, and other logs will be written to the new location.

Change the logging level on your site
Set the logging level appropriately to enable logging in `vamessages.log`.

1. Edit `$RUNTIME/jboss/bin/jboss-cli.sh` to enable logging in `vamessages.log`.

   ☞ **Note:** You need **not** restart the site for JBoss to pick up these changes.

2. To change log levels, as a root user, perform the following:
   a) To enable debug logging, run the following command:

      ```
      /subsystem=logging/root-logger=ROOT:change-root-log-level(level=DEBUG)
      ```
   b) To disable debug logging, run the following command:

      ```
      /subsystem=logging/root-logger=ROOT:change-root-log-level(level=INFO)
      ```
3. To change log levels using your LDAP credentials, perform the following:
   a) To enable debug logging, run the following command:

      ```
      /subsystem=logging/logger=com.vasoftware:write-attribute(name="level",
      value="DEBUG")
      ```
   b) To disable debug logging, run the following command:

      ```
      /subsystem=logging/logger=com.vasoftware:write-attribute(name="level",
      value="INFO")
      ```
   c) To enable trace logging for LDAP, run the following command:

      ```
      /subsystem=logging/
      logger=org.jboss.security.auth.spi.LdapExtLoginModule:add(level=TRACE,handlers=["VAR
      ```
   d) To disable trace logging for LDAP, run the following command:

      ```
      /subsystem=logging/
      logger=org.jboss.security.auth.spi.LdapExtLoginModule:remove()
      ```

      ☞ **Tip:** The LDAP debug output will be very limited unless you add `<module-option`
      `name="throwValidateErrors" value="true"></>`to the entry for the corresponding log-in
      module.

• Change in log levels will not need any site restart and these changes will not survive the Jboss restart.

- Changes made using CLI will survive a restart but not a runtime recreation.

  ☞ **Note:** Changes are immediately saved to `runtime/jboss/standalone/configuration/standalone-full.xml` and the file, `standalone-full.xml` is recreated every time the TeamForge runtime is rebuilt.

- To retain the configurations after the site restart, edit `$RUNTIME/jboss/standalone/configuration/standalone-full.xml`.
- To make the configurations survive the recreate runtime, edit `$SITE_DIR/dist/conf-snippets/jboss/standalone-full.xml.d/20-standalone-full.xml.py`.

Rotate TeamForge log files

To keep log files under control, move them to a compressed archive.

Rotate logs automatically

Set up your site to move log files automatically to a compressed archive every day.

1. Check that the system's cron jobs are set.

   ```
   sudo crontab -l
   ```

   If the cron jobs are in place, you get output like this:

   ```
   # Begin sourceforge cron jobs (Generated by the build.  DO NOT HAND EDIT.)
    --
   4,9,14,19,24,29,34,39,44,49,54,59 * * * * /opt/collabnet/teamforge/
   runtime/scripts/run-groups.sh 5mins
   0 * * * * /opt/collabnet/teamforge/runtime/scripts/run-groups.sh hourly
   0 0 * * * /opt/collabnet/teamforge/runtime/scripts/run-groups.sh daily
   0 0 * * 0 /opt/collabnet/teamforge/runtime/scripts/run-groups.sh weekly
   0 0 1 * * /opt/collabnet/teamforge/runtime/scripts/run-groups.sh monthly
   # End sourceforge cron entries
    ---------------------------------------------
   * * * * * /usr/bin/aiupdate -n -d http://mgr/dist >/dev/null 2>&1
   ```

   If the output does not appear, you need to set up the cron jobs.

2. In the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf` file, set the value of the *INSTALL_CRON* option to true.
3. Recreate the runtime environment.

   ```
   ./install.sh -V -r -d /opt/collabnet/teamforge
   ```

All your site's logs will now be automatically rotated into a compressed archive every 24 hours.

Rotate logs manually

Use the `rotatelogs.pl` script to move log files to a compressed archive.

Do this when a log file has become so large it is difficult to use.

Run the `rotatelogs.pl` script.

```
sudo /opt/collabnet/teamforge/runtime/scripts/rotatelogs.pl
```

All your site's logs are moved into a compressed archive and new log files are started.

Search index optimization

For an optimal use of the search index feature, follow these tips.

Context for the current task

Rebuild TeamForge search indexes

You can rebuild your site's search index without stopping TeamForge.

Any new objects created during this time will not be immediately indexed, but will be queued until after the re-indexing.

1. Make sure the TeamForge site is up.
2. Run the re-index script.

    /opt/collabnet/teamforge/runtime/scripts/SearchReindex.py

After the script completes, everything is queued for re-indexing. It will take some time to process the re-index requests.

☞ **Note:** Re-indexing could take several hours, and the index data could double in size.

Permit big file uploads

When many users store very large files on your site, you may sometimes notice a slowdown in your site's performance. You can reduce the impact of such a use pattern by telling TeamForge not to index files larger than a certain size.

☞ **Note:** It's also a good idea to let your users know that the Documents tool in TeamForge is not designed primarily as a storage device. As a best practice, upload documents to make them available for collaboration, not for backup or long-term storage.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

    vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf

    ☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Add the *SEARCH_MAX_FILE_SIZE* parameter and give it a value equal to the maximum size (in bytes) of files to be indexed.

    ☞ **Note:** The default value is 10M. With this value, files larger than 10M are not indexed.

    A value of zero or less specifies that there is no limit, which is the same as the default behavior without the variable.
3. Review the variables you have changed, then save the `site-options.conf` file.

Allow searching of archive files

By default, users can't search the content of archive files uploaded to TeamForge, such as `zip`, `tar`, or `docx` files. If your users need it, you can provide this ability.

Microsoft Office 2007 files, such as files with the `.docx` extension, are archive files. By default their content is not indexed and does not show up in search results. However, information that TeamForge maintains about those documents, such as title, author, description and version, does appear in search results.

If you permit archive searching, watch for performance slowdowns associated with the larger volume of indexing that TeamForge is doing. Depending on your site members' use patterns, the performance cost may or may not be acceptable to your users.

☞ **Note:** It's also a good idea to let your users know that the Documents tool in TeamForge is not designed primarily as a storage device. As a best practice, upload documents to make them available for collaboration, not for backup or long-term storage.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

    vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf

    ☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Add the *SEARCH_SUPPRESS_ARCHIVE_SUB_DOCS* parameter and set it to false.
    This enables files inside archives (such as `.zip`, `.gz`, or `.tar`) to be indexed for search.
3. Review the variables you have changed, then save the `site-options.conf` file.

Reports

Here is additional information on maintaining reports in TeamForge.

Context for the current task

> Task step.

Turn on site-wide reporting

To use the site-wide reporting functionality, you have to configure a collection of variables in the `site-options.conf` file.

> ☞ **Note:** Site administrators can view the status of the ETL server in **System Tools** > **Server Status**. The status displays the following values:
>
> - "OK" if enabled and running
> - "N/A" when disabled
> - "Could not connect" when enabled and not running

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

   ```
   vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
   ```

   > ☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Comment out the default *HOST_localhost=app database subversion cvs* variable, then add (or uncomment) the variable that has the reporting services enabled:

3. Add these variables (or uncomment them, if they are already present) and give them the appropriate values:

   ```
   REPORTS_DATABASE_NAME=ctfrptdb
   REPORTS_DATABASE_PASSWORD=ctfrptpwd
   REPORTS_DATABASE_USERNAME=ctfrptuser
   REPORTS_DATABASE_READ_ONLY_USER=ctfrptreadonly
   REPORTS_DATABASE_READ_ONLY_PASSWORD=rptropwd
   REPORTS_DATABASE_MAX_POOL_SIZE=30
   ```

4. If you want your datamart to use a different port than your site database, we recommend the following:

   - For small dataset customers: use the same instance as that of `ctfdb`; leave REPORTS_DATABASE_PORT commented.
   - For medium to large dataset customers: use a separate instance by setting REPORTS_DATABASE_PORT=5632.

5. For ETL to function, add these variables:

   ```
   ETL_SOAP_SHARED_SECRET=<arbitrary_string>
   SOAP_ANONYMOUS_SHARED_SECRET=<arbitrary_string>
   ```

   > ☞ **Tip:** For more information about configuring variables, see *site-options.conf* on page 775.

6. Set the time you want the reporting data to be collected, if different from the default 2:30 a.m. (local time).

   ```
   ETL_JOB_TRIGGER_TIME=<cron expression>
   ```

   > ☞ **Tip:** This value is a cron expression, not a time value. See *ETL_JOB_TRIGGER_TIME* on page 788 for more.

7. Review the variables you have changed, then save the `site-options.conf` file.

SCM

These are some additional details you need to know about SCM.

Context for the current task

> Task step.

Provide more than one source control server

To run more than one source control server of the same type on your site, each integration must have its own unique name and data directory.

☞ **Note:** No single server can host more than one source control integration of the same type. If you want to have more than one Subversion integration, they must run on separate machines. The same is true for CVS integrations.

1. Manually create the necessary directories on NetApp.

   ☞ **Important:** Each directory must have its own unique name. For example, if the first Subversion instance is named `svnroot`, you might name the second instance `svnroot_2`.

2. Set the permissions on the new directories to `ctf-admin:ctf-admin`

3. For each such directory on the NetApp, create a separate symlink in the local filesystem pointing to the NetApp mount folder.
   For example, assuming the NetApp mount is mounted on the `/shared` mount point in the local filesystem:

   ```
   sudo ln -s /svnroot /shared/svnroot_2
   ```

   ☞ **Note:** Only one source code integration of any one type can run on a machine.

Email

These are additional details you can follow while configuring your email settings.

Context for the current task

Removing users from monitoring objects

As a site or project administrator, if one or more users are no longer project members, you can remove them from monitoring selected TeamForge objects they once subscribed for monitoring.

However, you cannot remove a user from the monitoring list if the user is *monitoring applications* such as trackers, documents, tasks, and so on instead of individual TeamForge objects.

By default, this feature is disabled. To enable this feature, set the *USER_MONITORING_REMOVE_ENABLED* variable to true in the `site-options.conf` file.

☞ **Note:** Every user removal operation is being logged in the database for audit purposes.

1. Go to the item, from which you want to remove users from monitoring.

2. Select users to remove from monitoring list.
   a) If you want to remove one or more users from monitoring one of the items, select the item, then click **Monitor** > **Users Monitoring Selected**.
      The **Users Monitoring This Item** window appears.
   b) If you want to remove one or more users from monitoring more than one item, select all the items, then click **Monitor** > **Users Monitoring This Folder**.
      The **Users Monitoring This Folder** window appears.
   c) Select one or more check boxes corresponding to the users you want to remove from monitoring.

3. Click **Remove**.
   The `Are you sure you want to remove the selected user(s) from monitoring?` message appears.

4. Click **OK**.

The selected users are removed from monitoring the selected object. An e-mail notification is sent to all active users that are removed from monitoring selected objects.

Limit the size of message attachments

To avoid overtaxing your mail server or your storage volume, you may want to set a ceiling on the size of the attachments that users can send to a forum via email.

When a user sends an attachment that is larger than the limit, the message is rejected and the user gets an email from the Site Administrator explaining that the attachment exceeded the limit.

☞ **Tip:** Before imposing a file attachment size limit, it's a good idea to point your users to better ways of collaborating around large files. Consider suggesting source code repositories, backup systems, or other appropriate solutions.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

   `vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf`

   ☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Set the value of the *DISCUSSION_MAX_ATTACHMENT_SIZE* variable to a reasonable level.
   For example, if your users are given to using Microsoft Word documents on the site, you might set *DISCUSSION_MAX_ATTACHMENT_SIZE* to 10 MB, and increase the value by two or three MB at a time if users need more headroom.

3. Review the variables you have changed, then save the `site-options.conf` file.

Limit the size of document attachments

When many users store very large documents on your site, you may sometimes notice a slowdown in your site's performance. You can reduce the impact of such a use pattern by telling TeamForge not to attach documents larger than a certain size.

☞ **Note:** It's also a good idea to let your users know that the Documents tool in TeamForge is not designed primarily as a storage device. As a best practice, upload documents to make them available for collaboration, not for backup or long-term storage.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

   `vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf`

   ☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Add the *DOCUMENT_MAX_FILE_UPLOAD_SIZE* parameter and give it a value equal to the maximum size (in megabytes) of documents to be uploaded.

3. Review the variables you have changed, then save the `site-options.conf` file.

Discussions

Here is some additional information on discussions.

Context for the current task

Who can post to discussions by email?

To help reduce the risk of spam or other mischief, you may need to limit the users who can post to discussion forums by email.

To leverage the advantages of community collaboration, you should keep your forums as open as you can. However, some sites require tighter control over who can participate in discussions. TeamForge enables you to balance openness against privacy along a spectrum of choices.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

   `vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf`

   ☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Add the *DISCUSSION_EMAIL_POSTING* variable, and give it the value that reflects the degree of openness your site's discussion forums require.
   Use one of these values:

Value	Description
0	Allow only forum admins.
1	Users with roles and permissions.
4	All logged in users.
5	Allow known email addresses only.
6	Allow all site users and guests.

3. Review the variables you have changed, then save the `site-options.conf` file.

The value you set here determines the maximum degree of openness to email posting for all projects on your site. For example, consider a site where project members can post by email (level 3). For a project that requires extra security, the project administrator can choose to accept email only from users with the appropriate role (level 1). However, a project owner cannot accept email posts from a less restrictive category of users, such as all users who are logged in (level 4).

Who can monitor discussions?
You may need to limit the users who can monitor discussion forums.

To leverage the advantages of community collaboration, you should keep your forums as open as you can. However, some sites require tighter control over how users keep track of discussions. TeamForge enables you to balance openness against privacy along a spectrum of choices.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Add the *DISCUSSION_EMAIL_MONITORING* variable, and give it the value that reflects who you want to get updates on discussions via email.

Use one of these values:

Value	Description
0	Allow only forum admins.
1	Users with role permissions.
4	All logged in users.
5	Allow all site users and guests.

3. Review the variables you have changed, then save the `site-options.conf` file.

The value you set here determines the maximum degree of openness to monitoring discussions for all projects on your site. For example, consider a site where project members can monitor discussions (level 3). For a project that requires extra security, the project administrator can dictate that only users with the appropriate role can monitor discussions (level 1). However, a project owner cannot allow monitoring for a less restrictive category of users, such as all users who are logged in (level 4).

Reduce discussion spam
You can filter out some kinds of spam from your project's discussion forums.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Add one or more MIME types to the Reject MIME types filter.

   The presence of any of these MIME types in an incoming message (via email) will cause its rejection with appropriate notification to the posting user.

   For example:*DISCUSSION_REJECT_MIME_TYPES*=application/pdf,text/xml

3. Add one or more MIME types to the Drop MIME types filter.

   The presence of any of these MIME types in an incoming message (via email)will cause its attachment to be deleted with appropriate notification to the posting user.

   For example:*DISCUSSION_DROP_MIME_TYPES*=image/jpeg,image/jpg,text/xml

4. Add one or more header names to the remove headers filter.

   If an incoming email posting contains any of these headers, they will be quietly removed from the message before it is archived and before subscribers are notified.

   For example:*DISCUSSION_REMOVE_HEADERS*=precedence,x-no-archive,Return-Path

5. Add one or more header names to reject headers filter to be rejected or moderated (if discussion is moderated).

   Use regular expressions, each regular expression must match an entire header. The match of any of these headers in an incoming message (via email) will cause its rejection with appropriate notification to the posting user.

   For example:*DISCUSSION_REJECT_HEADERS*=(?s).*headername1:value2.*,(?s).*name2:value2.*

6. Add one or more entries for Reject content filter.

   Use regular expressions, each regular expression must match an entire entry. The match of any of these entries in discussion body and subject of an incoming message (via email) will cause its rejection with appropriate notification to the posting user.

   For example: *DISCUSSION_REJECT_CONTENT*=(?s).*word.*,(?s).*spam.*

   ☞   **Note:** The content entry is a case sensitive.

7. Review the variables you have changed, then save the `site-options.conf` file.
8. Restart the site.

General

You can do these things to help maximize the effectiveness and productivity of your site's users.

Context for the current task

   Task step.

Get information about a CollabNet TeamForge site

Use the `snapshot.py` utility to determine what processes are running on your CollabNet TeamForge site, how much free memory is available, and other information.

1. Log into the server.
2. Find the application in distress.
3. Run the `snapshot.py` script.

       /opt/collabnet/teamforge/runtime/scripts/snapshot.py

Snapshot gathers data from several processes running on the system, including:

- JBoss
- Tomcat
- James
- PostgreSQL
- Apache
- C6Migration

The information is written to `LOG_DIR/runtime/snapshot.log` and `LOG_DIR/apps/server.log`.

☞ **Note:** `LOG_DIR` is the directory you defined as the logging directory in the `site-options.conf` file.

Specify DNS servers

Define the DNS servers with which you want CollabNet TeamForge to resolve URLs by listing them in the `resolv.conf` file.

1. In the `/etc/resolv.conf` file, list the servers you want to use for resolving Internet addresses.
2. Rebuild the runtime environment.

    ```
    ./install.sh -V -r -d /opt/collabnet/teamforge
    ```
3. Restart CollabNet TeamForge .

    ```
    /etc/init.d/httpd start
    /etc/init.d/postgresql-9.3 start
    /etc/init.d/collabnet start
    ```

Change your site's domain name

To change the domain name of your site, you must change the name in the file system, the database, and any integrated applications such as the Project Tracker or the Review Board. TeamForge provides a script for each of these.

1. Make sure your site's database and file system are backed up.
2. In the `site-options.conf` file, update the domain name (and hostname if needed), then save the file.

    ```
    DOMAIN_<Host_Name>=<Domain_Name>
    JAMES_POSTMASTER_EMAIL=root@<Domain_Name>
    JAMES_MTA_HOST=<Domain_Name>
    ```

    For example:

    ```
    DOMAIN_mybox.supervillain.net=www.smileyface.com
    JAMES_POSTMASTER_EMAIL=root@www.smileyface.com
    JAMES_MTA_HOST=www.smileyface.com
    ```
3. Recreate the runtime environment.

    ```
    cd /opt/collabnet/teamforge-installer/8.2.0.0
    ./install.sh -r -I -V
    ```
4. Go to the TeamForge scripts directory.

    ```
    cd /opt/collabnet/teamforge/runtime/scripts/
    ```
5. Run the script to change the domain in the file system.

    ```
    ./domain_change_fs.pl --old=www.myoldsitename.net --
    new=www.mynewsitename.net --dir=/opt/collabnet/teamforge/var | tee /tmp/
    domain_change_fs.out 2>&1
    ```
6. Run the script to change the domain in the database.

    ```
    ./domain_change_db.py --old=www.myoldsitename.net --
    new=www.mynewsitename.net > /tmp/domain_change_db.out 2>&1
    ```
7. If your site has Project Tracker integrated, run the script to change the domain in Project Tracker.

    ```
    ./domain_change_pt.py --oldDomain=www.myoldsitename.net --
    newDomain=www.mynewsitename.net > /tmp/domain_change_pt.out 2> &1
    ```
8. If your site has Review Board integrated, run the following commands:
    a) Start the TeamForge application services.

```
sudo /etc/init.d/collabnet start all
```
b) Update the `install.conf` file with the new domain name.

```
vi /opt/collabnet/RBInstaller-1.8.0.0.12/installer/install.conf
```
c) Recreate the runtime on Review Board.

```
sudo python ./install.py -r
```
d) Restart the TeamForge application services.

```
sudo /etc/init.d/collabnet restart all
```

Let users see what's in a project template

Help your site's project administrators choose a project template by enabling them to see the contents of the templates that are available.

By default, only site administrators can see project template detail, but project administrators normally create a project from a project template. To choose the right template, a project manager may want to know if tasks, documents, wiki pages or other kinds of content are included in a given template.

1. Open the `site-options.conf` file, the master configuration file that controls your TeamForge site.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

☞ **Note:** `vi` is an example. Any *nix text editor will work.

2. Change the value of the SHOW_PROJECT_TEMPALTE_DETAIL_TO_NON_SITEADMINS variable to true.

☞ **Note:** Use the parameter name as given, including the typo.

3. Review the variables you have changed, then save the `site-options.conf` file.

4. Recreate the runtime environment.

```
./install.sh -V -r -d /opt/collabnet/teamforge
```

Integrate TeamForge 8.2 with other tools

TeamForge 8.2 supports integrations with third-party tools for versioning, reviewing, searching source code, binary repository management, and test management.

Orchestrate

New to Orchestrate? Read through these topics to know more about Orchestrate that provides traceability for product life cycle activities such as work items, SCM commits, continuous integration (CI) builds, and code reviews.

Get started
New to Orchestrate? Read through these topics to get oriented.
Overview
TeamForge Orchestrate is a TeamForge capability that provides traceability for product life cycle activities such as work items, SCM commits, continuous integration (CI) builds, and code reviews.

Problem statement and Solution

Organizations currently tend to have modern mixed-vendor, heterogeneous environments with complex lifecycles including work item, commit, review, build/test, deploy, and other tasks that are monitored or managed by stand-alone tools. The tools may be vendor-supplied or open-source, and may reside on-premises or in private, public, or hybrid clouds. These tools, however, lack the ability to associate with one another and do not lend themselves to end-to-end traceability: organizations cannot easily see the connections between activities derived from disparate lifecycle tools. Yet to achieve traceability, organizations are often forced to use an all-in-one, monolithic solution that excludes popular point tools. TeamForge Orchestrate offers a traceability solution that preserves the advantages of your best-of-breed tools.

While TeamForge Orchestrate is extensible, this version provides the following packaged adapters for connecting lifecycle activities:

Table 2: TeamForge Orchestrate Packaged Adapters

Type of service	Products
SCM/Version Control	Git, Subversion
Code Review	ReviewBoard, Gerrit, Atlassian Crucible
Build/Test	Jenkins, JetBrains TeamCity
Issue Tracker	CollabNet TeamForge, Atlassian JIRA

TeamForge Orchestrate Vision

In a modern and frequently fractured development environment employing tools from different vendors, TeamForge Orchestrate aims to provide:

- Aggregation — Using packaged adapters and an extensible architecture, TeamForge Orchestrate obtains and stores metadata about key lifecycle activities like work items, commits, builds/tests, and code reviews across vendors and platforms.
- Traceability — TeamForge Orchestrate draws associations between activities across the entire lifecycle and enables traceability through the various steps or stages of the lifecycle.

Feature overview

TeamForge Orchestrate aims to aggregate lifecycle metadata across various tools and establishes networks of associations across those lifecycle activities. The use cases for TeamForge Orchestrate include visibility into development activities, visibility into associations between lifecycle activities, and requirements traceability for auditing purposes.

Activity Streams: This is the most prominent view in TeamForge Orchestrate. Activity streams provide a "project chronology" showing the most recent activities at the top of the page and the oldest at the bottom. See *Activity Stream* for more information.

Activity Details: Each activity has an "activity details" page that displays more detailed metadata related to the activity. For example, a commit detail page has information on the SCM repository name, committer information, commit comments, etc. The detail page also has an "Associations" section that shows all associated activities. These associations are created automatically either by the source systems or explicitly through special notation in a commit message. See *Activity Details* for details on the different activity pages.

Interface to TeamForge: If a TeamForge work item is seen on the Associations section in a TeamForge Orchestrate activity detail page, the "tracker" view for that work item in TeamForge will reflect those activities in its Associations tab. Clicking on those associations from the TeamForge tracker takes you back to the corresponding activity detail page in TeamForge Orchestrate.

Extensibility

While TeamForge Orchestrate ships with some adapters, its extensibility ensures that users can write custom adapters to extend TeamForge Orchestrate to other work item trackers, version control, CI, and code review systems. New product classes may also be extended using Extensible Data Sources (XDS). See *Extending TeamForge Orchestrate* on page 663 for more information.

Access TeamForge Orchestrate

TeamForge Orchestrate is accessible as a project tool inside TeamForge.

Once installed and configured for your project, access TeamForge Orchestrate by navigating to your target project inside TeamForge and select the "Orchestrate" project button.

To install and configure TeamForge Orchestrate, see *Installation* on page 616.

☞ **Note:** Be sure that your TeamForge user has the appropriate project permissions as documented in *User authentication and authorization* on page 598.

1. Log into your TeamForge instance.
2. Select your desired project.
   Navigate to **Projects** > **My Projects** and select the desired project.
3. Click the Orchestrate project tool bar icon.

   If your project is configured to use Orchestrate, locate and click the **Orchestrate** tool bar icon:

   Continue with the *User interface overview* on page 598 to learn more about using Orchestrate.

Architecture

The TeamForge Orchestrate architecture aims for extensibility, fault tolerance, and scalability.

Overview

End users experience TeamForge Orchestrate as a seamless component of TeamForge. However, under the covers Orchestrate is a separate application with a distinct architecture. This article intends to give an overview of that architecture to assist with installation and administration of the underlying services as well as an overview of the message queue (MQ) architecture used to integrate with and collect data from external sources.

Architectural Components

At the highest level, there are three server side components of TeamForge Orchestrate:

- Orchestrate application server: Orchestrate is a Ruby on Rails application that runs on the Phusion Passenger application server with Nginx as a web server. The application server also hosts a number of daemons that read messages from RabbitMQ.
- MongoDB: Orchestrate persists end user-facing data to MongoDB, a NoSQL variety document-store database.
- RabbitMQ: Orchestrate relies on a message queue architecture to integrate with external sources. RabbitMQ is used to collect activity data from sources like work item, SCM, build, and code review systems.

Hosting Considerations

TeamForge Orchestrate is designed for host separation, where each component resides on different operation systems. Any component may be run on virtualized hardware. In the absence of separate hardware or virtualization, Orchestrate components may be collocated. Resource requirements for components are documented in the *Installation* on page 616 section.

Load balancing and clustering

TeamForge Orchestrate components may be load balanced and/or clustered. The Orchestrate application server may be installed on multiple hosts and load balanced (session clustering is not required). MongoDB and RabbitMQ both support clustered modes as well. However, such configurations are not documented or supported and are only offered as supplemental services from CollabNet.

Integration with external sources

TeamForge Orchestrate uses a MQ architecture to collect data from external sources like work item, SCM, build/ CI, and code review systems. The diagram below demonstrates a clustered network of MQ servers that share data collected from external sources. When a relevant activity occurs, the external source uses an MQ agent to publish a specially formatted message to the MQ network. MQ servers can be clustered and strategically located to prevent loss of messages during to outages.

Once a message is published by an external source, it is consumed by a daemon service running on the Orchestrate application server. The message is parsed and the data is persisted. MQ adapters are configured with "association keys": unique identifiers that enable Orchestrate to file incoming data into the appropriate source.

Extensibility

Orchestrate ships with a handful of packaged adapters to common work item, SCM, build/CI, and review systems. But what if you are using different tools in your process? You can integrate your tools by building adapters that work by publishing MQ messages that are consumed by Orchestrate. See the Extensibility documentation for more on writing TeamForge Orchestrate adapters: *Extending TeamForge Orchestrate* on page 663.

User authentication and authorization

TeamForge Orchestrate is authenticated and authorized by TeamForge, using TeamForge users, permissions and roles. To access TeamForge Orchestrate features, TeamForge users need additional permissions on a project by project basis.

TeamForge Orchestrate requires a valid TeamForge user session for access. If you try to access TeamForge Orchestrate when you are not logged in or when your session has expired, you will be directed to the TeamForge login page. After you successfully log in, you will be automatically redirected to the target TeamForge Orchestrate screen.

Permissions

TeamForge Orchestrate permissions are set in the "Project Admin" area on a project by project basis. The following Orchestrate permissions may be used in creating TeamForge roles:

- Pipeline read — The minimum required permission for a user to use TeamForge Orchestrate. Users with this permission level can view activity details and make comments.
- Pipeline edit — Required to modify data sources, which feed into the activity stream. Users with the "Pipeline edit" permission are implicitly granted the "Pipeline read" permission.
- Pipeline create — Required to create new data sources. Users with the "Pipeline create" permission are implicitly granted the "Pipeline read" and "Pipeline edit" permissions.
- REST API — Required to retrieve data from the Orchestrate API. TeamForge site administrators do not explicitly require this permission; however, all other project members, including project administrators, require this permission to use the API.

☞ **Note:** TeamForge Orchestrate does not immediately reflect permission changes. Permissions are automatically refreshed daily. To reflect permission changes more quickly, log out of TeamForge, wait 10 minutes, and then log back in again.

User interface overview
Overview of the TeamForge Orchestrate user interface including activity streams and activity detail pages.

Activity Stream View

Clicking a project's Orchestrate toolbar button displays the *"activity strea"m*: a chronological list of life-cycle activities related to the selected project.

1. *Click to read about the filter option*
2. *Click to read about the "Review Detai"l page*
3. *Click to read about the "Build Detai"l page*
4. *Click to read about the "Commit Detai"l page*

Activity Detail View

From the activity stream, clicking an activity widget displays a page with detailed information pertaining to the selected commit.

1. *Click to read about activity associations*
2. *Click to read about the "traceabilit"y feature*

Traceability report

The *traceability report* shows associations between related activities. Clicking an activity widget displays the detail view for the selected activity.

User interface icons

A summary of all the icons used in TeamForge Orchestrate.

Status Icons

Icon	Status	Usage
✔	Successful	• Successful builds • Successful and submitted reviews • Closed (Done) work items • Successful tests associated with a build activity
✖	Failure	• Failed builds • Failed tests associated with a build activity
🕐	Pending	• Pending builds • Pending work items • Open reviews
⊖	Ignored	• Ignored tests associated with a build activity
⊘	Aborted	• Aborted builds

Icon	Status	Usage
⚠	Unstable	• Discarded (dead) reviews • Unstable and unknown builds

Other Icons used in TeamForge Orchestrate

Table 3: Miscellaneous Icons

Icon	Name	Description
	TeamForge Orchestrate Icon	Represents TeamForge Orchestrate in the TeamForge toolbar
	Build Icon	Represents a build activity
	Review request icon	Represents a review activity
	Code Commit icon	Represents a commit activity
	Work Item icon	Represents a work item activity
	Filter icon	Represents the filtering option used in activity stream pages and the traceability graph. See *Filter Option in User Interface*
99+	Comment icon	Represents the specified number of comments associated with an activity

Activity Sources

Add "sources" that populate activity streams with relevant "activity" data such as work item updates, commits, CI builds, and code reviews.

What is a source? - TeamForge is an open development platform with a strong notion associations. Within TeamForge, any two objects can be associated together, and when multiple associations are chained together, they create "traceability". But what about tools outside of TeamForge? Orchestrate provides just such a mechanism, allowing you to add relevant project tools to the TeamForge project context. This is accomplished through "sources", which represent light-weight integrations to external products. Once established, "sources" capture activity data from external tools, bringing them into the TeamForge project context and associations fabric.

For work items, sources are issue trackers; for SCM commits, sources are defined as the code repository of interest; for CI builds, sources are "jobs" or "build configurations"; for code reviews, sources are defined as the project or repository of interest.

How do sources work? - Whenever an "activity" occurs in a tool that has been setup as a project source, a special "adapter" sends notification of the activity to TeamForge Orchestrate. For example, if a specific Jenkins job is configured as a source, build information will be sent to TeamForge Orchestrate when that Jenkins job is executed. The resulting activity is mapped to the TeamForge project in question. A TeamForge project can have multiple sources, all sending their activity data to the TeamForge project.

Associations - Other than capturing external activities and displaying them in an activity stream, Orchestrate draws associations between these activities. In some cases, this is entirely automatic, other cases require user input to help Orchestrate draw accurate associations. See *Associations* on page 614.

Extensibility - TeamForge Orchestrate is extensible; users may author their own adapters for new source types and even register new product domains. If your organization uses tools that do not yet have adapters, you can write your own adapter that sends activity data to TeamForge Orchestrate. See *Extending TeamForge Orchestrate* on page 663.

Manage build source

Add build sources to bring automated build job results into the TeamForge project and traceability context.

Build sources bring build/CI data into Orchestrate for archival of metadata, participation in traceability, and activity reporting.

Create or find an existing Tool to group your build source under

1. From the desired project context, navigate to **Project Admin** > **Tools**.

2. Determine whether a "Tool" exists that represents a logical container for your desired source. For instance, if you are adding a Jenkins source, look for a tool representing the instance of Jenkins in question.

   a) If a Tool exists already, click the Tool's title to edit the Tool.

   b) If a Tool does not exist, click **Add Tool**.

Edit an existing build source

3. To edit an existing build source, find the Data Source in the list of sources associated with the tool.

   You can edit the display name or associate a different commit source for this build source. If you do not select a commit source, no associations will be drawn for the build activities coming from this build source. TeamForge displays a warning if a build source is not associated with a commit source. See *Manage commit sources*.

4. Click **Update** to save your changes.

   ☞ **Note:** While editing an existing source configuration, you can toggle a source from Active to Inactive, which will stop TeamForge from collecting data from that source; toggle back to Active to resume data collection from the source.

Create a new build source

5. If this is a new Tool, first select the Tool Type in the drop-down menu. For instance, if you're adding a Jenkins Tool instance, select Jenkins.

6. Select the **Include Traceability** check box.

   A blank data source appears. If this is an existing Tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.

   a) Provide a display name for the build source. This display name will help you differentiate the source in lists throughout the user interfaces. The display name can be up to 100 alphanumeric characters in length.

   b) Select a commit source.

   Select the commit source from the list of existing commit sources (or create a new one).

   ☞ **Important:**

   - You must select a commit source from the list of existing commit sources to let TeamForge create associations between your build and commit activities.
   - If you do not select a commit source, no associations will be drawn for the build activities coming from this build source. TeamForge displays a warning if a build source is not associated with a commit source. See *Manage commit sources* for more information about adding a commit source.
   - For most source types, TeamForge generates a unique "source association key". The source association key uniquely identifies and helps route data from sources properly in TeamForge. You must supply this string while configuring adapters for most source types. You can copy the key by clicking on the small clipboard icon. No association key is necessary when you configure a TeamForge project code repository, though.

7. Click **Update**.

TeamForge saves the new build source and activates it.

Manage review sources

Add review sources to bring code reviews into the TeamForge project and traceability context.

Review sources bring code review data into EventQ for archival of meta data, participation in traceability, and activity reporting.

Create or find an existing Tool to group your build source under

1. From the desired project context, navigate to **Project Admin** > **Tools**.

2. Determine whether a "Tool" exists that represents a logical container for your desired source. For instance, if you are adding a Review Board source, look for a tool representing the instance of Review Board in question.

   a) If a Tool exists already, click the Tool's title to edit the Tool.

   b) If a Tool does not exist, click **Add Tool**.

Edit an existing build source

3. To edit an existing review source, locate the **Data Source** in question on the **Edit Tool** screen. You can edit the display name or commit association prefix for this source.

4. Click **Update** to save your changes.

   ☞ **Note:** While editing an existing source configuration, you can toggle a source from Active to Inactive, which will stop TeamForge from collecting data from that source; toggle back to Active to resume data collection from the source.

Create a new review source

5. If this is a new Tool, first select the Tool Type in the drop-down menu. For instance, if you're adding a Review Board Tool instance, select Review Board.

6. Select the **Include Traceability** check box.

   A blank data source appears. If this is an existing Tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.

   a) Provide a display name for the review source. This display name will help you differentiate the source in lists throughout the user interfaces. The display name can be up to 100 alphanumeric characters in length.

   b) Enter the Commit Association Prefix.

   This prefix is used to associate code commits with code reviews using the commit message. To associate a review with a code commit, you must enter the association prefix and review ID in the commit message in this format:

   ```
   [key:review_ID]
   ```

   Where key is any alphanumeric string. A commit association key may be up to 20 characters, including alphanumeric and underscore ("_") characters.

7. Click **Update**.

TeamForge saves the new review source, and activates it.

Manage commit sources

Commit sources enable your external repositories to participate in traceability.

Commit sources bring commit data into TeamForge for archival of metadata, participation in traceability, and activity reporting.

   ☞ **Note:** TeamForge Project SCM repositories are added automatically as sources, both on upgrade and when new repositories are defined in the TeamForge project context.

Create or find an existing Tool to group your commit source under

1. From the desired project context, navigate to **Project Admin** > **Tools**.

2. Determine whether a "Tool" exists that represents a logical container for your desired source. For instance, if you are adding a source to represent an external Git repository, look for a tool representing the Git server in question.

   a) If a Tool exists already, click the Tool's title to edit the Tool.

   b) If a Tool does not exist, click **Add Tool**.

Edit an existing commit source

3. To edit an existing commit source, locate the **Data Source** in question on the **Edit Tool** screen.

You can edit the display name for any commit source. Commit sources can be defined with one of two repository types:

- **Project repositories**: housed within TeamForge projects
- **External repositories**: housed outside of TeamForge projects

☞ **Important:** Once defined, you cannot switch the repository type. For external repositories, you can edit the repository URL. For project repositories, "Source Code view" permission is required and once selected and saved to a source, the project repository selection may not be altered. To work around this constraint, create a new source with a new repository and deactivate the old source.

4. Click **Update** to save your changes.

☞ **Note:** While editing an existing source configuration, you can toggle a source from Active to Inactive, which will stop TeamForge from collecting data from that source; toggle back to Active to resume data collection from the source.

Create a new commit source

5. If this is a new Tool, first select the Tool Type in the drop-down menu. For instance, if you're adding an external Git Tool instance, select External Git.

6. Select the **Include Traceability** check box.

A blank data source appears. If this is an existing Tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.

a) Provide a display name for the commit source. This display name will help you differentiate the source in lists throughout the user interfaces where this source appears. The display name can be up to 100 alphanumeric characters in length.

b) Select the repository type. Your source may be either a TeamForge project repository or an "external" repository.

- **Project repositories**: repositories housed within TeamForge projects
- **External repositories**: repositories housed outside of TeamForge projects

☞ **Note:** Sources for Project repositories are created automatically.You need "Source Code view" permission to see available project repositories.

c) For External repositories, enter your repository URL.

This is likely the URI used to check out code or a WebDAV-enabled URL.

- **For Subversion repositories**: run the `svn info` command inside the working copy and copy/paste the value of the "URL" field.
- **For Git repositories**: run the `git remote show origin` inside the working tree and use the value of the "Fetch URL" field, without the "username@".

For example, if your Subversion repository URL is set to `https://forge.example.com/svn/repos/myproject`, TeamForge Orchestrate collects messages from the "myproject" repository and all repositories under "myproject" such as `https://forge.example.com/svn/repos/myproject/branches/mybranch`.

7. Click **Update**.

TeamForge saves the new commit source, and activates it.

☞ **Note:** TeamForge Orchestrate displays all the defined sources for a step in the order in which they were defined.

Manage work item sources

Add work item sources to tickets, issues, defects and other work items to bring work items into the TeamForge project and traceability context.

Work item sources bring tracker data into TeamForge for archival of metadata, participation in traceability, and activity reporting.

☞ **Note:** TeamForge trackers are added automatically as sources, both on upgrade and when new trackers are defined in the TeamForge project context.

Create or find an existing Tool to group your work item source under

1. From the desired project context, navigate to **Project Admin** > **Tools**.

2. Determine whether a "Tool" exists that represents a logical container for your desired source. if you are adding a JIRA® source, look for a tool representing the instance of JIRA in question.

   a) If a Tool exists already, click the Tool's title to edit the Tool.

   b) If a Tool does not exist, click **Add Tool**.

Edit an existing work item source

3. To edit an existing work item source, locate the **Data Source** in question on the **Edit Tool** screen.

   You can edit the display name, while all other properties are system defined.

4. Click **Update** to save your changes.

   ☞ **Note:** While editing an existing source configuration, you can toggle a source from Active to Inactive, which will stop TeamForge from collecting data from that source; toggle back to Active to resume data collection from the source.

Create a new work item source

5. If this is a new Tool, first select the Tool Type in the drop-down menu. For instance, if you're adding an external JIRA Tool instance, select the appropriate Tool Type.

6. Select the **Include Traceability** check box.

   A blank data source appears. If this is an existing Tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.

   a) Provide a display name for the work item source. This display name will help you differentiate the source throughout the user interfaces where this source appears. The display name can be up to 100 alphanumeric characters in length.

7. Click **Update**.

TeamForge Orchestrate saves the new work item source, and activates it.

Extensible Data Source (XDS) Overview

This section provides a description of Extensible Data Sources (XDS), a means to add arbitrary, user-defined sources.

XDS Schemas and Sources

Orchestrate can be configured to accept data from a wide range of sources through the Extensible Data Source (or XDS) feature. In addition to the stock activity APIs defined for commits, builds, reviews, and work items, the XDS feature enables you to create your own activity APIs by defining an "XDS Schema" that describes the expected activity message format for a particular tool or product domain in question. See *Extending TeamForge Orchestrate* on page 663 for more details and links to API documentation.

"XDS Sources" can be added to your project through the "Custom" source type. Once added, XDS activities will be visible in the activity stream. XDS sources can also have associations just like any other source resulting in traceability between XDS and other activities, even between multiple XDS sources.

Manage XDS sources

Add Extensible Data Sources (XDS) to bring a wide variety of tools into the TeamForge project and traceability context.

XDS sources provide a means to integrate tools that do not fall under the stock activity classes (commit, build, review, and work item). You can add XDS sources to represent a wide range of tools and product domains. See *Extending TeamForge Orchestrate* on page 663 for more on how XDS works.

Create or find an existing Tool to group your XDS source under

1. From the desired project context, navigate to **Project Admin** > **Tools**.

2. Determine whether a "Tool" exists that represents a logical container for your desired source. if you are adding a Chef™ source, look for a tool representing the Chef instance in question.

a) If a Tool exists already, click the Tool's title to edit the Tool.

b) If a Tool does not exist, click **Add Tool**.

Edit an existing XDS source

3. To edit an existing XDS source, locate the **Data Source** in question on the **Edit Tool** screen.

You can edit the display name, look and feel, and tags. If you save an XDS source without defining the Associated Source, you may edit the XDS source later and add an Associated Source. Once the Associated Source has been defined it may not be altered.

4. Click **Update** to save your changes.

☞ **Note:** While editing an existing source configuration, you can toggle a source from Active to Inactive, which will stop TeamForge from collecting data from that source; toggle back to Active to resume data collection from the source.

Create a new XDS source

5. If this is a new Tool, first select the Tool Type in the drop-down menu. For instance, if you're adding a Nexus Tool instance, select the matching XDS Schema name from the drop-down. Alternatively, select "Other" if no applicable choice exists yet.

6. Select the **Include Traceability** check box.

A blank data source appears. If this is an existing Tool with existing sources, click the **Data Source + (Add Source)** icon to add a new blank source.

a) Provide a display name for the XDS source that is depicted on all TeamForge Orchestrate user interfaces. The display name can be up to 100 alphanumeric characters in length.

b) Select an icon and set the icon background color to differentiate your source visually.

c) Select an Associated Source.

• Select a source from the list of existing sources. TeamForge Orchestrate associates activities from the XDS source to the selected source. For instance, if you want the XDS source in question to associate to builds from job XYZ, select the source corresponding to job XYZ in the **Associated Source** drop-down list.

• If you do not select an associated source, no associations will be drawn to the activities coming from this XDS source.

d) Add tags to the Source as a means to organize and categorize XDS Sources.

Tags are particularly useful as filters for reporting via the Reporting API.

⚠ **Caution:** Tags are shared across the entire site. Exercise caution so that no superfluous tags are created which may make reporting more difficult.

7. Click **Update**.

TeamForge Orchestrate saves the new custom source, and activates it.

Activity stream

The activity stream page displays a list of Orchestrate "activities" such as work item updates, commits, builds, and review updates.

Overview

The activity stream is a feed of recent project activities, listed in reverse chronological order (newest activities on top). There is one activity stream per TeamForge project. You can use the activity stream to:

• Keep up to date on recent project updates
• Find a recent activity
• Filter for relevant updates using *filters*

Interface and navigation

The activity stream user interface features "activity widgets" on the left side, a concise representation of the activity including source information and unique identifiers. The right side of the stream features a synopsis of the activity.

The activity stream is paginated, showing 50 activities per page. All activities in the stream appear according to the timestamp set by the source system. If an activity has been commented upon, you see a comment "bubble" displaying the number of comments on that activity. For a list of status icons and their contexts, see *User Interface Icons*.

Notification of new activities

The activity stream interface will indicate the availability of new activities. While an activity stream is open in a browser tab, notifications will appear on the activity stream and in your browser tab indicating the number of newly available activities. Refresh the tab or click the notification on the activity stream to load the newly available activities.

Commits

Commits are depicted in blue, and are identified by a "source code" icon.

The activity stream displays the following commit attributes :

Commit Attribute	Description	Location
SCM repository commit ID	The commit revision number.	Commit activity header
SCM source name	The user defined "friendly name" alias the source.	Commit activity header
Committer	Name and user profile picture of the user responsible for the commit transaction. A default user avatar is depicted if the user has not defined a profile picture in TeamForge.	Commit summary
Commit message	Up to four lines of the committer's explanatory text submitted at commit time.	Commit summary
Committed At	The date and time of the commit as reported by the source SCM system.	Commit summary

Builds

Build activities are depicted in purple, and are identified by a "building block" icon.

The activity stream displays the following build attributes:

Build Attribute	Description	Location
Build number	The build identifier as supplied by the source build/CI system.	Build activity header
Build source name	The user defined "friendly name" alias the source.	Build activity header
Timestamp	The date and time the build finished as reported by the build server.	Build summary
Build result status	Can be one of these states: `Successful`, `Failed`, `Aborted`, or `Unstable`.	Build summary
Build duration	The time it took for the build to run, from start to finish.	Build summary
Test results	If the source job is configured to run automated tests, TeamForge Orchestrate presents the test results, summarized by these states: `passed`, `failed`, `aborted`, or `ignored`. If there are no test results due to an aborted build or other errors which prevent an anticipated test from starting, TeamForge Orchestrate displays "`No test results`".	Build summary

Review requests

Review request activities are depicted in green, and are identified by a "magnifying glass" icon.

The activity stream displays the following review attributes:

Review Attribute	Description	Location
Review request number	The review identifier as supplied by the source review system.	Review request activity header
Review source location display name	The user-defined "friendly name" alias for the source.	Review request activity header
Review creator or respondent	The user who created or updated the review as reported by the source review system.	Review summary
Created and updated at:	The date and time that the review request was requested and updated as reported by the source review system.	Review summary
Review status	The current state of the request represented by an icon and text description. Text descriptions vary based on the underlying source review system. Statuses are associated to a status type that conforms to the TeamForge Orchestrate API. Conforming status types include: `open`, `submitted`, `rejected`, `discarded`, or `other`.	Review summary
Review summary	Text summarizing the review request creation event or updates to an existing review. The review summary may contain the following information: • Change in review title • Change in review description (summary) • Change in review status • Review files added or removed • Addition of new reviewers	Review summary

Work item updates

Work item activities are depicted in orange, and are identified by a corresponding "tracker" icon.

The activity stream displays the following work item attributes:

Work Item Attribute	Description	Location
Artifact ID	The work item identifier as supplied by the source work item tracking system.	Work item activity header
Work item display name	The user-defined "friendly name" alias for the work item source.	Work item activity header
Updated on:	The date and time that the work item was updated as reported by the source work item system.	Work item summary
Work item status	The current state of the work item represented by an icon and text description. Text descriptions vary based on the underlying source review system. Status types are shown in parenthesis to further clarify the status intention (e.g., Closed, Open, In Progress).	Work item summary

Work Item Attribute	Description	Location
Work item title	Text summarizing the Work item creation event or updates to an existing work item. The work item summary may contain the following information: • Change in work item title • Change in review description (summary) • Change in review status • Review files added or removed • Addition of new reviewers	Work item summary
Relevant changes to the work item	Text summarizing the Work item creation event or updates to an existing work item. The work item summary may contain the following information: • Change in Description • Change in Status • Change in Priority • Change in User Assignment	Work item summary

Use activity stream filters

You can filter activity streams by date to narrow the range of visible activities or to trim a lengthy activity list.

The activity stream may be filtered using the **Filter** button in the header area.

1. Click **Filter**.
2. (Optional) Enter valid dates for start and/or end dates; you can use the pop-up calendars. If no date filter is set, TeamForge Orchestrate depicts all activities up to the current date. To remove a previously selected date, click **Clear**.
3. Click **OK** to save the filter criteria and display the filtered activity stream or click **Close** to cancel any selected filters.

   To restore the filters to their previous values, you must reload the page.

Activity detail pages

Activities in TeamForge Orchestrate often have associated detail pages that give more information about the activity, the ability to comment, and to view associated activities.

For example, a commit detail page has information on the SCM repository name, the committer information, commit comments, files affected, etc. The detail page also has an "Associations" section that shows all activities and TeamForge work items that are associated with it.

Note that external activities do not have detail pages but instead may link to an external URL.

Build detail

The Build Detail shows information pertaining to a single build including attributes like the build number, duration, build status, and associations to other activities like commits, reviews, and work items associated with the build. If the CI job is configured to run tests, this view also shows test summary information including the number of passed, failed or ignored tests.

Build Detail Attributes

The step associated with the current activity is highlighted. To see the detail for a different activity, click on the associated step.

The Build Detail displays these attributes:

Build Detail Attribute	Description	Location
Build number	The ID for a specific build supplied by the CI server.	Build activity header
Build source display name	The user defined "friendly" label for the build source job.	Build activity header
Timestamp	The date and time the build completed as reported by the CI server.	Build summary
Build duration	The time it took for the build to run, from start to finish.	Build summary
Build result	The final outcome of the build and corresponding external link to the build summary. The build result can be one of these states: `succeeded`, `failed`, `aborted`, `unstable`, or `unknown`.	Build summary
Test result	Test summary data is displayed in the following categories: `passed`, `failed`, or `ignored`. An external link to test results is also displayed. If there are no associated test results, the following message is displayed instead: `"No test results"`.	Build summary (icon and value)
Comment	Comment and user profile picture of the user who created the comment. A default user avatar is depicted if the user has not defined a profile picture in TeamForge.	Comment section

Build Associations

Builds may be associated with other activities such as commits. The Build Detail displays this information in the "Associations" section::

- Commits — Commits are identified by the commit ID, and may be linked to the corresponding commit details page. TeamForge Orchestrate makes no assumptions when associating commits to builds; the list of associated commits originates from the build source. Commits are linked to the related commit detail only if TeamForge Orchestrate is monitoring the SCM repository in question; else, only the commit ID is displayed.

  Missing commit details — TeamForge Orchestrate sometimes learns about commits before the details of these commits are officially reported by the commit source. Such commits are grayed out temporarily in the "Associations" sections of, for example, review and build detail pages until TeamForge Orchestrate receives this information. Traceability graphs include these commits, but you cannot click on the commit nodes to get commit details. See *Troubleshooting TeamForge Orchestrate* for information on commits with missing detail information.

  See *Commit detail* on page 610 for information about commit details.
- Reviews — Reviews are associated with builds through the commit associations that they both share in common. Review requests are identified by the source display name (user inputted) and review ID. Reviews are linked to the corresponding Review Detail page.
- Work Items — Work items are associated with builds through the commit associations that they both share in common. See *Associate code commits with other items*

Commit detail

The Commit Detail shows information pertaining to a single version control commit including attributes like the repository commit ID, full commit message, commit author, date and time of commit, etc. This view also shows the relationship of the commit to other activities like builds, reviews, and work items.

Commit Defined for TeamForge Orchestrate

Since Orchestrate gathers information from disparate sources, we must define the conditions under which Orchestrate categorizes SCM actions as "commits". Usage of the term "commit" should be construed generically rather than in the context of a specific brand of SCM. For instance, SCMs like Subversion and CVS have a relatively straightforward definition of commit, typically meaning the sending of code changes from working copy to the repository. Git and dynamic version control systems (DVCS) have more nuanced definitions. To be considered a commit in TeamForge

Orchestrate a DVCS action must be public. Therefore, a TeamForge Orchestrate commit encompasses both a Subversion commit and a Git "push", but doesn't include a Git "commit" (which is local).

Commit Detail Attributes

The step associated with the current activity is highlighted. To see the detail for a different activity, click on the associated step.

The Commit Detail displays these SCM commit attributes:

Commit Detail Attribute	Description	Location
SCM repository commit ID	The ID marker denoting a particular revision or version of the SCM repository. For Git, this is the full hash.	Commit activity header
SCM source display name	The user defined "friendly" label for an SCM source repository.	Commit activity header
Committer	Name and user profile picture of the user responsible for the commit transaction. A default user avatar is depicted if the user has not defined a profile picture in TeamForge.	Commit summary
Commit message	The committer's explanatory text submitted at commit time.	Commit summary
Committed At	The date and time of the commit.	Commit summary
Comment	Comment and user profile picture of the user who created the comment. A default user avatar is depicted if the user has not defined a profile picture in TeamForge.	Comment section

☞ **Note:** See *Integrating with TeamForge Orchestrate* for a list of supported build, SCM, and review sources.

Commit Associations

Commits may be associated with other project activities such as builds, reviews, work items. Such commit associations are created by TeamForge Orchestrate in either of these ways:

* Implicitly — For example, when a commit is associated to an automated build or is referenced by a code review.
* Explicitly — For example, by referencing a formatted external ID in the commit message.

The Commit Detail displays this information in the "Associations" section:

* Reviews — Reviews that are associated with this commit are identified by their source display name and review ID. Reviews are linked to the corresponding Review Detail page. Commits are associated with a particular review by commit message reference (explicit) or by inclusion in a post-commit type review (implicit). See *Review detail* on page 612 for information about the review detail.
* Builds — Builds are identified by the source display name and build ID, and are linked to the related Build Detail pages. See *Build detail* on page 609 for information about the build details. Builds are automatically associated with the commits as reported by the build source (implicit).

    ☞ **Note:** The Commit Detail shows builds associated with that commit. If you do not see a particular build, ensure that the selected commit source in the build source configuration is correct.

* Work Items — Work items can be directly associated to commits by referencing the work item artifact ID in the commit message. To explicitly link a commit with a work item, prepend to your commit message:

    `[<artifact_id>]`

    where the *artifact_id* is the ID of the work item in the originating source. You may specify more than one work item in your commit message.

Files and Operations

The "Files and Operations" section lists all files affected by the commit, and the operations performed on those files. Common SCM operations include `add`, `delete`, `modify`, and `rename`.

Review detail

The Review detail shows the current and historical information pertaining to a code review request including attributes like the review request number, status, summary, description, and so on. This view also shows the list of modified files included in the review, and the activities associated with the review like commits, builds, and work items.

Review Detail Attributes

The step associated with the current activity is highlighted. To see the detail for a different activity, click on the associated step.

The Review Detail displays these attributes:

Review Detail Attribute	Description	Location
Review request number	The ID for a specific review request supplied by the review server.	Review request activity header
Review source display name	The user-defined "friendly" label for the source of the code review.	Review request activity header
Review creator or respondent	The default user avatar is depicted.	Review summary
Review summary	Summary or title text from the review request.	Review summary
Description	The full description of the request from the submitter.	Review summary
Reviewed at	The date and time that the review was last updated.	Review summary
Review status	The most-recent state of the review request. State values vary based on review server type. TeamForge Orchestrate classifies all status values into one of four status types: `open`, `submitted`, `rejected`, `discarded`, or `other`.	Review summary
Reviewed by	The list of usernames that have submitted reviews for the review request.	Review summary
Review history	This section details the history of updates to review requests in reverse chronological order, starting with a creation event. TeamForge Orchestrate indicates changes to: • Title • Description (summary) • Status • Review files added or removed • Reviewer responses	Review summary
Comment	Comment and user profile picture of the user who created the comment. A default user avatar is depicted if the user has not defined a profile picture in TeamForge.	Comment section

Review Associations

Reviews may be associated with other activities such as commits as well as TeamForge work items. The Review Detail view displays this information in the "Associations" section:

- Commits — Commits are identified by the commit ID, and may be linked to the corresponding commit details page. Commits are associated with a particular review through the commit message or by inclusion in a post-commit review. To explicitly link a commit with a review, prepend to your commit message:

  ```
  [<association_prefix>:<review_id>]
  ```

  where *<association_prefix>* is defined per review source and the *review_id* is the ID of the review request in the originating code review system. You may specify more than one review in your commit message.

  Missing commit details — TeamForge Orchestrate sometimes learns about commits before the details of these commits are officially reported by the commit source. Such commits are grayed out temporarily in the "Associations" sections of, for example, review and build detail pages until TeamForge Orchestrate receives this information. Traceability graphs include these commits, but you cannot click on the commit nodes to get commit details. See *Troubleshooting TeamForge Orchestrate* for information on commits with missing detail information.
- Builds — Builds are associated with reviews through the commit associations that they both share in common. Builds are identified by the source display name (user inputted) and build ID. Builds are linked to the corresponding Build Detail page. See *Build detail* on page 609 for information about the build details.
- Work Items — Work items are associated with reviews through the commit associations that they both share in common. See *Associate code commits with other items*.

Files for Review

The "Files for Review" section lists all modified files included in the review request.

Work item detail

The Work item detail shows the current and historical information pertaining to a work item artifact including attributes like the artifact ID, status, summary, description, and so on. This view also shows the list of activities associated with the work item like commits, builds, and reviews.

Work Item Detail Attributes

The step associated with the current activity is highlighted. To see the detail for a different activity, click on the associated step.

The Work Item Detail displays these attributes:

Work Item Attribute	Description	Location
Artifact ID	The work item identifier as supplied by the source work item tracking system.	Work item activity header
Work item source display name	The user-defined "friendly name" alias for the work item source.	Work item activity header
Updated on:	The date and time that the work item was updated as reported by the source work item system.	Work item summary
Work item status	The current state of the work item represented by an icon and text description. Text descriptions vary based on the underlying source review system. Status types are shown in parenthesis to further clarify the status intention (e.g., Closed, Open, In Progress).	Work item summary
Work item title	Summary or title text from the work item.	Work item summary
Work item description	The full description field of the work item.	Work item summary
Work item history	This section details the history of updates to work items in reverse chronological order. The work item history may show changes to: • Title and/or Description • Status	Work item summary

Work Item Attribute	Description	Location
	• Priority • User Assignment	
Comment	Comment and user profile picture of the user who created the comment. A default user avatar is depicted if the user has not defined a profile picture in TeamForge.	Comment section

Work Item Associations

Work items may be associated with other activities such as commits, builds and reviews. The Work Item Detail view displays this information in the "Associations" section:

• Commits — Commits can be directly associated to work items by referencing the work item artifact ID in the commit message. To explicitly link a commit with a work item, prepend to your commit message:

    [<artifact_id>]

where the *artifact_id* is the ID of the work item in the originating source. You may specify more than one work item in your commit message.
• Builds — Builds are associated with work items through the commit associations that they both share in common. Builds are identified by the source display name (user inputted) and build ID. Builds are linked to the corresponding Build Detail page. See *Build detail* on page 609 for information about the build details.
• Code reviews — Work items are associated with reviews through the commit associations that they both share in common. See *Associate code commits with other items*.

Create a comment

You can comment on any activity items that you have access to in TeamForge Orchestrate.

TeamForge Orchestrate associates a comment with the user icon that a user has defined in TeamForge. A default user icon is depicted along with the comment if the user has not defined an icon in TeamForge. Users with "Pipeline read" permission for a given project can comment on activities associated with that project.

1. Navigate to the details page for the activity (commit, build, review) that you wish to comment on.
2. Click in the comment text area, and enter your comment. Your comment may contain up to 65,000 characters.
3. Click **Update**.
   Your new comment shows up in the comment list, directly below the comment input area.

   ☞ **Note:** You can also link to any comment by using the link provided in the time-stamp of every individual comment.

Associations

TeamForge Orchestrate creates associations between activities such as reviews, commits, builds, and work items.

Associations are relationships between activities like work items, commits, builds, and reviews. You can see associations for a particular activity on its TeamForge Orchestrate detail page or create a traceability graph of associations using the **Trace associations** button from an activity details page.

Most associations in TeamForge Orchestrate center around commits. That is, builds, work items, and reviews associate directly to commits, while a build-to-review association are implied through a common commit.

Builds are associated to commits by the underlying CI/build system. Typically, CI/build systems obtain source code from a target repository and build based on that specific revision of source code. TeamForge Orchestrate therefore relies on the CI/build system to report the associated revisions, which then get mapped as associations between commits and builds in TeamForge Orchestrate.

Reviews associate to commits in one of two ways, depending on the process being employed and the source code review product. Please see *Review detail* on page 612 for a detailed explanation on commit-to-review associations.

When creating code reviews, note that the code review's repository URL must match the one specified while configuring the related SCM source. Otherwise, TeamForge Orchestrate will not be able to register associations between commits and reviews.

Work items can be directly associated to commits by referencing the work item artifact ID in the commit message. Please see *Work item detail* on page 613 for a detailed explanation on commit-to-work-item associations.

TeamForge Associations

TeamForge Orchestrate adds build and review associations to TeamForge objects like commits and artifacts.

TeamForge Orchestrate draws associations between various lifecycle activities. If TeamForge commits or artifacts are a part of the associations chain, Orchestrate creates associations in the TeamForge object's "associations" area. This allows TeamForge users to see Orchestrate derived associations directly inside a TeamForge record.

Traceability report

View relationships between lifecycle activities using TeamForge Orchestrate's traceability report feature which graphs associations between activities such as TeamForge work items, builds, commits, and reviews.

To use the traceability report, click on "Trace Associations" from work item, commit, build, or review activity detail pages.

> ☞ **Note:** The traceability report only shows association relationships between activities that are monitored by TeamForge Orchestrate.

Traceability report - Radial graph

The traceability report uses a radial graph to show relationships, with the query subject at the center of several concentric circles. The query or root node is represented by its activity widget with information on the activity ID, step number and other specific activity information available upon hover. The complete scope of relationships is mapped - activities that are directly related to the query subject appear on the first circle closest to the center, relationships to those activities appear on the second concentric circle, and so on. Straight lines between mapped activities indicate direct associations between them.

Missing commit details — TeamForge Orchestrate sometimes learns about commits before the details of these commits are officially reported by the commit source. Traceability graphs include these commits, but you cannot click on the commit nodes to get commit details. See *Troubleshooting TeamForge Orchestrate* for information on commits with missing detail information.

Use Cases

The output of a traceability report may be used to explore the relationships between activities in a lifecycle. For instance, you can see all activities related to a particular work item including commits, builds and reviews. This type of reporting is useful for root cause analysis and audit-oriented traceability investigation.

Generate a traceability report

Using the "trace associations" feature of TeamForge Orchestrate, you can generate a graph of all activities associated with a particular work item, build, commit, or review activity.

TeamForge Orchestrate communicates with activity sources to obtain information on the correlations between different activities. You can create a graph of associations for any activity that is monitored by TeamForge Orchestrate.

1. To display a traceability report for an activity, navigate to the specific activity "Details" page.
2. In the "Associations" section, click **Trace Associations**.
   The Traceability report is displayed. The graph displays all commit, build, review, and work item associations for your selected activity.

   > ☞ **Note:** On the traceability report, all related activities are represented by their respective icons; hovering over the icon will reveal the activity ID, step number and other information specific to that activity.

   You can use these controls on a traceability report:

- Hover over a node — View the activity information for the node.
- Click a node — Open, in a new tab, the details page for that activity.
- Control-click a node — Re-plot the report with the selected node as center.

  ☞ **Tip:** With Mac OS X, command-clickthe node to re-plot the report.

- Use the scroll wheel — Zoom the graph in and out.
- Click and drag the graph — Move the graph canvas.

Use filters in traceability reports

You can filter activities by date and select the level of associations you wish to depict on a traceability report. You can also choose to remove "second level commits" from the report.

You must be viewing a traceability report to see the filter icon and **Filter Results** button. The traceability page header indicates current filtering status - whether your activities are being filtered by date, whether indirect commits are included, and what the association depth is. In addition, you also see the specific activity ID for which the traceability report is being depicted and the total number of associations for that activity.

1. Click **Filter Results**.
2. (Optional) Enter valid dates for start and/or end dates; you can use the pop-up calendars. If no date filter is set, TeamForge Orchestrate depicts all activities up to the current date. To remove a previously selected date, click **Clear**.

   Downstream activities associated with an activity outside the date range are omitted even if those downstream activities are themselves within the specified date range. If your date range precludes all activities, an empty report is depicted.

3. (Optional) Enter the association depth, choosing a value from 1 to 5. This value controls the number of levels of associations to the central node displayed in the report. Note that by selecting a higher association depth load times will increase.

   You may see more "rings" on the report than the number of levels you specify although you will only see activities for the specified number of levels.

4. (Optional) Choose **Hide indirect commits** to limit the display to only commits that have a direct relationship to the item being traced. These "second level commits" are often not related to the item being traced, and can therefore be excluded, along with any associations that occur downstream.

5. Click **OK** to save the search criteria and redraw the report or click **Close** to cancel the search.

   To restore the filters to their previous values, you must reload the page.

Installation

TeamForge Orchestrate requires

Installation Overview

TeamForge Orchestrate is a component of TeamForge and requires the installation of new software services. This set of documentation provides you with detailed installation instructions, with a high-level overview provided below. Please read all of the installation documents prior to embarking on the installation process so that you are aware of the prerequisites and steps involved ahead of time.

The installer installs and configures three primary classes of software services.

- Application and web server (App server)
- Database server (DB server)
- Message queue service (MQ server)

During the installation process you may choose to use pre-existing services instead of installing them. The diagram below illustrates the product architecture.

Before you begin

It is assumed that the Orchestrate administrator is familiar with basic Linux administration commands and terminology. Please read *Installation requirements* on page 618 for more information before starting the installation process.

Installation orientation

Read this orientation to get a handle on the scope of the installation and post-install process

The goal of this topic is to give you an idea for the scope of work involved in installing and configuring TeamForge Orchestrate. It serves as a high-level orientation with links to specifics for the express purpose of encapsulating the whole process as concisely as possible. By its nature, TeamForge Orchestrate touches multiple systems and therefore installation and configuration may require cross-functional involvement.

This quick-start orientation does not substitute for complete documentation. Please read all aspects of the installation documentation, including *Installation requirements* on page 618.

Installation of server components

While a feature of TeamForge, Orchestrate requires the installation of new server-side components on a dedicated host.

1. Prep host(s) for installation

   Read *Installation requirements* on page 618 for hardware and software requirements and prepare new host(s).

2. Firewall considerations

   The installation process and subsequent functioning requires firewall access. Read *Installation requirements* on page 618 and make firewall adjustments as needed.

3. Proxy considerations

   The Orchestrate service is designed to run proxied behind TeamForge's web server. The default configuration works for situations where Orchestrate and TeamForge are intended to run on the same machine. However, if you intend to install Orchestrate on a separate host, the TeamForge site-option ORC_HOSTNAME must be uncommented and changed from localhost to the desired hostname. Please read *TeamForge-Orchestrate proxy*

settings for a complete list of proxy configuration options and make any needed changes in TeamForge's `site-options.conf` file before installation begins.

4. Install server-side components

    Now you are ready to run the installation process itself, see *Installation walk-through* on page 620.

TeamForge configuration

Once the server components are installed, configure TeamForge for end-user access to Orchestrate.

5. Create administrative Orchestrate role

    Installation of TeamForge Orchestrate creates four new permissions in TeamForge. Create a new Orchestrate administrator role, or update existing roles with the "Pipeline Create" permission, see *TeamForge documentation*. Add the role to desired users.

6. Create a read-only Orchestrate role

    End-users without admin rights need a read-only role to see TeamForge Orchestrate data and features. Create a new Orchestrate read-only role, or update existing roles with the "Pipeline read" permission, see *TeamForge documentation*. Add the role to desired users.

Install adapters

Now that you have projects and users configured properly, you must make sure firewalls are configured to permit the flow of data from adapters to Orchestrate and install the adapters.

7. Configure firewall rules

    In preparation for adding adapters, the next step requires that firewall rules be created to permit data from adapters to the TeamForge Orchestrate MQ server, see *Installation requirements* on page 618.

8. Install and configure adapters

    Download, install, and configure the corresponding adapter. See *Adapters: Integration using TeamForge Orchestrate* on page 637.

Installation requirements

TeamForge Orchestrate installation requires the following hardware and software services.

Hardware requirements - Single-host installation

This section outlines hardware requirements for installing TeamForge Orchestrate services on a single server. The single-host installation places all three TeamForge Orchestrate services (App server, MQ server, and DB server) on a single operating system. This configuration is designed for trials and moderate load, while the multi-host option outlined below is provided as a first step toward scaling your TeamForge Orchestrate instance.

Prepare for installation by setting up:

- Host running RHEL or CentOS, release 6.7or 7.2
- User credentials with sudo privileges on the host

☞ **Note:** Orchestrate services may be collocated with TeamForge on the same physical server. However, we encourage you to install Orchestrate on its own server for optimal scalability and performance.

The following are minimum resource specification requirements:

Host	CPU	RAM	Storage	Description
App/DB/MQ server	Quad-core 2.5 GHz	8 GB	74 GB	Single host, all services

Hardware requirements - Multi-host installation

For high load instances or when performance is critical, install TeamForge Orchestrate services on separate servers or virtual machines.

Multi-host installation originates on a single server and installs core services and the TeamForge Orchestrate application on that first server (the App server). Then, the installer remotely installs the database service on a second server (the DB server) and the message queue service on a third server (the MQ server).

Prepare for a multi-host installation by setting up:

*   Empty hosts running RHEL or CentOS, release 6.7or 7.2
*   User credentials with sudo privileges on all three hosts
*   ssh routes from the App server host to the other two hosts

The following are minimum resource specification requirements:

Host	CPU	RAM	Storage	Bandwidth	Description
App server	Quad-core 2.4 GHz	4 GB	16 GB	1 Gbps to DB server	Application server, initiates installation process
DB server	Dual-core 2.4 GHz	4 GB	50 GB	1 Gbps to App server	MongoDB database server
MQ server	Dual-core 2.4 GHz	4 GB	16 GB	N/A	AMQP Message queue server

Software requirements and services

*   Proper TeamForge version:

    Orchestrate 8.2 - TeamForge 8.2
*   SSH and SFTP clients are required, i.e. openssh-client. To test for this dependency, on the target server issue the command: `sftp localhost`
*   createrepo package is required by the disconnected mode installation. To test for this dependency, on the target server issue the command: `createrepo --version`
*   Orchestrate has a number of other open source software dependencies, all of which are installed as part of the installation process. During the installation process you may choose to use pre-existing MongoDB or RabbitMQ services instead of installing them, so long as version requirements are satisfied. Aside from that exception, the following software services and their dependencies will be installed:

Product	Version	Home Page	Documentation	Host
Nginx	1.5.1	*http://nginx.org*	*http://nginx.org/en/docs/*	App server
Phusion Passenger	4.0.5	*https:// www.phusionpassenger.com*	*https:// www.phusionpassenger.com/ support*	App server
Redis	2.6.13	*http://www.redis.io*	*http://redis.io/documentation*	App server
MongoDB	2.6	*http://www.mongodb.org*	*http://docs.mongodb.org/manual*	DB server
RabbitMQ	3.1.0	*http://www.rabbitmq.com*	*http://www.rabbitmq.com/ documentation.html*	MQ server

Proxy settings and ports used by TeamForge Orchestrate services

The Orchestrate service is designed to run proxied behind TeamForge's web server. The default configuration works for situations where Orchestrate and TeamForge are intended to run on the same machine. However, if you intend to install Orchestrate on a separate host, the TeamForge site-option `ORC_HOSTNAME` must be uncommented and changed from localhost to the desired hostname. Please read *TeamForge-Orchestrate proxy settings* for a complete list of proxy configuration options and make any needed changes in TeamForge's `site-options.conf` file before installation begins.

Please be aware of the following port requirements to avoid conflicts during installation. Note that the HTTP/HTTPS port listed below is Orchestrate's internal, non-proxied port. Once proxied, Orchestrate will be accessible through TeamForge's HTTP(S) hostname and port.

Port	service	Host
8844	HTTP / HTTPS	App server
6379	Redis	App server
27017	MongoDB	DB server
28017	MongoDB HTTP console	DB server
5672	RabbitMQ	MQ server
15672	RabbitMQ management console	MQ server

Firewall/Routing requirements

The following use cases detail TeamForge Orchestrate's firewall/routing requirements. By default, end-user web access is proxied through the primary TeamForge web server. TeamForge Orchestrate adapters supply data using the MQ layer and therefore need access to the MQ server (default port 5672). There are also private access requirements between the various installed services as detailed below.

Port	From	To	Description
443/80	App server	TeamForge server	App communication with TeamForge server
8844	TeamForge server	App server	TeamForge communication with App server
5672	TeamForge Orchestrate Adapters	MQ server	Message communication between Adapters and MQ server
5672	App server	MQ server	App communication with MQ server
5672	TeamForge server	MQ server	TeamForge communication with MQ server
15672	ALL	MQ server	App administration of MQ server
27017	App server	DB server	App server communication with DB server
22	App server	MQ server	App ssh to MQ server, installation only
22	App server	DB server	App ssh to DB server, installation only

The App server also needs access to the Redis service, but since it is installed on the App server by default, it is not included in the table above.

Installation walk-through
Installation instructions for TeamForge Orchestrate

This guide is a step-by-step walk-through of the TeamForge Orchestrate installation process. **We recommend reading this entire document before starting the installation process; the installer asks several questions and it is beneficial to have the answers handy**. Where applicable, default answers are denoted in square brackets.

As suggested in *Installation requirements* on page 618, prepare your target server(s) prior to installation. We recommend these preparations on the target host(s):

- prepare SSH users and passwords/keys
- ensure SSH users have sudo "root" privileges
- if using SSH keys, locate your SSH private key on the primary installation server (App server) (e.g., /home/user/.ssh/id_rsa).
- locate your SSL key and cert files on the primary installation server (App server)

Configure proxy for web access

1. Before starting installation, TeamForge must be configured to proxy Orchestrate's web traffic. See *Proxying Orchestrate through TeamForge* and *Proxy settings and ports used by TeamForge Orchestrate services* on page 619.

Download and run install_[version].sh

2. To gain access to the TeamForge Orchestrate installation script `install_[version].sh`, contact *CollabNet Support*

3. Once Support has given you access, download `install_[version].sh` from *Open CollabNet*

   where [version] represents the target Orchestrate version.

4. Log into the host designated to be the TeamForge Orchestrate application server (App server) with a privileged account.

5. Copy `install_[version].sh` to `/tmp` and change directories to that location.

   ```
   cp install_[version].sh /tmp
   cd /tmp
   chmod 755 install_[version].sh
   ```

   At this point, if the host(s) you are installing on do not have internet access, follow the steps in *Installation without internet access* on page 625 before continuing with the next step in this guide.

6. Run `install_[version].sh` with root privileges

   ```
   sudo ./install_[version].sh
   ```

   The installer then performs the following actions: installs the CollabNet yum repository, installs the collabnet-orchestrate-core RPM, which causes several dependencies to be installed (passenger, ruby, redis, nginx, gems, etc.).

7. Read the TeamForge Orchestrate license agreement

   ```
   > Do you agree to the terms of the license?
   ```

   Type "Yes" to proceed, or "No" to exit. Once agreed, the installation process continues by asking a series of questions necessary to configure your installation TeamForge Orchestrate.

Installing MongoDB - The first set of questions concerns installation and configuration of MongoDB, the database server.

8. Install MongoDB or use an existing installation

   ```
   > Do you have an existing MongoDB system you would like to use? [No]:
   ```

   Input "No" unless you have an existing instance of MongoDB you prefer to use. If you input "Yes", reference the documentation in *Use an existing MongoDB installation* on page 625 to complete MongoDB configuration, then return to this document and skip to the next section concerning RabbitMQ.

9. Provide the SSH hostname of target DB server

   ```
   > What is the SSH hostname of your MongoDB host:
   ```

   Enter the hostname of the server that has been designated as the database server (DB server). If installing on a single host, use "localhost".

10. Provide the SSH username of target DB server

    ```
    > What is the SSH username for your MongoDB host (NOTE: This user needs sudo
     access):
    ```

    Enter the username to use when installing the MongoDB software on your DB server. This user must be able to connect via SSH to the machine where MongoDB will be installed, even for single-host installations (local SSH). Note, this is the NOT the database user.

11. Provide SSH key to authenticate DB server, if any

```
> Does this account need an SSH key for authentication [No]:
```

Input "Yes" if you have set up an SSH key for authentication. Note: you will still be prompted for a private password (passphrase), if you set one when you created your private key.

a) If you answered "Yes" above, provide the SSH key location

```
> What is the full path to the SSH key you would like to use [~/.ssh/
id_rsa]:
```

Enter the full path to your private SSH key file (Example: /home/preseed/.ssh/id_rsa). This file must be on the local machine where the installer is running.

☞ **Note:** The default here is to use the current user's home directory, which would be root, and may not be the correct SSH key location.

12. Provide the SSH password

```
> What is the password for the USERNAME user on HOSTNAME []:
```

Please provide a password even if using SSH keys, as the installer will run sudo commands.

Installing RabbitMQ - The following set of questions concerns installation and configuration of RabbitMQ, the message queue service.

13. Install RabbitMQ or use an existing installation

```
> Do you have an existing RabbitMQ system you would like to use? [No]:
```

Input "No" unless you have an existing instance of RabbitMQ you prefer to use. If you input "Yes", reference the documentation in *Use an existing RabbitMQ installation* on page 626 to complete RabbitMQ configuration, then return to this document and skip to the next section concerning TeamForge Orchestrate configuration.

14. Provide the SSH hostname of target MQ server

```
> What is the SSH hostname of your RabbitMQ host:
```

Enter the hostname of the server that has been designated as the message queue server (MQ server). If installing on a single host, use "localhost".

15. Provide the SSH username of target MQ server

```
> What is the SSH username for your RabbitMQ host (NOTE: This user needs
 sudo access):
```

Enter the username to use when installing the RabbitMQ software on your MQ server. This user must be able to connect vian SSH to the machine where RabbitMQ will be installed, even for single-host installations (local SSH).

16. Provide SSH key to authenticate MQ server, if any

```
> Does this account need an SSH key for authentication [No]:
```

Input "Yes" if you have set up an SSH key for authentication. Note: you will still be prompted for a private password (passphrase), if you set one when you created your private key.

a) If you answered Yes above, provide the SSH key location

```
> What is the full path to the SSH key you would like to use [~/.ssh/
id_rsa]:
```

Enter the full path to your private SSH key file (Example: /home/preseed/.ssh/id_rsa). This file must be on the local machine where the installer is running.

☞ **Note:** The default here is to use the current user's home directory, which would be root, and may not be the correct SSH key location.

17. Provide the SSH password

```
> What is the password for the USERNAME user on HOSTNAME []:
```

Please provide a password even if using SSH keys, as the installer will run sudo commands.

18. Provide the FQDN of target MQ server

```
> What is FQDN or routable IP of your RabbitMQ host:
```

Enter the hostname of the server that has been designated as the message queue server (MQ server). The MQ component must be network accessible by adapters supplying data to Orchestrate. **"localhost" or "127.0.01" values are not permitted, even for single-host installations**. We also strongly suggest using a distinct DNS hostname or alias for the MQ component to make host separation easier for scaling in the future.

Configuring TeamForge Orchestrate - The following set of questions concerns the configuration of TeamForge Orchestrate.

19. Provide the SSH username of target App Server (the local machine)

```
> What is the SSH username for your Orchestrate host (localhost) NOTE: This
 user needs sudo privileges:
```

Enter the username to use when installing the Orchestrate application on the local server. SSH credentials with sudo root privileges are required at this step even though you are likely installing the App server locally.

20. Provide SSH key to authenticate App server, if any

```
> Does this account need an SSH key for authentication [No]:
```

Input "Yes" if you have set up an SSH key for authentication. SSH credentials are required even if this is a local installation. Note: you will still be prompted for a private password (passphrase), if you set one when you created your private key.

a) If you answered "Yes" above, provide the SSH key location

```
> What is the full path to the SSH key you would like to use [~/.ssh/
id_rsa]:
```

Enter the full path to your private SSH key file (Example: /home/preseed/.ssh/id_rsa). This file must be on the local machine where the installer is running.

☞ **Note:** The default here is to use the current user's home directory, which would be root, and may not be the correct SSH key location.

21. Provide the SSH password

```
> What is the password for the USERNAME user on HOSTNAME []:
```

Please provide a password even if using SSH keys, as the installer will run sudo commands.

22. Provide a hostname for your TeamForge Orchestrate service

```
> What hostname should we use for the Orchestrate web service:
```

You should be running the installer on the server designated as the Orchestrate App server; so the answer here would be the DNS hostname entry for the local host. If you're using SSL, the hostname supplied should agree with your certificate. Note that by default end-users will be proxied through the main TeamForge web services, which may be a different DNS name.

23. Provide port for the Orchestrate web service

```
> What server port should Orchestrate listen on [8844]:
```

By default, the Orchestrate service is proxied through the main TeamForge web server, so end- users will not use this port. However, we recommend sticking with the default 8844. If you modify this default, make sure the new value agrees with the TeamForge site token `ORC_PORT` in your TeamForge site-options.conf file.

24. SSL option

```
> Should Orchestrate use SSL?: [Yes]
```

Keeping in mind that end-users will access Orchestrate proxied through TeamForge, select whether you wish to protect the communication between Orchestrate and the TeamForge server with SSL If yes, ensure the value of the TeamForge site token `ORC_PROTOCOL` is `https` in your TeamForge site-options.conf file.

25. Provide location of SSL key (skipped if no SSL)

```
> Where is the SSL key you want to use for Orchestrate located:
```

This key file will be used to configure Orchestrate's web server for . Enter the full path of your locally stored SSL key file (Example: `/path/to/server.key`). Self-signed certificates are functional.

> ☞ **Important:** The SSL certificate may not be password protected or the installation process will fail with an error.

26. Provide location of SSL certificate (skipped if no SSL)

```
> Where is the SSL certificate you want to use for Orchestrate located:
```

Enter the full path of your locally stored SSL cert file (Example: `/path/to/server.crt`) and ensure the same certificate file is used with the TeamForge site token `ORC_SSL_CA_CERT_FILE` in your TeamForge site-options.conf file. Self-signed certificates are functional.

27. Provide location of CA certificate chain (skipped if no SSL)

```
> Where is the CA certificate chain you want to use for Orchestrate located:
```

Enter the full path of your locally stored CA certificate chain file. This file will be provide by the SSL certificate signing authority used to generate your SSL certificate. If you are using a self-signed certificate then a CA certificate is not required; however, the installation process needs a file here in order to continue, just create an empty file and input the path and name.

```
touch /path/to/emptyfile
```

Configuring TeamForge - The following set of questions concerns the configuration of TeamForge.

28. Provide your TeamForge URL

```
What is the URL for your TeamForge server:
```

Enter the URL for the TeamForge instance associated with this TeamForge Orchestrate installation.

29. Provide a TeamForge administrative user account

```
> What administrative user would you like to use to join TeamForge
 Orchestrate to TeamForge:
```

The installation process requires a TeamForge administrative account so to execute the integration of Orchestrate into TeamForge.

Add this user to the TeamForge site token USERS_WITH_NO_EXPIRY_PASSWORD in your TeamForge site-options.conf file.

30. Provide the TeamForge administrative user's password

```
> What is the password for the user you entered:
```

The installer will now perform a series of operations; installing and/or configuring MongoDB, RabbitMQ, and the Orchestrate server. When the installer finishes, Orchestrate will be installed and accessible using the TeamForge Orchestrate hostname configured above.

Installation without internet access
This guide applies to customers installing behind a firewall or without a connection to the internet.

This article provides supplemental installation instructions if you are installing TeamForge Orchestrate on a host(s) with restricted or no access to the internet. Follow the installation instructions in *Installation walk-through* on page 620 through step *5* on page 621. After following instructions here, continue with *Installation walk-through* on page 620, picking up with step *6* on page 621.

1. Contact *CollabNet Support* and request the TeamForge Orchestrate disconnected media package.

   The file is named according to the pattern `collabnet-orchestrate-disconnected-media-<Version>.el7.x86_64.rpm`, where <Version> is replaced by numbers.

2. Copy the disconnected media package onto each host you intend to use in the installation process.

   For each host you wish to use in the installation, copy the disconnected media rpm to the `/tmp` directory:

   ```
   cp collabnet-orchestrate-disconnected-media-<Version>.el7.x86_64.rpm /tmp
   ```

3. Disable all OS yum repositories to prevent yum from making calls to the Internet.

   ```
   cd /etc/yum.repos.d
   ```

   Within each .repo file, edit each repository and set the `enabled` property to "0". Add the following line to any repository without the `enabled` property:

   ```
   enabled=0
   ```

4. Unpack the disconnected media package on target host(s).

   ```
   sudo rpm -ivh collabnet-orchestrate-disconnected-media-<Version>.el7.x86_64.rpm
   ```

   Continue with installation

5. Once complete, continue with *Installation walk-through* on page 620, picking up with step *6* on page 621.

Use an existing MongoDB installation
Follow this process if you wish to use an existing MongoDB server with TeamForge Orchestrate

This article provides alternate installation instructions if you wish to use an existing MongoDB server with TeamForge Orchestrate. Follow the installation instructions in *Installation walk-through* on page 620 until it refers you to this article. After following instructions here, continue with *Installation walk-through* on page 620, picking up with the *RabbitMQ* section

1. Install MongoDB or use an existing installation

   ```
   > Do you have an existing MongoDB system you would like to use? [No]:
   ```

   The following steps assume you answer "Yes". If your answer is "No", skip these steps and refer to the primary installation guide: *Installation walk-through* on page 620.

2. Determine whether your MongoDB is clustered

   ```
   > Is your MongoDB clustered (replset) [No]:
   ```

   Answer Yes here if your MongoDB database is using sharding, which distributes a single logical database system across a cluster of machines. Answer "No" here if your MongDB runs on a single machine.

   a) If your MongoDB is clustered, provide hosts

   ```
   > What are the hostnames and ports for your Mongo cluster nodes (comma-separated):
   ```

Enter the server and port for each MongDB server in your cluster.

```
mongo1.example.com:27017,mongo2.example.com:27017
```

b) If your MongoDB is a single host, provide hostname

```
> What is the hostname for your mongo server [localhost]:
```

Enter the name of the server where your existing MongoDB server is currently running.

c) If your MongoDB is a single host, provide port number

```
> What port is MongoDB running on [27017]:
```

Enter the port being used by your existing MongoDB server. The default here is the standard port used by MongoDB.

3. Provide a database name for TeamForge Orchestrate use

```
> What is the name of the database you wish to use [orchestrate]:
```

Enter the name of the database to use. This name must be unique within your existing MongoDB server; use a name that is currently unused.

4. Provide a MongoDB user for accessing the TeamForge Orchestrate database

```
> What username should we use to connect to this database [collabnet-
mongodb]:
```

Enter a username to use when Orchestrate makes connections to the database server.

a) If this user does not currently exist you need to create this user for the specified database. Here are the steps to add the user to your existing MongoDB:

- connect to your MongDB server using the mongo shell as an admin user that can create databases and users
- run the following command:

```
db.getSiblingDB('MONGODB_ORCHESTRATE_DB_NAME').addUser({user: 'USER',
pwd: 'PASS', roles: ['readWrite', 'dbAdmin']})
```

- replace MONGODB_ORCHESTRATE_DB_NAME with the database name you specified for the database in the previous installer question
- replace USER with the username you chose in the previous installer question
- replace PASS with the password you chose in the previous installer question

Continue with *Installation walk-through* on page 620, picking up with the *RabbitMQ* section.

Use an existing RabbitMQ installation

Follow this process if you wish to use an existing RabbitMQ server with TeamForge Orchestrate

This article provides alternate installation instructions if you wish to use an existing RabbitMQ server with TeamForge Orchestrate. Follow the installation instructions in *Installation walk-through* on page 620 until it refers you to this article. After following instructions here, continue with the *configuration* section.

1. Install RabbitMQ or use an existing installation

```
> Do you have an existing RabbitMQ system you would like to use? [No]:
```

The following steps assume you answer "Yes". If your answer is "No", skip these steps and refer to the primary installation guide: *Installation walk-through* on page 620.

2. Provide a RabbitMQ hostname

```
> What is your RabbitMQ host [localhost]:
```

Enter the hostname of your existing RabbitMQ server.

3. Provide a RabbitMQ port number

```
> What port is RabbitMQ running on [5672]:
```

Enter the port being used by your existing RabbitMQ server. The default here is the standard port used by RabbitMQ.

4. Provide a RabbitMQ vhost

```
> What is your RabbitMQ vhost [/]:
```

Enter the desired vhost on your existing RabbitMQ server. The default here is the standard vhost used by RabbitMQ.

5. Provide a RabbitMQ username for use by Orchestrate to communicate with adapters:

```
> Provide a RabbitMQ username Orchestrate should use to communicate with
 adapters [guest]:
```

Enter the RabbitMQ username that Orchestrate should use to read and write information to and from Orchestrate related queues.

To setup a new RabbitMQ user with proper permissions:

```
sudo PATH_TO_RABBIT_CTL/rabbitmqctl add_user USERNAME PASSWORD sudo
PATH_TO_RABBIT_CTL/rabbitmqctl set_permissions -p VHOST USERNAME
"^orchestrate.*" "^orchestrate.*|amq.default" "^orchestrate.*|amq.default"
```

where `PATH_TO_RABBIT_CTL`, `VHOST`, `USERNAME` and `PASSWORD` should be replaced by your actual values.

6. Provide associated RabbitMQ password

```
> What is the password for this user? [guest]:
```

Enter the password for the user specified in the previous installer question.

7. Provide a RabbitMQ username with permission to create new users with write access

```
> Provide a RabbitMQ username that has permission to create new RabbitMQ
 users with write access to Orchestrate queues:
```

Supply a username that has permissions to create new users that have write access to Orchestrate queues. When creating data sources, TeamForge Orchestrate creates RabbitMQ users that have permissions to put data onto Orchestrate related queues. Creating RabbitMQ credentials on demand simplifies the process of disseminating queue credentials to users setting up adapters.

To setup a new RabbitMQ user with proper permissions:

```
sudo PATH_TO_RABBIT_CTL/rabbitmqctl add_user USERNAME PASSWORD sudo
PATH_TO_RABBIT_CTL/rabbitmqctl set_user_tags USERNAME administrator
```

where `PATH_TO_RABBIT_CTL`, `VHOST`, `USERNAME` and `PASSWORD` should be replaced by your actual values.

8. Provide associated RabbitMQ password

```
> What is the password for this user?:
```

Enter the password for the user specified in the previous installer question.

Continue with *Installation walk-through* on page 620, picking up with the *configuration* section.

Troubleshooting installation
You may encounter these issues and questions when installing TeamForge Orchestrate

| **How do I download TeamForge Orchestrate?** | Follow the download instruction in the *Installation walk-through* on page 620 |

Is there a disconnected/offline installation process for TeamForge Orchestrate?	Yes, follow the instruction in the *Installation walk-through* on page 620. After step *5* on page 621, link to the article *Installation without internet access* on page 625 for supplementary instructions on obtaining and installing TeamForge Orchestrate using the disconnected media package.
I don't see an option to run TeamForge Orchestrate without SSL.	SSL is a requirement at this time. You can use a self-signed certificate if you don't have one signed by a recognized authority.
If I'm using SSH keys, why do I also need to provide the user's password?	The user's password is required to successfully sudo to root privileges on the target host.
I've installed TeamForge Orchestrate but I don't see an "Orchestrate" tab in my project tool bar.	Normally all active projects automatically get the Orchestrate tab in the project tool bar. If Orchestrate is not available for your project, you can manually enable Orchestrate for your project.
	Log in to TeamForge as a project (or site) admin.
	Choose the TeamForge project to associate to TeamForge Orchestrate.
	Click **Project Admin** in the project toolbar.
	Navigate to **Project Toolbar** > **Integrated Applications**
	Click **Add**. Choose TeamForge Orchestrate and click **Configure & Add**
	Click **Save**. Repeat for any additional project that you'd like to configure to use TeamForge Orchestrate.
How do I get the queue server hostname and credentials? I need it to set up my adapter.	Adapters use a message queue server to transmit data to TeamForge Orchestrate. Adapters must therefore be configured with the queue server hostname, username, and password.
	When you create a new "Source" in TeamForge Orchestrate, a unique set of queue server credentials is created for use with your adapters. To find the queue server hostname and credentials, log into TeamForge as a privileged Orchestrate user and navigate to: **Orchestrate (from the project navigation bar)** > **Manage Sources**. The **Sources Overview** page appears.
	Select one of the adapters on the left hand side. The 'Source List' shows up on the right hand side.
	From the list of sources, click the **EDIT** button of the desired source.
	Locate the section named "Adapter Configuration Information".
	Copy the queue server's hostname, username, and password and use them to configure your adapter.
	Note that the same queue server hostname, username and password may be shared by multiple sources; however, the source association must be unique for each source.
I've set up build/review adapters but I'm not seeing any data in the activity stream.	Make sure you've configured firewall access properly. The adapters must be able to send data to the MQ server on the designated port (default 5672).

Installer fails with user permission errors

The installer assumes that the user installing orchestrate has permission to run commands as any user. If the user has a limited set of permissions, then the installer will fail with the following:

```
failed: "env PATH=/sbin:$PATH sh -c
 'cd /opt/collabnet/rabbitmq && sudo
 -p '\\''sudo password: '\\'' -u
 collabnet-rabbitmq /opt/collabnet/
rabbitmq/sbin/rabbitmqctl list_users'"
 on hostname
```

Ensure that the user installing Orchestrate has permission to run commands as the following set of users:

```
collabnet-redis, collabnet-nginx,
 orchestrate, collabnet-rabbitmq,
 collabnet-mongodb
```

These users are created by the installer and used to setup the various services.

I received an error related to zlib while installing on RHEL 6.3.

The following error is related to a documented bug in RHEL 6.3:

```
---> Package collabnet-ruby.x86_64
 0:1.9.3_p392-18.el6 will be installed
--> Processing Dependency:
 libz.so.1(ZLIB_1.2.2)(64bit)
 for package: collabnet-
ruby-1.9.3_p392-18.el6.x86_64
--> Finished Dependency Resolution
Error: Package: collabnet-
ruby-1.9.3_p392-18.el6.x86_64
 (collabnet)
           Requires:
 libz.so.1(ZLIB_1.2.2)(64bit)
 You could try using --skip-broken to
 work around the problem
 You could try running: rpm -Va --
nofiles --nodigest
```

The correct this issue, you must upgrade zlib on your system to a version that also provides zlib 1.2.2. The zlib 1.2.3.27 package does not bundle 1.2.2, but version 1.2.3-29 does include 1.2.2. There are many alternative paths for upgrading to 1.2.3-29, this article covers but one way.

```
wget http://public-yum.oracle.com/
repo/OracleLinux/OL6/latest/x86_64/
zlib-1.2.3-29.el6.x86_64.rpm sudo rpm
-Uvh zlib-1.2.3-29.el6.x86_64.rpm
```

Confirm that you have zlib 1.2.2:

```
rpm -q --provides lib
```

Restart the installation process from the beginning.

Upgrading TeamForge Orchestrate

This section provides steps for upgrading TeamForge Orchestrate installations to the latest release.

Upgrade Overview

The upgrade process varies for single host and multi host installations of Orchestrate. As such, this section contains separate documentation for each:

- *Single host upgrades* on page 630
- *Multi host upgrades* on page 631

Proxy configuration steps after upgrading from 1.4.1 (or earlier)

Starting with version 8.1, Orchestrate must be configured to proxy through TeamForge's web server. This provides users with a single hostname/port combination for accessing TeamForge and Orchestrate. It is required since some new features (like the JIRA adapter) rely on the proxy.

See *Proxy upgrade configuration* on page 634 after upgrading to set up proxying.

Data loss potential during upgrades

While we have designed the upgrade process to minimize data loss, there is still a chance that Orchestrate misses data from sources during an upgrade.

During normal operation, adapters communicate source data (such as builds and commits) directly with RabbitMQ, the message queue layer between your data sources and Orchestrate. Orchestrate then consumes these messages from the queue as fast as it can. Disruption to RabbitMQ or the database layer (MongoDB) may lead to data loss as follows:

- RabbitMQ cannot receive messages. If RabbitMQ is down, adapters are unable to deliver messages and Orchestrate never receives those "lost" messages.
- MongoDB cannot store received messages. If MongoDB is down, the Orchestrate queue reading processes unsuccessfully attempt to write the messages to the database and the message is lost.

Risk from these two scenarios can be mitigated by upgrading MongoDB and RabbitMQ components at times of low traffic (i.e., after work hours).

Backup your data

Before upgrading, it is a good idea to backup TeamForge Orchestrate system and data, see *Backup and Restore* on page 660.

Single host upgrades

This article provides steps for upgrading single host TeamForge Orchestrate installations to the latest release.

The following steps pertain to TeamForge Orchestrate upgrades when all services are located on a single machine. Upon successful completion, TeamForge Orchestrate will be upgraded to the latest available release.

☞ **Important:** The upgrade process stops the TeamForge Orchestrate App server while upgrading. Anticipate up to 20 minutes of end-user downtime from the time services are shut down.

Download upgrade_[version].sh

1. To gain access to the TeamForge Orchestrate upgrade script `upgrade_[version].sh`, contact CollabNet Support at *https://www.collab.net/support/secure-customer-login*
2. Once Support has given you access, download `upgrade_[version].sh` from *https://ctf.open.collab.net/sf/sfmain/do/viewProject/projects.sfdl*

   where [version] represents the upgrade target version of Orchestrate.

Back up the TeamForge Orchestrate system

3. Before upgrading, backup TeamForge Orchestrate system and data, see *Backup and Restore* on page 660

Upgrade process

4. Log into the TeamForge Orchestrate application server (App server) with a privileged account.
5. Copy `upgrade_[version].sh` to `/tmp` and change directories to that location.

```
cp upgrade_[version].sh /tmp cd /tmp chmod 755 upgrade_[version].sh
```

6. If you are using "disconnected media" to upgrade, follow the steps in *Upgrading without internet access* on page 632

7. Perform the upgrade by running upgrade_[version].sh, passing the all flag.

```
sudo ./upgrade_[version].sh all
```

> ☞ **Note:** The TeamForge Orchestrate web interface will be unavailable to end-users until after the upgrade process.

Services will be restarted automatically upon completion.

8. Verify that all services are running after upgrade_[version].sh terminates

The upgrade script should display a list of all services with [OK] denoting a successful restart.

Multi host upgrades

This article provides steps for upgrading multi-host TeamForge Orchestrate installations to the latest release.

The following steps pertain to TeamForge Orchestrate upgrades when all services are located on multiple hosts. Upon successful completion, TeamForge Orchestrate will be upgraded to the latest available release.

> ☞ **Important:** The upgrade process stops the TeamForge Orchestrate App server while upgrading. Anticipate up to 20 minutes of end-user downtime from the time services are shutdown.

Download upgrade_[version].sh

1. To gain access to the TeamForge Orchestrate upgrade script upgrade_[version].sh, contact CollabNet Support at *https://www.collab.net/support/secure-customer-login*

where [version] represents the upgrade target version of Orchestrate.

2. Once Support has given you access, download upgrade_[version].sh from *https://ctf.open.collab.net/sf/sfmain/do/viewProject/projects.sfdl*

3. Copy upgrade_[version].sh to /tmp on each host in question and change directories to that location.

```
cp upgrade_[version].sh /tmp cd /tmp chmod 755 upgrade_[version].sh
```

Back up the TeamForge Orchestrate system

4. Before upgrading, backup TeamForge Orchestrate system and data, see *Backup and Restore* on page 660

Disconnected media

5. If you are using "disconnected media" to upgrade, follow the steps in *Upgrading without internet access* on page 632. Otherwise continue with the MongoDB section below.

> ☞ **Important:** It is imperative that MongoDB and RabbitMQ be upgraded before the Orchestrate service to avoid potential data loss due to downed services. Please proceed in the order prescribed below.

Upgrade MongoDB - Before upgrading MongoDB, to prevent unintentional data loss, it is important that you shutdown the services running on the App server.

6. Log into the host running the TeamForge Orchestrate application (App server) with a privileged account.

7. Shutdown the "orchestrate" service, see *System Administration* on page 657.

```
sudo service orchestrate stop
```

> ☞ **Note:** The TeamForge Orchestrate web interface will be unavailable to end-users until after the upgrade process. RabbitMQ will remain up though, collecting and queuing data from sources.

8. Log into the host running MongoDB (DB server) with a privileged account.

9. Perform upgrade by running upgrade_[version].sh, passing the mongodb flag.

```
sudo ./upgrade_[version].sh mongodb
```

Upon success, MongoDB will be automatically restarted.

10. You may restart the Orchestrate services on the App server at this time, or continue to the RabbitMQ section.

```
sudo service orchestrate start
```

Upgrade RabbitMQ

11. Log into the host running RabbitMQ (MQ server) with a privileged account.

12. Perform upgrade by running `upgrade_[version].sh`, passing the `rabbitmq` flag.

```
sudo ./upgrade_[version].sh rabbitmq
```

Upon success, RabbitMQ will be automatically restarted.

Upgrade TeamForge Orchestrate

13. Log into the host running the TeamForge Orchestrate application (App server) with a privileged account.

14. Perform upgrade by running `upgrade_[version].sh`, passing the `orchestrate` flag.

```
sudo ./upgrade_[version].sh orchestrate
```

Upon success, Orchestrate services will be restarted automatically.

15. Verify that all services are running after upgrade_[version].sh terminates

The upgrade script should display a list of all services with `[OK]` denoting a successful restart.

Upgrading without internet access

This articles applies to upgrades behind a firewall or without a connection to the internet.

This article provides supplemental installation instructions if you are upgrading TeamForge Orchestrate on a host(s) with restricted or no access to the internet. Begin by reading the appropriate upgrade documentation, depending on whether you are upgrading a single host or multi host TeamForge Orchestrate instance:

- *Single host upgrades* on page 630
- *Multi host upgrades* on page 631

The above articles will reference this document at the appropriate junctures.

1. Contact *CollabNet Support* and request the TeamForge Orchestrate disconnected media package.

The file is named according to the pattern `collabnet-orchestrate-disconnected-media-<Version>.el6.x86_64.rpm`, where <Version> is replaced by numbers.

2. Copy the disconnected media package onto each host you intend to use in the installation process.

For each host you wish to use in the installation, copy the disconnected media rpm to the `/tmp` directory:

```
cp collabnet-orchestrate-disconnected-media-<Version>.el6.x86_64.rpm /tmp
```

3. Disable all OS yum repositories to prevent yum from making calls to the Internet.

```
cd /etc/yum.repos.d
```

Within each .repo file, edit each repository and set the `enabled` property to "0". Add the following line to any repository without the `enabled` property:

```
enabled=0
```

4. Upgrade the disconnected media package on target host(s).

```
sudo rpm -Uvh collabnet-orchestrate-disconnected-media-<Version>.el6.x86_64.rpm
```

Continue with upgrade

5. Once complete, continue with the upgrade process.

- *Single host upgrades* on page 630 picking up with step *7* on page 631
- *Multi host upgrades* on page 631 picking up with step *6* on page 631

Proxy upgrade overview

Orchestrate is a key component of TeamForge which extends associations and traceability to third party products like Jenkins, JIRA, and others. Unlike earlier TeamForge releases, TeamForge 8.1 (and later) is equipped with a set of Orchestrate specific `site-options.conf` tokens, which if configured, facilitates the proxying of Orchestrate through TeamForge's web server.

☞ **Important:** Proxying Orchestrate via TeamForge's web server is required and has a number of benefits. Without proxying, JIRA integration, for example, is not possible. End-users have a more consistent experience when TeamForge and Orchestrate use the same host name/port. Another derived benefit of proxying is the ability to run both Orchestrate and TeamForge on the same server. While you can run both Orchestrate and TeamForge on the same server, CollabNet recommends such an approach only for testing purposes. It's always recommended to run Orchestrate on a separate server for optimal scalability.

The following table lists the Orchestrate specific `site-options.conf` tokens.

`site-options.conf` token	Default value
ORCHESTRATE_ENABLED	true
ORC_PROXIED_PATH	/orc
ORC_PORT	8844
ORC_HOSTNAME	localhost
ORC_PROTOCOL	http
ORC_SSL_CA_CERT_FILE	This token has no default value. You must enter the path of the SSL cert file once you set ORC_PROTOCOL token to https, which you must if you have Orchestrate.

☞ **Note: About the CA certificate file**

The certificate file used for ORC_SSL_CA_CERT_FILE token can be the CA certificate file or the CA bundle. For certificates that are signed by a common CA, this may already be installed by the yum package 'ca-certificates', or can be obtained from a common source, such as https://raw.githubusercontent.com/bagder/ca-bundle/master/ca-bundle.crt.

If the certificate is self-signed, this file is likely the certificate itself.

For more information, see the 'Prepare TeamForge' section of the *Proxy upgrade configuration* procedure.

You can install Orchestrate on the TeamForge application server (ORC_HOSTNAME=localhost, which is the default) or you can install Orchestrate on a separate server. You must set the values of the Orchestrate specific `site-options.conf` tokens depending on whether Orchestrate and TeamForge are on the same server or on two separate servers. The following tables list the tokens and their values for Orchestrate on the TeamForge application server and separate server scenarios.

Orchestrate and TeamForge on the same server

`site-options.conf` token	Default value
ORCHESTRATE_ENABLED	true
ORC_PROXIED_PATH	/orc
ORC_PORT	8844
ORC_HOSTNAME	localhost
ORC_PROTOCOL	https
ORC_SSL_CA_CERT_FILE	<Path of the SSL cert file>

Orchestrate and TeamForge on two separate servers

`site-options.conf` token	Default value
ORCHESTRATE_ENABLED	true
ORC_PROXIED_PATH	/orc
ORC_PORT	8844

site-options.conf token	**Default value**
ORC_HOSTNAME	<Orchestrate server's host name>
ORC_PROTOCOL	https
ORC_SSL_CA_CERT_FILE	<Path of the SSL cert file>

☞ **Important:**

- You must recreate the runtime environment and start TeamForge after setting up the site-options.conf tokens as discussed earlier in this topic.
- See *Proxy upgrade configuration* if you already have Orchestrate and are upgrading to TeamForge 8.2.

Proxy upgrade configuration

Instructions for configuring Orchestrate to run behind TeamForge's web server.

To provide a seamless experience for users, the Orchestrate HTTP(S) service is proxied through TeamForge by default as of version 8.1, meaning end users access Orchestrate using the TeamForge hostname and port. This proxy configuration must be set up manually for those upgrading from Orchestrate versions 1.4.1 or before. This document provides a step-by-step guide for configuring the proxy. Failure to configure the proxy will interfere with functionality and is therefore a required part of the upgrade process.

Assumptions - for purposes of this guide, assume the following:

- An existing TeamForge host using SSL named ctf.example.com
- An existing Orchestrate host using SSL named orc.example.com
- All hosts have been upgraded to the target version (e.g. 8.1), see *Upgrading TeamForge Orchestrate* on page 629
- All hosts are running SSL services on port 443
- Firewall permits traffic on port 443 between TeamForge and Orchestrate hosts
- Hosts have existing certificate files
- Certificate files are signed by "legitimate" CAs
- Hosts are otherwise using their 'standard' production installs (e.g., Orchestrate installs nginx configs in /opt/collabnet/nginx/conf/collabnet-nginx.conf, TeamForge uses /opt/collabnet, etc.)
- The user has sudo or equivalent access on both hosts

Prepare TeamForge (without certificate validation)

1. Configure TeamForge site-options. Comment out the *ORC_SSL_CA_CERT_FILE* token in the TeamForge host's site-options.conf file if you are not enabling certificate validation between Orchestrate and TeamForge.

   Supply the following values to the TeamForge host's site-options.conf (whichever file your host is using as the source of its site options):

   ```
   ORCHESTRATE_ENABLED=true
   ORC_PROXIED_PATH=/orc
   ORC_PORT=443
   ORC_HOSTNAME=orc.example.com
   ORC_PROTOCOL=https
   #ORC_SSL_CA_CERT_FILE
   ```

2. Re-create TeamForge runtime

Prepare TeamForge (with certificate validation)

3. Locate or create CA certificate files

   If the TeamForge host does not already have a root CA bundle which can validate Orchestrate's certificate, create or download such a file. Options include:

   - For certificates that are signed by a common CA, this may already be installed by the yum package 'ca-certificates';

- Obtain from a common source, such as *https://raw.githubusercontent.com/bagder/ca-bundle/master/ca-bundle.crt*;
- If the certificate is self-signed, this file is likely the certificate itself;
- Otherwise, this file can be created by concatenating the certificates of the CA that signed Orchestrate's certificate, along with the certificates of all intermediary signing CAs.

See *http://httpd.apache.org/docs/2.4/mod/mod_ssl.html#sslproxycacertificatefile* for more info on precisely what the root CA certificate file is expected to contain.

If the CA bundle file is not already available/accessible on the TeamForge host, copy it to a suitable location on the TeamForge host, e.g., `/var/ops/ssl/ca-bundle.crt`.

4. Test the CA certicate file

The following is a simple command line sanity check, run from the TeamForge server.

```
[root@ctf.example.com]$ echo |openssl s_client -connect orc.example.com:443
-CAfile /var/ops/ssl/ca-bundle.crt > /dev/null depth=2 /C=US/
O=GeoTrust Inc./CN=GeoTrust Global CA verify return:1 depth=1 /
C=US/O=GeoTrust, Inc./CN=RapidSSL CA verify return:1 depth=0 /
serialNumber=sXupsaptbZJc4PUuawYNAYey2fPMfuXA/OU=GT99362291/OU=See
www.rapidssl.com/resources/cps (c)14/OU=Domain Control Validated -
RapidSSL(R)/CN=*.example.com verify return:1 DONE
```

You are looking for the presence of `verify return:1` in each step of the certificate validation chain. If the certificate was unable to be validated you may instead see:

```
[root@ctf.example.com]$ echo |openssl s_client -connect orc.example.com:443
-CAfile /var/ops/ssl/ca-bundle.crt > /dev/null verify error:num=20:unable to
get local issuer certificate verify return:0
```

5. Configure TeamForge site-options

Supply the following values to the TeamForge host's site-options.conf (whichever file your host is using as the source of its site options):

```
ORCHESTRATE_ENABLED=true
ORC_PROXIED_PATH=/orc
ORC_PORT=443
ORC_HOSTNAME=orc.example.com
ORC_PROTOCOL=https
ORC_SSL_CA_CERT_FILE=/var/ops/ssl/ca-bundle.crt
```

6. Re-create TeamForge runtime

Prepare Orchestrate

7. Edit Orchestrate's nginx configuration

Edit `/opt/collabnet/nginx/conf/collabnet-nginx.conf` to have the following (omitting the non-relevant bits):

```
...
  server {
    listen 443;
    server_name orc.example.com;
    ...
    # Redirect root to /orc
    location / {
      rewrite ^ https://orc.example.com:443/orc$request_uri? permanent;
    }
  }
  ...
```

```
    # This provides a HTTP->HTTPS redirect for orchestrate
    server {
      listen 80;
      server_name orc.example.com;
      ...
      # redirect 80 to 443
      location / {
        rewrite ^ https://orc.example.com:443$request_uri? permanent;
      }
    }
    passenger_pre_start https://orc.example.com:80/orc/;
    passenger_pre_start https://orc.example.com:443/orc/;
    ...
```

8. Test nginx configuration

   Test the edited configuration before restarting. The example below assumes a user orchestrate with sudo access:

   ```
   [orchestrate@orc.example.com]# sudo /opt/collabnet/nginx/sbin/nginx -c /
   opt/collabnet/nginx/conf/collabnet-nginx.conf -t nginx: the configuration
   file /opt/collabnet/nginx/conf/collabnet-nginx.conf syntax is ok nginx:
   configuration file /opt/collabnet/nginx/conf/collabnet-nginx.conf test is
   successful
   ```

9. Edit Orchestrate's defaults.yml

   Edit Orc's /opt/collabnet/orchestrate/config/defaults.yml to ensure that the teamforge.production.server and teamforge.production.iaf.orc_url have the correct proxied URLs (omitting irrelevant bits):

   ```
     ...
    teamforge:
      production:
        server: https://ctf.example.com
        iaf:
          ...
          orc_url: https://ctf.example.com/orc
    ...
   ```

10. Restart nginx

    ```
    [orchestrate@orc.example.com]#sudo service collabnet-nginx restart
    ```

Update TeamForge's IAF settings

11. Update the existing IAF and Site-wide linked app

    This assumes that the Orc service runs under the user orchestrate on the orchestrate host, and can bring up a rails console (hence use of sudo, and the console invocation). Use that user to bring up a console, and manually call CtfConfigurer.new.edit_iaf_application:

    ```
    [orchestrate@orc.example.com]#cd /opt/collabnet/orchestrate
    sudo su - orchestrate bundle exec rails c production
    rb(main):003:0> CtfConfigurer.new.edit_iaf_application =>
    {:set_integrated_application_icon_response=>nil}
    ```

12. Update existing Site-wide linked app

    This step requires login to the TeamForge web application

    a) In TeamForge, as a site-admin navigate to: **Admin** > **Integrations** > **Site-wide Linked Applications**.
    b) Click the **Orchestrate** application to edit.
    c) Edit the URL to the one being proxied

       e.g., https://ctf.example.com/orc/users/auth/teamforge/callback
    d) Save edits

Adapters: Integration using TeamForge Orchestrate

TeamForge Orchestrate integrates with different flavors of work item, SCM/commit, build, and review servers using "adapters".

TeamForge Orchestrate collects metadata about key lifecycle activities from external sources like work item trackers, SCM repositories, CI/build servers, and code review systems. To do so, TeamForge Orchestrate relies on "adapters" to collect the metadata from sources. TeamForge Orchestrate packages the following adapters:

Type of service	Product Description
SCM/Version Control	Git, Subversion
Builds/CI server	Jenkins, JetBrains TeamCity
Code Review	ReviewBoard, Atlassian Crucible
Work Item/Issue Tracker	CollabNet TeamForge, Atlassian JIRA

Download adapters from the *TeamForge Orchestrate Community*.You can also create adapters as documented in *Extending TeamForge Orchestrate* on page 663.

Source association keys

TeamForge Orchestrate uses source association keys to uniquely identify incoming data from activity sources that it monitors.

Adapters supply data to TeamForge Orchestrate through the message queue (MQ) layer. When a relevant activity occurs in source systems, TeamForge Orchestrate adapters generate a message with key metadata about the activity and place that message on the designated MQ server. When TeamForge Orchestrate receives the message, it must associate it to the proper source somehow. To do this, TeamForge Orchestrate supplies a unique "source association key" for each external source created. Adapters must supply this unique source association key with each message for the data to correlate with the appropriate project source.

Note that commit sources based on TeamForge project repositories do not need source associations keys since they are internal and are identified by other means. All other sources require source association keys.

Locate a source association key
Source association key for existing activity sources are located on the corresponding source configuration page.

1. From the activity stream, click **Manage Sources**.

2. Select the desired source type toggle.
   TeamForge Orchestrate displays all existing sources.

3. Click **Edit** on the source whose key you wish to obtain.
   You see the configuration page for the selected source.

4. Locate the source association key for the selected source. On most browsers you can copy the key by clicking the small clipboard icon.

JIRA adapter overview

The TeamForge Orchestrate JIRA integration brings associations and traceability to JIRA.

The TeamForge Orchestrate JIRA adapter enables JIRA as an alternative tracker for use with TeamForge by detecting and storing associations between JIRA issues and TeamForge managed SCM activities, like commits. Traceability is therefore possible between JIRA, TeamForge, and various orchestrated third-party tools.

Inside TeamForge, JIRA issues are visible in activity streams and association visualizations. Within JIRA, all the SCM related activities for the particular JIRA issue are tracked and displayed under the **TeamForge** tab on the JIRA issue page.

The **TeamForge** tab provides a summary of associations and also details a full listing of associations. This is a listing of immediate associations, activities with direct relationships to the JIRA issue at hand. To explore the chain of associations, click the **Trace** button. The traceability chain that includes commits, builds, reviews, deploys, and other XDS-based integration is displayed. On the **Trace Associations** page, clicking any node lights up all the association paths to that node. In addition, a small pop-up appears with details about the node in question. You can use **SET AS**

TARGET button to expand the associations from the selected point. Another function exists to **SEE** more details about the node, opening a new browser tab.

The TeamForge Orchestrate JIRA adapter is packaged as a JIRA "add-on". Once installed and configured, the add-on supplies issue tracker "work item" metadata to TeamForge Orchestrate. The JIRA adapter needs configuration on a per project basis in JIRA, such that JIRA (many) to TeamForge (one) project mappings are established. It is therefore required to configure a TeamForge project URL and a set of TeamForge credentials for each JIRA project. Note that once configured with the aforementioned URL and credentials, the JIRA adapter will create and manage the needed sources and source associations keys. For more about source association keys, see *Source association keys* on page 637.

Install a JIRA adapter
Install the JIRA adapter using the JIRA add-on Manager.

Use the JIRA adapter to notify Orchestrate of updates to JIRA issues and to visualize the associations between JIRA and other tools. The JIRA adapter must be installed once on each JIRA server you wish to connect to TeamForge Orchestrate. Install the adapter using the JIRA add-on Manager. Refer to the *Atlassian Marketplace* for supported versions.

Installing the JIRA add-on

- As a privileged JIRA user, navigate to **Administration** > **Add-ons** > **Find new add-ons**
- Search the marketplace for the **TeamForge Associations** add-on for JIRA. Type 'TeamForge Associations' and search.

- Click **Install**.

Upgrading the add-on

- New versions of the TeamForge Associations JIRA add-on will be visible in the **Manage add-ons** section (**Administration** > **Add-ons** > **Manage Add-ons**).

Configure a JIRA adapter

Configure "TeamForge Associations" to notify TeamForge EventQ about work item activities.

Before you start configuration, make sure that you have installed the TeamForge Associations add-on: verify that **TeamForge Associations** is present in the **User-installed Add-ons** list.

1. Configure the add-on from the **Project Administration** page to notify TeamForge-EventQ about the work item activities.

   a) Navigate to **Administration** > **TeamForge**.

   The **TeamForge Setup** page appears.

   b) Find the TeamForge project to associate with and enter its url in the **Project Home URL** field.

   Obtain the TeamForge Project Home URL by navigating to the target TeamForge project and clicking **Project Home**. Copy the URL from the browser.

   c) Using your valid TeamForge login credentials, click **Test Configuration**.

   The user whose credentials are supplied must have API permissions in both TeamForge and EventQ, or is a Project Admin.

d) Click **Save** to save the configuration.

When you save your configuration, for each issue type in the JIRA project a corresponding source is created in the TeamForge EventQ.

Sync Existing JIRA issues into TeamForge

2. To associate TeamForge objects to existing JIRA issues, click **Sync Issues**.

This step bootstraps JIRA issue data from the current project into the TeamForge EventQ data store. Note that this process may take several minutes (even hours) to complete, depending on the number of issues in the JIRA project.

Test or modify configuration

3. Test the current configuration by clicking **Test Configuration**.

The system indicates whether the supplied URL and credentials work properly.

4. To modify the supplied credentials, click **Edit Credentials**.

   a) Modify either the **Username** or **Password** fields.
   b) Click **Test Configuration** to verify your settings.
   c) Click **Save** once testing is satisfactory.

5. To modify the TeamForge project mapping, click **Modify Project Mapping**.

   a) Modify the **ProjectHome URL** field to reflect the desired TeamForge project.
   b) If required, update the credentials to match the TeamForge project.
   c) Click **Test Configuration** to verify that the supplied user has adequate permissions in the selected TeamForge project.
   d) Click **Save Mapping** once testing is satisfactory.

   ☞ **Note:** Upon successfully re-mapping the JIRA project, Run **Sync Issues** again to populate the TeamForge data store.

Delete Project Mapping Configuration

6. Click **Delete** if you wish to completely disassociate the JIRA project from the configured TeamForge project.

   ⚠ **Warning:** Note that doing so will abandon all existing association data. The JIRA project can be mapped to another project subsequently, but existing associations will be lost. Do this if you were testing the integration using a production JIRA project but now wish to remove any test association data.

Associate JIRA Issues to Version Control Commits

Associations between JIRA® issues and TeamForge-aware version control commits can be created via commit message references. Commits to TeamForge project repositories and external repositories that have been configured as EventQ sources are also supported.

Creating associations

1. Using your desired terminal or IDE, instantiate a version control commit to your repository of choice.

2. In the commit message, make reference to the JIRA ID(s) to which you wish to associate the commit surrounded by square brackets.

   Sample commit messages

   ```
   "[DEMO-123] This commit message will result in an association between
    JIRA issue DEMO-123 and this commit."

   "[DEMO-123, DEMO-124] Here I'm associating two JIRA issues with project
    identifiers DEMO."
   ```

Viewing Associations inside JIRA

3. Navigate to the desired JIRA issue details page.

4. Click the **TeamForge** tab.

5. A list of immediate associations appears. In other words, these objects are directly associated to the JIRA issue in question.

6. Click **Trace** to view the first three levels of the traceability chain.

☞ **Note:**

- All activities are mapped chronologically.
- Lines indicate direct associations.
- Use the "+" icon to explore further levels of traceability.

Troubleshooting

My associations do not appear. What should I do?	Verify the following:
	• Only repositories scoped to the configured TeamForge project will result in associations. That is, make sure the version control repository you're using is a part of the TeamForge project that has been mapped to the JIRA project in question.
	• When a TeamForge project is first mapped to a JIRA project, the JIRA issues need to be synced into the TeamForge data store. Association will fail if you attempt to create an association between a commit and JIRA issue that has not yet been synced. Please refer to the Sync Issues functionality.
	• Make sure the association syntax is correctly using square brackets. Association syntax is case sensitive.

Troubleshooting the JIRA adapter
Solutions to common issues with the JIRA adapter:

I just installed the Orchestrate JIRA adapter and made a commit against a JIRA ticket but the association doesn't appear.	Orchestrate is an event system that associates activities across systems. If the JIRA issue you made a commit against hasn't been modified since installing the Orchestrate JIRA adapter, Orchestrate is unaware of the given issue and therefore won't process an association to it. The solution is to make a nominal change to the JIRA issue. This will automatically register the issue with Orchestrate and associations will start happening after that.
Associations were working fine but now stopped working or appearing in JIRA.	First, test the connection from the JIRA project to the TeamForge project. In JIRA: **Project administration** > **TeamForge** > **Test Configuration**
	If you receive an error, make sure that the credentials used to map the JIRA project to the TeamForge project are still valid. TeamForge credentials can expire, user can be disabled or removed, which would invalidate the integration. The solution is to update the credentials in JIRA and re-test the configuration.
"Test Configuration" fails with authentication errors	Try the following troubleshooting steps:
	• Check the username and password supplied. Note that this is a TeamForge username and password.
	• Check that the Project Home URL is correct and is the intended Project Home URL.

	• Check that the user supplied has proper permissions for the TeamForge project in question. The user whose credentials are supplied must have API permissions in both TeamForge and Orchestrate, or is a project admin.
I just re-mapped a JIRA project to a different TeamForge project and made a commit against a JIRA ticket but the association doesn't appear.	Orchestrate is an event system that associates activities across systems. If the JIRA issue you made a commit against hasn't been modified since re-mapping the JIRA project to another TeamForge project, Orchestrate is unaware of the given issue and therefore won't process an association to it. The solution is to make a nominal change to the JIRA issue. This will automatically register the issue with Orchestrate and associations will start happening after that.
I get an error trying to re-map a JIRA project to a different TeamForge project	Check that the user supplied has proper permissions in both the original and re-mapped TeamForge projects. It is not sufficient for the user to have permission in just one of the two TeamForge projects in question as the re-mapping process touches endpoints in both projects.

Jenkins adapter overview

The TeamForge Orchestrate Jenkins adapter supplies build data from your Jenkins continuous integration server to TeamForge Orchestrate.

The TeamForge Orchestrate Jenkins adapter is packaged as a Jenkins plug-in. Once installed and configured, the plug-in supplies build results metadata for configured Jenkins jobs to TeamForge Orchestrate. The build results metadata is associated to the proper source using source association keys (see *Source association keys* on page 637). Configuration therefore requires that the MQ server URL and credentials be supplied, along with the appropriate source association key.

Install a Jenkins adapter

Install the Jenkins adapter using the web interface.

Use the Jenkins adapter to notify TeamForge Orchestrate when builds complete. The Jenkins adapter must be installed once on each Jenkins server you wish to connect to TeamForge Orchestrate. Install the plug-in using the web interface.

TeamForge Orchestrate currently supports Jenkins version 1.621.

- Use the web interface to install the plug-in:
  a) Download the `Orchestrate-Jenkins-adapter-[version].hpi` Jenkins plug-in file from the *TeamForge Orchestrate Community* web site.
  b) As a privileged Jenkins user, navigate to **Manage Jenkins** > **Manage Plugins** > **Advanced**
  c) Under the "Upload Plugin" section, click **Choose File**, and select your `Orchestrate-Jenkins-adapter-[version].hpi` file. Then click **Upload**
  d) Restart your Jenkins server.

Configure a Jenkins adapter

Configure the Jenkins adapter to notify TeamForge Orchestrate when builds complete.

Before you start configuration, make sure that you have installed the Jenkins plug-in using the web or console interface. The TeamForge Orchestrate Jenkins adapter has a system configuration as well as job level configurations. The system configuration sets defaults to be used by subsequently configured job level configurations.

> ☞ **Important:** When a job configuration is copied in Jenkins, the source association key is not carried over. You must add a corresponding build source in TeamForge Orchestrate, and enter the source association key while configuring the new job.

Configuring an individual Jenkins job

1. As a privileged Orchestrate user, edit (or add) the relevant Build source you wish to associate with this Jenkins job.

   Manage Sources > **Toggle Build** > **Edit the Source**

2. On the Edit Source screen, locate the section titled "Adapter Configuration Information" and copy all values; these values are used later in the configuration process.

3. As a privileged Jenkins user, locate the job that must report build data to TeamForge Orchestrate and navigate to its configuration page.

4. Add a post-build action to **Notify TeamForge Orchestrate when a build completes**.

5. Populate the Server URL field.

   Copy and paste the Queue Server value from the Edit Source screen in TeamForge Orchestrate. Please ensure to include the "ampq" protocol and the port information.

6. Populate the Server Username and Server Password fields.

   Copy and paste the queue username and password value from the Edit Source screen in TeamForge Orchestrate.

7. Populate the Build source association key field.

   Copy and paste the source association key value from the Edit Source screen in TeamForge Orchestrate. For more on association keys, see *Source association keys* on page 637.

8. Save the job configuration.

9. Run a build to test the new configuration and verify configuration.

   Information and errors will be reported to your Jenkins log and to the build console.

Configuring global default Jenkins settings

10. Obtain build adapter configuration from any existing build source.

    Manage Sources > **Toggle Build** > **Edit the Source**

11. On the Edit Source screen, locate the section titled "Adapter Configuration Information" and copy all values; these values are used later in the configuration process.

12. As a privileged Jenkins user, navigate to **Manage Jenkins** > **Configure System** screen on the target Jenkins server.

13. On the Jenkins configuration page, find the TeamForge Orchestrate section.

    a) Populate the Server URL field.

       Copy and paste the Queue Server value from the Edit Source screen in TeamForge Orchestrate. Please ensure to include the "ampq" protocol and the port information.

    b) Populate the Server Username and Server Password fields.

       Copy and paste the queue username and password value from the Edit Source screen in TeamForge Orchestrate.

14. Save the system configuration.

Troubleshooting the Jenkins adapter
You may encounter these problems when using the Jenkins adapter:

How do I get the queue server hostname and credentials? I need it to set up my adapter.	Adapters use a message queue server to transmit data to TeamForge Orchestrate. Adapters must therefore be configured with the queue server hostname, username, and password.
	When you create a new "Source" in TeamForge Orchestrate, a unique set of queue server credentials is created for use with your adapters. To find the queue server hostname and credentials, log into TeamForge as a privileged Orchestrate user and navigate to: **Manage Sources** > **Toggle Build** > **Edit the Source**
	Locate the section named "Adapter Configuration Information".

	Copy the queue server's hostname, username, and password and use them to configure your adapter.
	Note that the same queue server hostname, username and password may be shared by multiple sources; however, the source association must be unique for each source.
My build is marked "Unstable"	Builds are marked "Unstable" if the Jenkins system configuration has not been saved and the Jenkins adapter cannot successfully communicate with TeamForge Orchestrate or the message queue. The console output for the build will include a message like this:
	`Build information NOT sent: plug-in needs a Jenkins URL`
	Workaround: As a privileged Jenkins user, navigate to **Manage Jenkins** > **Configure System** and populate and save the required configuration.
Automated test results for my Jenkins builds are not showing in TeamForge Orchestrate	TeamForge Orchestrate relies on Jenkins to supply automated test results. Jenkins has a post-build action called "Publish JUnit test result report" which must be activated for test results to show up in TeamForge Orchestrate, regardless of whether JUnit is actually used.
	Workaround: To enable this configuration, add the **Publish JUnit test result report** post-build action to the desired Jenkins job configuration.

TeamCity adapter overview

The TeamForge Orchestrate TeamCity adapter supplies build data from your TeamCity continuous integration server to TeamForge Orchestrate.

The TeamForge Orchestrate TeamCity adapter is packaged as a TeamCity plug-in. Once installed and configured, the plug-in supplies build results metadata for selected TeamCity build configurations to TeamForge Orchestrate. The build results metadata is associated to the proper source using source association keys (see *Source association keys* on page 637). Configuration therefore requires that the MQ server URL and credentials be supplied, along with the appropriate source association key.

Install a TeamCity adapter

Install the TeamCity adapter as a TeamCity plug-in

TeamForge Orchestrate currently supports Team City — 8.0.0.

The TeamCity adapter must be installed once on each TeamCity server you wish to connect to TeamForge Orchestrate. The TeamForge Orchestrate TeamCity adapter installs as a TeamCity plug-in. Installation of the adapter requires command line access to the TeamCity server.

1. Shutdown the TeamCity server. You must shut down the TeamCity server prior to installing the adapter. For more information on how to shut down and restart the TeamCity server, see the JetBrains TeamCity documentation.
2. Download the `Orchestrate-TeamCity-adapter-[version].zip` adapter file from the *TeamForge Orchestrate Community* web site.
3. Copy the TeamCity adapter zip archive to: `[TeamCity Data Directory]/plugins`

   Do not unzip the file. The TeamCity Data Directory location can be set using the `TEAMCITY_DATA_PATH` environment variable. The default data directory is `$HOME/.BuildServer` on Linux and `C:\Users \<user_name>\.BuildServer` on Windows.

   See *TeamCity plug-in Information* for more information on TeamCity plug-ins.
4. Start the TeamCity server and then configure the adapter.

Configure a TeamCity adapter

Configure the TeamCity adapter to notify TeamForge Orchestrate when builds complete.

Before you start configuration, make sure that you have installed the TeamCity adapter. The TeamForge Orchestrate TeamCity adapter has a system configuration as well as build configuration level settings.

☞ **Important:** When a build configuration is copied in TeamCity, the source association key is not carried over. You must add a corresponding build source in TeamForge Orchestrate, and enter the source association key while configuring the new build configuration.

1. As a privileged Orchestrate user, edit (or add) the relevant Build source you wish to associate with this TeamCity build configuration.

   Manage Sources > **Toggle Build** > **Edit the Source**

2. On the Edit Source screen, locate the section titled "Adapter Configuration Information" and copy all values; these values are used later in the configuration process.

3. As a privileged TeamCity user, navigate to **Administration** > **TeamForge Orchestrate**.

4. Populate the Server URL field.

   Copy and paste the Queue Server value from the Edit Source screen in TeamForge Orchestrate. Please ensure to include the "ampq" protocol and the port information.

5. Populate the Server Username and Server Password fields.

   Copy and paste the queue username and password value from the Edit Source screen in TeamForge Orchestrate.

6. Save the system configuration.

   Note that steps related to system configuration need only be done once per TeamCity server.

7. Click on the Job/Build you wish to attach to TeamForge Orchestrate.

8. Click on **Orchestrate** tab.

9. Populate the Build source association key field.

   Copy and paste the source association key value from the Edit Source screen in TeamForge Orchestrate. For more on association keys, see *Source association keys* on page 637.

10. Save the build configuration.

11. Run a build to test the new configuration and verify configuration.

Troubleshooting the TeamCity plug-in

You may encounter these problems when using the TeamCity adapter:

How do I get the queue server hostname and credentials? I need it to set up my adapter.

Adapters use a message queue server to transmit data to TeamForge Orchestrate. Adapters must therefore be configured with the queue server hostname, username, and password.

When you create a new "Source" in TeamForge Orchestrate, a unique set of queue server credentials is created for use with your adapters. To find the queue server hostname and credentials, log into TeamForge as a privileged Orchestrate user and navigate to: **Manage Sources** > **Toggle Build** > **Edit the Source**

Locate the section named "Adapter Configuration Information".

Copy the queue server's hostname, username, and password and use them to configure your adapter.

Note that the same queue server hostname, username and password may be shared by multiple sources; however, the source association must be unique for each source.

External Git repository integration overview

With a customized post-receive extension (hook) installed, Git servers external to TeamForge can notify TeamForge Orchestrate of Git "push" activity.

TeamForge project Git repositories are already wired to supply push activity data to TeamForge Orchestrate; no special configuration needed. This section instead details integration of Git repositories external to TeamForge.

The TeamForge Orchestrate Git adapter is packaged as a Git post-receive hook script. Once installed and configured, the hook script supplies push metadata for the selected repository to TeamForge Orchestrate. The push data is associated to the proper source using source association keys (see *Source association keys* on page 637). Configuration therefore requires that the message queue server URL and credentials be supplied, along with the appropriate source association key.

Install a Git adapter

Install a Git adapter as a post-receive hook on Mac OS X/Linux.

Minimum requirements:

- Git 1.8.3.1 or later.
- python 2.6. Version 3.0 and later are not compatible.

Each external Git repository you wish to monitor with TeamForge Orchestrate requires the installation of a customized post-receive hook script as follows.

Install Git post-receive hooks

1. Download the `Orchestrate-git-adapter-[version].zip` adapter file from the *TeamForge Orchestrate Community* web site and unpack it inside your repository's hooks directory. For example, if your repository's hooks directory is located at `/path/to/repo.git/hooks`:

   ```
   cd /path/to/repo.git/hooks
   ```

   Download the zip file to the current location, then:

   ```
   unzip Orchestrate-git-adapter-1.0.zip
   ```

   This will unpack the following into your hooks directory:

   - Files: `LICENSE.txt README.txt post_receive_orc_publish_amqp.conf.sample post_receive_orc_publish_amqp.py`
   - Directories: `orchestrate/ pika/`

Configuration

2. In a web browser, log into TeamForge Orchestrate as a privileged Orchestrate user and edit (or add) the relevant Commit source you wish to associate with this Git repository.

   Manage Sources > **Toggle Commit** > **Add or Edit a Source** > **External repository**

3. On the Edit Source screen, locate the section titled "Adapter Configuration Information" and copy all values; these values are used later in the configuration process.

4. Switch back to your command line console and copy `post_receive_orc_publish_amqp.conf.sample` to `post_receive_orc_publish_amqp.conf`.

5. Edit `post_receive_orc_publish_amqp.conf` and modify options for your specific installation, for example:

   Logging section
   - filename — Full path to the desired log file, for example, `/tmp/post-commit.log`.
   - level — The logging level, which is one of these: `debug`, `info`, `warning`, `error`, or `critical`.

   RabbitMQ (`amqp`) section
   - host — The hostname of the `amqp` server. Copy and paste the hostname from the Queue Server

value from the Edit Source screen in TeamForge Orchestrate.

- port — The port number of the `amqp` server (default is 5672). Copy and paste the port number the Queue Server value from the Edit Source screen in TeamForge Orchestrate.
- username— Username used to access the `amqp` server. Copy and paste the queue username value from the Edit Source screen in TeamForge Orchestrate.
- password — Password used to access the `amqp` server. Copy and paste the queue password value from the Edit Source screen in TeamForge Orchestrate.

TeamForge Orchestrate section
- source_association_key — key provided by TeamForge Orchestrate. Copy and paste the source association key value from the Edit Source screen in TeamForge Orchestrate. For more on association keys, see *Source association keys* on page 637.

Activate post-receive hook

6. Copy or create a `symlink` called "post-receive" linking to `post_receive_orc_publish_amqp.py`.

External Subversion repository integration overview
With a customized post-commit extension (hook) installed, Subversion servers external to TeamForge can notify TeamForge Orchestrate of Subversion commit activity.

TeamForge project Subversion repositories are already wired to supply commit activity data to TeamForge Orchestrate; no special configuration needed. This section instead details integration of Subversion repositories external to TeamForge.

The TeamForge Orchestrate Subversion adapter is packaged as a Subversion post-commit hook script. Once installed and configured, the hook script supplies commit metadata for the selected repository to TeamForge Orchestrate. The commit data is associated to the proper source using source association keys (see *Source association keys* on page 637). Configuration therefore requires that the message queue server URL and credentials be supplied, along with the appropriate source association key.

Install a Subversion adapter on Mac OS X/Linux
Install a Subversion adapter as a post-receive hook on Mac OS X/Linux.

Minimum requirements:

- Subversion 1.7 or later.
- python 2.6. Version 3.0 and later are not compatible.

Each external Subversion repository you wish to monitor with TeamForge Orchestrates requires the installation of a customized post-commit hook script as follows. CollabNet Subversion Edge 4.0 or later is also supported.

Install Subversion post-commit hooks

1. Download the `Orchestrate-svn-adapter-[version].zip` adapter file from the *TeamForge Orchestrate Community* web site and unpack it inside your repository's hooks directory. For example, if your repository's hooks directory is located at `/path/to/svn/repos/my_repo/hooks` :

`cd /path/to/svn/repos/my_repo/hooks`

Download the zip file to the current location

`unzip Orchestrate-svn-adapter-1.0.zip`

This will unpack the following into your hooks directory:

- `LICENSE.txt`
- `README.txt`
- `configdefaults.py`
- `configwrapper.py`
- `orchestrate_commit_gate_hook.py`
- `orchestrator_commit_gate_hook_snippet.sh`
- `pika/`
- `post-commit.bat.sample`
- `post_commit_orc_publish_amqp.conf.sample`
- `post_commit_orc_publish_amqp.py`

Configuration

2. In a web browser, log into TeamForge Orchestrate as a privileged Orchestrate user and edit (or add) the relevant Commit source you wish to associate with this Subversion repository.

   Manage Sources > **Toggle Commit** > **Add or Edit a Source** > **External repository**

3. On the Edit Source screen, locate the section titled "Adapter Configuration Information" and copy all values; these values are used later in the configuration process.

4. Switch back to your command line console and copy `post_commit_orc_publish_amqp.conf.sample` to `post_commit_orc_publish_amqp.conf`.

5. Edit `post_commit_orc_publish_amqp.conf` and modify options for your specific installation, for example:

Logging section	• filename — Full path to the desired log file, for example, `/tmp/post-commit.log`. • level — The logging level, which is one of these: `debug`, `info`, `warning`, `error`, or `critical`.
svnlook section	• svnlook — The path to your svnlook executable, for example, `/usr/bin/svnlook` or `/opt/subversion/bin/svnlook`.
RabbitMQ (amqp) section	• host — The hostname of the `amqp` server. Copy and paste the hostname from the Queue Server value from the Edit Source screen in TeamForge Orchestrate. • port — The port number of the `amqp` server (default is 5672). Copy and paste the port number the Queue Server value from the Edit Source screen in TeamForge Orchestrate. • username— Username used to access the `amqp` server. Copy and paste the queue username value from the Edit Source screen in TeamForge Orchestrate. • password — Password used to access the `amqp` server. Copy and paste the queue password value from the Edit Source screen in TeamForge Orchestrate.
TeamForge Orchestrate section	• source_association_key — the key provided by TeamForge Orchestrate. Copy and paste the source association key value from the Edit Source screen

in TeamForge Orchestrate. For more on association keys, see *Source association keys* on page 637.

Activate post-commit hook

6. Copy or create a `symlink` called "post-commit" linking to `post_commit_orc_publish_amqp.py`.

Install a Subversion adapter on Windows

Install a Subversion adapter as a post-receive hook on Windows.

Minimum requirements:

- Subversion 1.7 or later.
- python 2.6. Python version 3.0 and later are not compatible. Note, python is bundled with Subversion Edge.

This procedure assumes that you have installed CollabNet Subversion Edge, version 4.0 or higher, although other Windows distributions can be similarly configured. Each external Subversion repository you wish to monitor with TeamForge Orchestrates requires the installation of a customized post-commit hook script as follows.

Install Subversion post-commit hooks

1. Download the `Orchestrate-svn-adapter-[version].zip` adapter file from the *TeamForge Orchestrate Community* web site and unpack it inside your repository's hooks directory. For example, if your repository's hooks directory is located at `C:\csvn\repositories\my_repo\hooks`:

    cd C:\csvn\repositories\my_repo\hooks

    # Download the zip file to the current location

    unzip Orchestrate-svn-adapter-1.0.zip

This will unpack the following into your hooks directory:

- `LICENSE.txt`
- `README.txt`
- `configdefaults.py`
- `configwrapper.py`
- `orchestrate_commit_gate_hook.py`
- `orchestrator_commit_gate_hook_snippet.sh`
- `pika/`
- `post-commit.bat.sample`
- `post_commit_orc_publish_amqp.conf.sample`
- `post_commit_orc_publish_amqp.py`

Configuration

2. In a web browser, log into TeamForge Orchestrate as a privileged Orchestrate user and edit (or add) the relevant Commit source you wish to associate with this Subversion repository.

Manage Sources > **Toggle Commit** > **Add or Edit a Source** > **External repository**

3. On the Edit Source screen, locate the section titled "Adapter Configuration Information" and copy all values; these values are used later in the configuration process.

4. Copy `post_commit_orc_publish_amqp.conf.sample` to `post_commit_orc_publish_amqp.conf`.

5. Edit `post_commit_orc_publish_amqp.conf` and modify options for your specific installation, for example:

Logging section

- filename — Full path to the desired log file, for example, `C:\Temp\post-commit.log`.

  ☞ **Note:** Ensure that the log file exists, or create it, before proceeding to the next step.

- level — The logging level, which is one of these: `debug` , `info`, `warning`, `error`, or `critical`.

svnlook section

- svnlook — The path to your svnlook executable, for example, `C:\csvn\bin\svnlook.exe`.

RabbitMQ (`amqp`) section

- host — The hostname of the `amqp` server. Copy and paste the hostname from the Queue Server value from the Edit Source screen in TeamForge Orchestrate.
- port — The port number of the `amqp` server (default is 5672). Copy and paste the port number the Queue Server value from the Edit Source screen in TeamForge Orchestrate.
- username— Username used to access the `amqp` server. Copy and paste the queue username value from the Edit Source screen in TeamForge Orchestrate.
- password — Password used to access the `amqp` server. Copy and paste the queue password value from the Edit Source screen in TeamForge Orchestrate.

A RabbitMQ server is optionally set up during the installation process. The installation process sets a default password for the "orchestrate" user and stores it for reference in the following file on the RabbitMQ host (root access required):

```
/opt/collabnet/
rabbitmq/.rabbit_passwords
```

TeamForge Orchestrate section

- source_association_key — key provided by TeamForge Orchestrate. Copy and paste the source association key value from the Edit Source screen in TeamForge Orchestrate. For more on association keys, see *Source association keys* on page 637.

Activate post-commit hook

6. Copy `post-commit.bat.sample` to `post-commit.bat`.
7. Edit `post-commit.bat` for your specific installation: set `PYTHON_EXE` to the full path to `python.exe` executable. For CollabNet Subversion Edge, this is typically `C:\csvn\Python25\python.exe`

Troubleshooting the Subversion adapter
You may encounter these problems when using the Subversion adapter:

How do I get the queue server hostname and credentials? I need it to set up my adapter.

Adapters use a message queue server to transmit data to TeamForge Orchestrate. Adapters must therefore be configured with the queue server hostname, username, and password.

When you create a new "Source" in TeamForge Orchestrate, a unique set of queue server credentials is created for use with your adapters. To find the queue server hostname and credentials, log into TeamForge as a privileged Orchestrate user and navigate to: **Orchestrate**

(from the project navigation bar) > **Manage Sources**. The **Sources Overview** page appears.

Select one of the adapters on the left hand side. The 'Source List' shows up on the right hand side.

From the list of sources, click the **EDIT** button of the desired source.

Locate the section named "Adapter Configuration Information".

Copy the queue server's hostname, username, and password and use them to configure your adapter.

Note that the same queue server hostname, username and password may be shared by multiple sources; however, the source association must be unique for each source.

Commits fail after installing the adapter hook script

After installing the hook script, I see an error complaining about svnlook:

```
Warning: post-commit hook failed
 (exit code 1) with output:
svnlook: Expected FS format between
 '1' and '4'; found format '6'
svnlook: Expected FS format between
 '1' and '4'; found format '6'
Traceback (most recent call last):
File "/home/user/SVN_Edge/
csvn-4.0.0/data/repositories/
svn_edge/hooks/post-commit", line
 488, in <module>
raise Exception("post-commit failed:
 %s" % e)
Exception: post-commit failed: this
 constructor takes no arguments"
```

Workaround: This problem is caused by having different versions of svnlook installed: typically one in /usr/bin and another bundled with Subversion Edge. To remedy, edit the Orchestrate Subversion Adapter configuration file `post_commit_orc-publish_amqp.conf` and modify the "svnlook" value to point to the binary packaged with Subversion Edge (example: `/home/user/SVN_Edge/current/bin`).

Crucible adapter overview

The TeamForge Orchestrate Crucible adapter enables communication between your Crucible server and TeamForge Orchestrate.

The TeamForge Orchestrate Crucible adapter is packaged as a Crucible add-on. Once installed and configured, the add-on supplies code review metadata for selected Crucible projects to TeamForge Orchestrate. The code review metadata is associated to the proper source using source association keys (see *Source association keys* on page 637). Configuration therefore requires that the MQ server URL and credentials be supplied, along with the appropriate source association key.

Install a Crucible adapter

Install the Crucible adapter using the Crucible add-on Manager.

TeamForge Orchestrate currently supports Crucible 3.0.1. Use the Crucible adapter to notify TeamForge Orchestrate when reviews are created or modified, or when files are added or removed from reviews. The Crucible adapter must

be installed once on each Crucible server you wish to connect to TeamForge Orchestrate. Install the add-on by using the Crucible add-on Manager.

- Download the `Orchestrate-Crucible-adapter-[version].jar` Crucible plug-in file from the *TeamForge Orchestrate Community* web site.
- As a privileged Crucible user, navigate to **Administration** > **Manage Add-ons** > **Upload Add-on**
- In the **From my computer** field, select your `Orchestrate-Crucible-adapter-[version].jar` file, then click **Upload**.
  You are now returned to the "Manage Add-ons" view and the "TeamForge Orchestrate Adapter" now be visible under "User-installed Add-ons".

> ☞ **Note:** In the future you can upload new copies of the adapter and they will automatically upgrade the old adapter while retaining your old settings.

Configure a Crucible add-on

Configure the "TeamForge Orchestrate Adapter" add-on to notify TeamForge Orchestrate about review changes.

Before you start configuration, make sure that you have installed the Crucible add-on: verify that **TeamForge Orchestrate Adapter** is present in the **User-installed Add-ons** list.

1. As a privileged Orchestrate user, edit (or add) the relevant Review source you wish to associate with this Crucible project.

   Manage Sources > **Toggle Review** > **Edit the Source**

2. On the Edit Source screen, locate the section titled "Adapter Configuration Information" and copy all values; these values are used later in the configuration process.

3. As a privileged Crucible user, navigate to **Administration** > **Manage Add-ons**

4. Click **TeamForge Orchestrate Adapter** in the **User-installed Add-ons** list.
   The TeamForge Orchestrate Adapter section displays new options, including "Configure".

5. Click **Configure**.
   You see a configuration form for each of your projects.

6. Find the project to associate with TeamForge Orchestrate and populate the following information.

   a) Populate the Server URL field.

   Copy and paste the Queue Server value from the Edit Source screen in TeamForge Orchestrate. Please ensure to include the "ampq" protocol and the port information.

   b) Populate the Server Username and Server Password fields.

   Copy and paste the queue username and password value from the Edit Source screen in TeamForge Orchestrate.

   c) Populate the Source Association Key field.

   Copy and paste the source association key value from the Edit Source screen in TeamForge Orchestrate. For more on association keys, see *Source association keys* on page 637.

7. Click **Save**.
   Crucible now reports code review events for the configured project to TeamForge Orchestrate.

Troubleshooting the Crucible adapter

You may encounter these problems when using the Crucible adapter:

How do I get the queue server hostname and credentials? I need it to set up my adapter.	Adapters use a message queue server to transmit data to TeamForge Orchestrate. Adapters must therefore be configured with the queue server hostname, username, and password.
	When you create a new "Source" in TeamForge Orchestrate, a unique set of queue server credentials is created for use with your adapters. To find the queue server hostname and credentials, log into TeamForge as

a privileged Orchestrate user and navigate to: **Manage Sources** > **Toggle Review** > **Edit the Source**

Locate the section named "Adapter Configuration Information".

Copy the queue server's hostname, username, and password and use them to configure your adapter.

Note that the same queue server hostname, username and password may be shared by multiple sources; however, the source association must be unique for each source.

Where are communication errors reported?

Communication problems with the TeamForge Orchestrate message queue server — such as inability to access the server URL or incorrect authentication credentials — will be reported in the Fisheye-Crucible error log, usually found in `[FISHEYE_INST]/var/log/atlassian-fisheye-YYYY-MM-DD.log`

Review Board adapter overview

The TeamForge Orchestrate Review Board adapter enables communication between your Review Board server and TeamForge Orchestrate.

The Review Board adapter is packaged as a Review Board extension, and once installed provides web-based configuration inside Review Board. TeamForge Orchestrate supports Review Board versions 1.7 and higher.

The adapter supplies code review metadata for selected repositories to TeamForge Orchestrate. Code review metadata is associated to the proper source using source association keys (see *Source association keys* on page 637). Configuration therefore requires that the MQ server URL and credentials be supplied, along with the appropriate source association key.

Install a Review Board adapter on Mac OS X/Linux

Install a Review Board adapter as a Review Board extension to retrieve code review data from instances of Review Board 1.7 and higher.

Minimum requirements:

- Review Board 1.7 or higher
- python 2.6 or 2.7. Version 3.0 and later are not compatible.
- setuptools 2.1 or higher
- pip 1.5 - 6.0.6 (6.0.7 has issues and is not supported)

Please refer to the installation steps below, then continue to configuration.

1. Upgrade pip

   Upgrade pip if you do not have the minimum required version.

   ```
   sudo pip install --upgrade pip
   ```

2. Upgrade setuptools

   Upgrade setuptools if you do not have the minimum required version.

   ```
   sudo pip install --upgrade setuptools
   ```

3. Install the Orchestrate adapter as a Review Board extension

   If your server has internet access, use the command below to install the Orchestrate adapter as an extension. If upgrading, skip to the next step.

   ```
   sudo pip install --use-wheel --no-index --find-links=http://
   orchestrate.collab.net/RHEL/6/x86_64/wheelhouse orchestrate_rb
   ```

   If your server cannot access the internet do not use the command above. Instead, please see *Install a Review Board adapter without internet access* on page 656

4. Upgrading the Review Board adapter

If upgrading (instead of installing), add the -U flag to upgrade to the latest version of the Review Board adapter

```
sudo pip install -U --use-wheel --no-index --find-links=http://
orchestrate.collab.net/RHEL/6/x86_64/wheelhouse orchestrate_rb
```

5. Restart Review Board's web server to register the newly installed Orchestrate adapter.

Proceed to *Configure Review Board adapter (TeamForge integrated)* on page 656.

Install a Review Board adapter without internet access
Install a Review Board adapter without internet access.

The following instructions pertain to situations where your Review Board server does not have outbound access to the internet. Follow the instructions in *Install a Review Board adapter on Mac OS X/Linux* on page 655 until you are directed here to continue.

1. Download the Orchestrate Review Board adapter adapter archive file from the *TeamForge Orchestrate Community* web site.

2. Extract the archive file on the target server.

```
unzip orchestrate_rb-[version].zip
```

3. Install the Orchestrate adapter as a Review Board extension

Use the command below to install the Orchestrate adapter as an extension. If upgrading the Review Board adapter, skip to the next step.

```
sudo pip install --use-wheel --no-index --find-links=PATH_TO_WHEELHOUSE
orchestrate_rb
```

where `PATH_TO_WHEELHOUSE` is the path to the directory where you unzipped the adapter file. Suppose you unzipped the adapter file to the `/tmp` directory. In such a case, the `PATH_TO_WHEELHOUSE` is `/tmp`.

Example: `sudo pip install --use-wheel --no-index --find-links=/home/myusername/wheelhouse/ orchestrate_rb`

4. Upgrading the Review Board adapter

If upgrading (instead of installing), add the -U flag to upgrade to the latest version of the Review Board adapter

```
sudo pip install -U --use-wheel --no-index --find-links=PATH_TO_WHEELHOUSE
orchestrate_rb
```

5. Continue with installation documentation

Please continue with *Install a Review Board adapter on Mac OS X/Linux* on page 655 picking up with step *5* on page 656.

Configure Review Board adapter (TeamForge integrated)
Configure a TeamForge integrated Review Board adapter to retrieve code review metadata.

Once the adapter has been installed successfully, configure Review Board by following these steps. The instructions below pertain to Review Board instances integrated with CollabNet TeamForge. See *Configure Review Board adapter (standalone)* on page 657 for configuration instructions when using a standalone version of Review Board.

Navigate to Review Board extension administration

1. Log into TeamForge with a site-admin user.

2. Click the **Admin** link in the header area.

3. Click the **Integrated Apps** tab

4. Click the radio button pertaining to Review Board

5. With the radio button selected, click **Administer**

You should now see Review Board administrative page.

6. Click **Extensions** on the menu bar to list your currently installed extensions.

orchestrate-rb should be listed.

7. Click **Enable** to activate the Orchestrate extension.

8. Click **Configure** on the orchestrate-rb extension.

   The orchestrate-rb configuration page will be displayed. From here you can select and configure the repository whose reviews you would like to push to an Orchestrate review source. To create a review source in Orchestrate see: *Manage review sources* on page 602.

Configure a Review Board respository

9. Select the repository name from the **Repository** dropdown menu.

10. Copy and paste the fields from the Orchestrate review source that you wish to associate to this repository

11. Click **Save**

   Repeat steps *9* on page 657 through *11* on page 657 for each Review Board repository you wish to monitor in TeamForge Orchestrate.

Configure Review Board adapter (standalone)

   Configure a standalone instance of Review Board adapter to retrieve code review metadata.

   Once the adapter has been installed successfully, configure Review Board by following these steps. The instructions below pertain to standalone Review Board instances. See *Configure Review Board adapter (TeamForge integrated)* on page 656 for configuration instructions when using an instance of Review Board integrated with CollabNet TeamForge.

Navigate to Review Board extension administration

1. Log into your Review Board instance as a privileged user.

2. Click the **Admin** link to reach the administrative page.

3. Click **Extensions** on the menu bar to list your currently installed extensions.

   orchestrate-rb should be listed.

4. Click **Enable** to activate the Orchestrate extension.

5. Click **Configure** on the orchestrate-rb extension.

   The orchestrate-rb configuration page will be displayed. From here you can select and configure the repository whose reviews you would like to push to an Orchestrate review source. To create a review source in Orchestrate see: *Manage review sources* on page 602.

Configure a Review Board respository

6. Select the repository name from the **Repository** dropdown menu.

7. Copy and paste the fields from the Orchestrate review source that you wish to associate to this repository

8. Click **Save**

   Repeat steps *6* on page 657 through *8* on page 657 for each Review Board repository you wish to monitor in TeamForge Orchestrate.

System Administration
Manage the TeamForge Orchestrate server-side components

System administration introduction

The scope of this guide is limited to key administrative features like starting/stopping services, log locations, etc. Administration and configuration of dependent services (e.g., nginx, MongoDB, RabbitMQ) is not covered; see respective product documentation. It is assumed that as TeamForge Orchestrate administrator you are familiar with basic Linux administration commands and terminology.

Background services

TeamForge Orchestrate installs and starts six background services on the App server. Each service is managed using init scripts, which are installed and started as part of the installation process. Init scripts are all located in /etc/init.d/ on the respective hosts. Once started, each service creates a pid file.

PID file	Host	Description
/var/run/collabnet-mongodb/mongodb.pid mongod	DB server	Database service
/var/run/collabnet-nginx/nginx.pid nginx	App server	Web and application server
/var/run/collabnet-rabbitmq/rabbitmq.pid rabbitmq-server	MQ server	AMQP message service
/var/run/collabnet-redis/redis.pid redis	App server	Internal job queue service
/var/run/orchestrate/ctf_job_runner.pid ctf-jobs	App server	Background communication with TeamForge
/var/run/orchestrate/scm_queue.pid queue-commit	App server	Processes message from commit queue
/var/run/orchestrate/build_queue.pid queue-build	App server	Processes message from build queue
/var/run/orchestrate/review_queue.pid queue-review	App server	Processes message from review queue

Starting and stopping services

Use the init scripts to start, stop, restart, or get status on a service individually.

status	Is the process currently running? Does it have a PID (even if it's not running - e.g. in the case of a crash)
start	Starts the process
stop	Stop the process
restart	Stop and then start the process

Example usage:

```
sudo service orchestrate-queue-commit restart
```

☞ **Tip:** While each service can be managed individually, the orchestrate init script will start and stop all App server services: /etc/init.d/orchestrate

Example usage:

```
sudo service orchestrate start
```

Service administration passwords

During the installation of TeamForge Orchestrate, if RabbitMQ and MongoDB were installed as part of the installation process then the administrative passwords have been written to disk for reference:

- DB server: /opt/collabnet/mongodb/.mongo_passwords
- MQ server: /opt/collabnet/rabbitmq/.rabbit_passwords

Log files

The following log files are relevant to TeamForge Orchestrate services and may be useful for debugging. Log rotation on all log files is enabled by default.

Orchestrate App server logs:	The server.log is the main log file for TeamForge Orchestrate, detailing the server's page serves and any errors that Orchestrate may encounter. It relates to the Phusion Passenger service, which is started with nginx.
	/var/log/orchestrate/
	• build_queue.log
	• ctf_job_runner.log
	• nginx-error.log
	• nginx.log
	• review_queue.log
	• scm_queue.log
	• server.log
	/var/log/collabnet-redis/
	• redis.log
MongoDB server log:	• /var/log/collabnet-mongodb/mongodb.log
RabbitMQ server log:	/var/log/collabnet-rabbitmq/
	• rabbit@localhost.log
	• rabbit@localhost-sasl.log
	• startup_err
	• startup_log

Backup and restore

See *Backup and Restore* on page 660

Monitoring TeamForge Orchestrate with New Relic

TeamForge Orchestrate may be configured for monitoring using New Relic, a third party service.

This article explains how to set up monitoring of TeamForge Orchestrate using a third party product, New Relic. Once set up, configure New Relic to alert you of outages, monitor application performance, observe errors, etc. It is assumed that you have a New Relic account and license key ready for use.

☞ **Note:** New Relic is not affiliated with CollabNet and may bill you for usage of their services.

1. Create a New Relic account and obtain your license key

   If you haven't done so already, create an account on the New Relic web site. Obtain the license key associated with your account.
2. Contact *CollabNet Support* and request the TeamForge Orchestrate New Relic monitoring script.
3. Copy the New Relic script onto the TeamForge Orchestrate application server host (App server)

```
cp newrelic.sh /tmp cd /tmp chmod 755 newrelic.sh
```

4. Execute the New Relic script with root privileges, passing your license key as an argument

```
sudo ./newrelic.sh LICENSE_KEY
```

where `LICENSE_KEY` is replaced with your New Relic provided license key

After a few moments, your New Relic account will reflect the TeamForge Orchestrate server. Follow New Relic's instructions for setting up monitoring, alerts or other desired functionality.

Backup and Restore

The TeamForge Orchestrate backup and restore strategy involves a snapshot process along with periodic database and RabbitMQ configuration backups for use during data restoration.

Backup and Restore Overview

The TeamForge backup and data restoration strategy involves two workflows:

1. **Snapshot of Orchestrate program files**. On single-server installations the snapshot process outputs a single compressed tar archive containing all TeamForge Orchestrate program files. For multi-box installations, a snapshot must be taken of each server. For detailed instructions and links to download snapshot tools, see the *Snapshot backup and restore* on page 660 documentation.

2. **Periodic backups**. Alone, snapshots may be used to restore the system to the point in time when the original snapshot was taken. Use the data backup process to periodically backup the database and RabbitMQ configuration to supplement your snapshots with more current data. Please note that data backups alone are insufficient to restore in case of catastrophic failure; at least one snapshot is required for full system restoration. For detailed instructions on data backups, see the *Data back up and restore* on page 661 documentation.

Snapshot backup and restore

Create snapshots of your TeamForge Orchestrate program files to restore system after a failure.

The goal of this topic is to help you take "snapshots" of your TeamForge Orchestrate installation. Snapshots make a copy of all TeamForge Orchestrate program and configuration files and may be used to restore your system in case of a failure. The snapshot process involves shutting down TeamForge Orchestrate services, as such, we recommend that a single snapshot of each Orchestrate host be taken after installation and subsequent upgrades, and periodic database backups be made to restore to a specific (more recent) data set. See the *Data back up and restore* on page 661 documentation to create periodic data backups once you've created baseline snapshots.

> ☞ **Note:** The TeamForge Orchestrate hostname cannot be changed when restoring from a snapshot. Please contact support if you need to change the hostname of your TeamForge Orchestrate service.

Creating snapshots

A snapshot must be taken for each host running TeamForge Orchestrate services. If all of your TeamForge Orchestrate services are on a single host, then only one snapshot is required. However, if you've spread services across multiple hosts, then a single snapshot is required per host. The following steps describe making a single snapshot, so you may need to repeat these steps if you have services spread across multiple hosts.

1. Obtain `orchestrate-snapshot-tools.tar.gz` from the App server, located in `/opt/collabnet/orchestrate/tools/`

2. Untar the `orchestrate-snapshot-tools.tar.gz` on the target host that you wish to snapshot.

3. Stop the TeamForge Orchestrate service(s) running on the target host. See *System Administration* on page 657.

4. Run the take_snapshot.sh script with the appropriate argument.

   The `take_snapshot.sh` script takes one of four arguments (mongodb, rabbitmq, orchestrate, all). Run the script with the appropriate argument. For instance:

   ```
   sudo ./take_snapshot.sh all
   ```
   (single host installation)

   ```
   sudo ./take_snapshot.sh mongodb
   ```
   (multi-host, database server only)

   ```
   sudo ./take_snapshot.sh rabbitmq
   ```
   (multi-host, message queue server only)

```
sudo ./take_snapshot.sh orchestrate
```
(multi-host, application server only)

A snapshot will be created and this file should be moved to a safe location (off host). The filename of a snapshot with options 'orchestrate' or 'all' will include the version number of TeamForge Orchestrate.

5. Restart the TeamForge Orchestrate service(s). See *System Administration* on page 657.

Restore from a snapshot

6. Download the disconnected-media package corresponding to your version of Orchestrate. To obtain a disconnected-media package, see *Installation without internet access* on page 625.

To restore from a snapshot, you must first download and install the disconnected media RPM package matching your snapshot version number. For instance, if you have a snapshot file labeled 1.0.2, you should download the disconnected media package version 1.0.2.

7. Install the disconnected-media RPM:

```
sudo rpm -ivh collabnet-orchestrate-disconnected-media-
[Version].el6.x86_64.rpm
```

8. Run apply_snapshot.sh with the appropriate arguments

The `apply_snapshot.sh` script takes two arguments: the snapshot file you wish to apply, and the type of snapshot you wish to apply (mongodb, rabbitmq, orchestrate, all). Run the script with the appropriate argument. For instance:

```
sudo ./apply_snapshot.sh snapshot_file.tar.gz all
```
(single host installation)

```
sudo ./apply_snapshot.sh snapshot_file.tar.gz mongodb
```
(multi-host, database server only)

```
sudo ./apply_snapshot.sh snapshot_file.tar.gz rabbitmq
```
(multi-host, message queue server only)

```
sudo ./apply_snapshot.sh snapshot_file.tar.gz orchestrate
```
(multi-host, application server only)

9. Restore desired data backup. See *Data back up and restore* on page 661.
10. Restart the TeamForge Orchestrate service(s). See *System Administration* on page 657.

Data back up and restore
As part of a comprehensive backup strategy, take periodic backups of the TeamForge Orchestrate database and RabbitMQ configuration.

Once you have created a snapshot of your TeamForge Orchestrate system files, you can use this process to create periodic backups of the database and RabbitMQ configuration in order to restore data after a disaster. It is important to backup RabbitMQ configuration frequently as new users are created each time a source is created in TeamForge Orchestrate. This process does not require the shutdown of any services.

Backing up data

1. To back up data from the DB server, use the `mongodump` utility:
```
cd /opt/collabnet/mongodb sudo ./bin/mongodump -u backup -p <password> -d
orchestrate
```
The MongoDB backup user password is located in: `/opt/collabnet/mongodb/.mongo_passwords`

This will create a `dump` directory with an Orchestrate directory inside which contains the database backup dump file.

2. Backup RabbitMQ user and configuration data using `rabbitmqadmin` utility.
```
cd /opt/collabnet/rabbitmq/var/lib/rabbitmq/rabbit@localhost-plugins-expand/
rabbitmq_management-3.1.0/priv/www/cli/ sudo chmod +x rabbitmqadmin sudo ./
rabbitmqadmin -u guest -p <password> export <filename>
```
Only the guest user has permissions to backup RabbitMQ. The RabbitMQ guest user password is located in: `/opt/collabnet/rabbitmq/.rabbit_passwords`

Execution of this command requires Python 2.x, minimum 2.6 (Python 3.0 is not supported).

RabbitMQ must be running when you use the rabbitmqadmin command.

You may wish to copy or softlink this script to a more accessible location, see *http://www.rabbitmq.com/ management-cli.html*.

This script will output a file containing backup data regarding Users, Vhosts, User Persmissions, Queue information, and various parameters, policies, exchanges and bindings for advanced RabbitMQ configuration. Note, queued messages are not backed up.

Restoring from data backup

3. To restore data to the DB server, use the mongorestore utility:
```
cd /opt/collabnet/mongodb sudo ./bin/mongorestore -u backup -p <password> -d
orchestrate --drop dump/orchestrate/
```
This will restore MongoDB's Orchestrate database using the data contained in the database dump. The `--drop` parameter will remove your existing Orchestrate database before restoring the backup.

4. To restore RabbitMQ configuration, use the rabbitmqadmin utility:
```
cd /opt/collabnet/rabbitmq/var/lib/rabbitmq/rabbit@localhost-plugins-
expand/rabbitmq_management-3.1.0/priv/www/cli/ ./rabbitmqadmin -u guest -p
<password> import <filename>
```

Where <filename> is replaced by your exported RabbitMQ configuration file. The RabbitMQ guest user password is located in: `/opt/collabnet/rabbitmq/.rabbit_passwords`

Your RabbitMQ configuration will be restored.

Troubleshooting

You may encounter these problems when using TeamForge Orchestrate:

Certain activities appear grayed-out and I cannot get an activity's detail information

TeamForge Orchestrate sometimes learns about certain activities through an association reference before the details of the activity are officially reported by the source. Such activities are termed **Phantom Activities** and appear grayed-out temporarily. Once TeamForge Orchestrate receives information on this activity it creates a regular Details page and everything goes back to normal. Traceability graphs include these activities, but you cannot click on the commit nodes to get commit details.

Activities may be perpetually grayed-out when TeamForge Orchestrate is unable to retrieve activity information such as:

- Activity details of events that occurred before TeamForge Orchestrate was started — TeamForge Orchestrate begins to gather data from configured sources as soon as those connections are established, but cannot retrieve historical activity data.
- Activities that occurred during system down-time — In unusual circumstances such as power outage or network failure, TeamForge Orchestrate may never receive information about activities that occurred during such events.

Activities in the Activity stream appear out of chronological sequence

Solve this problem by synchronizing the clocks on servers that report data to TeamForge Orchestrate. The Activity stream shows activities in reverse chronological sequence. Orchestrate obtains the time stamp for activities from the originating server. Failure to synchronize the clocks of the various servers reporting

data to TeamForge Orchestrate may result in activities that appear out of sequence. On Unix systems, use NTP or similar services to synchronize server clocks.

TeamForge Orchestrate API documentation

TeamForge Orchestrate APIs can be classified into three groups: Extensibility APIs, Reporting APIs and HTTP APIs.

The API documentation here discusses the following APIs:

*   Extensibility APIs: To add new integrations to Orchestrate.
*   Reporting API: To retrieve data from the Orchestrate database.
*   HTTP APIs: To administrate Orchestrate programmatically.

Extending TeamForge Orchestrate

TeamForge Orchestrate provides a message queue (MQ) based interface for adding custom adapters and the Extensible Data Source (XDS) system for adding new product domains and tools.

Overview

Orchestrate can be extended via the Message Queue (MQ) API. The MQ API is provided for purposes of creating custom adapters for stock product classes/domains; build, commit, code review, and work item sources can post activity information to the corresponding message queues using the formats described in the API Documentation (see references below).

New product classes may be established using the Extensible Data Source API, or XDS API. For instance, TeamForge Orchestrate does not have stock product class for "deployment", so the XDS API may be used to establish a new product domain for "deployment" which can then be used like other sources.

MQ API — Writing your own adapters

Each TeamForge Orchestrate instance has an associated "message queue" service, which is used to collect and queue data from integrated tools. Extensibility is achieved by writing "adapters" that communicate specially formatted messages to the TeamForge Orchestrate message queue service.

The message communication protocol is called "AMQP" and the *RabbitMQ web site* links to open source AMQP protocol libraries that can be used to create adapters for numerous languages and platforms. The API documentation referenced below houses detailed message format specifications, example messages, and code samples for interacting with AMQP libraries.

TeamForge Orchestrate is extensible via MQ for these types of tools: SCM and version control, continuous integration/build, code review, and work item systems.

XDS API — Authoring new product/tool domains

Orchestrate can be configured to accept data from a wide range of sources through the XDS API. In addition to the stock activity APIs defined for commits, builds, reviews, and work items, the XDS feature enables you to create your own activity APIs by defining and registering an "XDS Schema" that describes the expected activity message format for a particular tool or product domain in question. Once an XDS Schema is registered with Orchestrate, adapters can send Orchestrate activity messages in compliance with the registered schema and the resulting XDS activities may show up in activity streams and as associations. In this way, XDS Schemas give users a way to publish new activity APIs for arbitrary, user-defined data sources.

XDS Schemas can be added to Orchestrate either through the MQ or HTTP API. Please read the *Extensible Data Sources (XDS) walk-through* on page 686 and refer to the API documentation below.

Once a Schema is established, "XDS Sources" can be added through the "Custom" step type. See *Extensible Data Source (XDS) Overview* on page 605.

Authentication

For the MQ interface, authentication is provisioned by the RabbitMQ server. When a new source is created, a unique username/password combination is created and exposed on the Orchestrate source edit screen. These credentials must be supplied by the posting client. These user credentials are independent of the TeamForge user store and belong only to RabbitMQ.

TeamForge Orchestrate Commits API

Submission parameters

Submission queue	• **queue_name**: orchestrate.commits • **auto_delete**: false • **durable**: true

Top-level

Parameter	Required/ Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge Orchestrate that links incoming commit messages with the appropriate source server.
		Example: "source_association_key": "31e8dd90-164f-0130-c4d0-406c8f04c05d"
commit_data	Required	The commit.

Objects

commit_data - The following fields should be in this commit post.

Field	Required/ Optional	Description
repo_id	Optional	A unique identifier for this repository. For example, the repo UUID.
revision_id	Required	A unique identifier for the commit action (eg., revision number or commit id).
branch_name	Optional	Identifier for the applicable repository branch.
type	Required	A string representing the activity type.
	Accepted values	
	post-commit - A commit has been completed.	
message	Optional	The message associated with this activity.

Field	Required/ Optional	Description
changes	Optional	An array of changed items. Required information about affected paths (an array of hashes).

Accepted values

- **path** (Required) - A string with the relative path to the file that created, modified, etc.
- **action** (Required) - A string representing the action.

  - added - Added
  - deleted - Deleted
  - modified - Modified
  - props_modified - File properties (not contents) modified
  - type_changed - Type changed (e.g., from 'file' to 'symlink', 'symlink' to 'directory', etc.)
  - copied - Path is copied from another source
  - renamed - Path is renamed from another source

- **from** (Optional) - If the activity is a copy or rename, the path from which it was copied or renamed (if it can be determined or inferred with some accuracy; this is not guaranteed to be absolutely correct)

Field	Required/ Optional	Description
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
created_by	Required	The user name of the person who took the action.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the "License"); you may not use this code except in compliance with the License. You may obtain a copy of the License at [*http://www.apache.org/licenses/LICENSE-2.0*]

Sample Submission JSON

```
{
      "api_version": "1",
      "source_association_key" :
 "150e8951-19e1-4066-8b74-601026dd6cde",
      "commit_data" : {
        "repo_id":
 "390e8951-19e1-4066-8b74-601026dd6cde",
        "revision_id": "2315",
        "branch_name": "master",
        "type": "post_commit",
        "message": "here's my commit
 [orc-277]",
        "changes": [
          { "path": "/source/
file/1", "action": "modified" },
          { "path": "/source/
file/2", "action": "added" },
          { "path": "/source/
file/3", "action": "deleted" },
          { "path": "/source/
file/4", "action": "modified" },
          { "path": "/
source/file/4", "action":
 "props_modified" }
        ],
        "event_time":
 "2012-10-02T17:15:32.320Z",
        "created_by": "username"
      }
    }
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the commit format specified on this page.

```
require 'amqp'

    QUEUE = 'orchestrate.commits'
    COMMIT = '{
      "api_version": "1",
      "source_association_key" :
 "150e8951-19e1-4066-8b74-601026dd6cde",
      "commit_data" : {
        "repo_id":
 "390e8951-19e1-4066-8b74-601026dd6cde",
        "revision_id": "2315",
        "branch_name": "master",
        "type": "post_commit",
        "message": "here\'s my
 commit [orc-277]",
        "changes": [
          { "path": "/source/
file/1", "action": "modified" },
          { "path": "/source/
file/2", "action": "added" },
```

```
            { "path": "/source/
file/3", "action": "deleted" },
            { "path": "/source/
file/4", "action": "modified" },
            { "path": "/
source/file/4", "action":
 "props_modified" }
        ],
        "event_time":
 "2012-10-02T17:15:32.320Z",
        "created_by": "username"
       }
     }'

     # Event loop
     EventMachine.run do
       connection =
 AMQP.connect('amqp://
guest:guest@example-mq')

       # Set up our RabbitMQ
 information
       channel =
 AMQP::Channel.new(connection)
       queue =
 channel.queue(QUEUE, :auto_delete
 => false, durable: true)
       exchange =
 channel.direct('')

       # Publish the activity and
 exit the loop
       exchange.publish
 COMMIT, :routing_key => queue.name
 do
         connection.disconnect
 {EventMachine.stop}
       end
     end
```

TeamForge Orchestrate Builds API

Submission parameters

Submission queue

- **queue_name**: orchestrate.builds
- **auto_delete**: false
- **durable**: true

Top-level

Parameter	Required/ Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge Orchestrate that links incoming builds with the

Parameter	Required/ Optional	Description
		appropriate source server.
		Example: "source_association_key": "6t5qM5AuLLWLkmqFJeKp"
build_data	Required	The raw build data, as provided by the build system.

Objects

build_data - The following fields should be in this build object.

Field	Required/ Optional	Description
remote_id	Required	A string to uniquely identify the build. It will appear in the UI.
duration	Required	A string containing time (in seconds) it took for a build agent to build.
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
build_url	Required	A string with the URL to the build summary. Please note that the fully qualified URL with the URL scheme is required.
created_by	Optional	A string containing the username of the person who triggered the build.
status	Required	An object that declares the status of the build object.

Accepted values

Field	Required/ Optional	Description

		• **type** (Required) - A string representing the type of the status. Please note that any type other than the listed below will be used as UNKNOWN inside TeamForge Orchestrate.
		
		  • SUCCESS
		  • FAILURE
		  • UNSTABLE
		  • ABORTED
		
		• **name** (Required) - A user-friendly display name for the status.

test_results	Optional	An object containing the test results summary. **Accepted values** • **passed_count** - A string containing the number of passed tests. • **failed_count** - A string containing the number of failed tests. • **ignored_count** - A string containing the number of ignored tests. • **url** - A string containing the URL to the test results. Please note that the fully qualified URL with the URL scheme is required.
revisions	Optional	An array of objects containing the SCM revisions related to this build. **Accepted values** • **revision** (Required) - A string containing the commit revision id. • **repository_url** (Required) - A string containing the repository url.
event_time	Required	A string containing a

Field	Required/ Optional	Description
		timestamp in UTC timezone and RFC 3339 format.
created_by	Required	The user name of the person who took the action.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the "License"); you may not use this code except in compliance with the License. You may obtain a copy of the License at [*http://www.apache.org/licenses/LICENSE-2.0*]

Sample Submission JSON

```
{
    "api_version": "1",
    "source_association_key":
 "29bf0c90-3b40-0130-ae2d-dddd5893",
    "build_data": {
    "remote_id": "700",
        "event_time":
 "2012-10-02T17:15:32.320Z",
        "duration": "200",
        "build_url": "http://
example.com/builds/600",
        "created_by": "admin",
        "status": {
            "type": "SUCCESS",
            "name": "SUCCESSFUL"
        },
        "test_results": {
            "passed_count": "10",
            "failed_count": "0",
            "ignored_count": "0",
            "url": "http://
example.com/builds/600/tests"
        },
        "revisions":
            [{ "revision":
 "fb849a2440dda438f4c6ab25f8c3266ed82d8797",
            "repository_url":
 "ssh://gitserver/project"
            }]
        }
  }
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the build format specified on this page.

```
require 'amqp'

    QUEUE = 'orchestrate.builds'
    BUILD = '{
      "api_version": "1",
```

```
        "source_association_key":
"29bf0c90-3b40-0130-ae2d-dddd5893",
        "build_data": {
            "remote_id": "700",
            "event_time":
"2012-10-02T17:15:32.320Z",
            "duration": "200",
            "build_url": "http://
example.com/builds/600",
            "created_by": "admin",
            "status": {
                "type": "SUCCESS",
                "name": "SUCCESSFUL"
            },
            "test_results": {
                "passed_count":
"10",
                "failed_count": "0",
                "ignored_count":
"0",
                "url": "http://
example.com/builds/600/tests"
            },
            "revisions":
            [{ "revision":
"fb849a2440dda438f4c6ab25f8c3266ed82d8797",
            "repository_url":
"ssh://gitserver/project"
            }]
        }
    }'
```

```
    # Event loop
    EventMachine.run do
      connection =
AMQP.connect('amqp://
guest:guest@example-mq')

      # Set up our RabbitMQ
information
      channel =
AMQP::Channel.new(connection)
      queue =
channel.queue(QUEUE, :auto_delete
=> false, durable: true)
      exchange =
channel.direct('')

      # Publish the build and exit
the loop
      exchange.publish
BUILD, :routing_key => queue.name
do
        connection.disconnect
{EventMachine.stop}
      end
    end
```

TeamForge Orchestrate Review Requests API

Submission parameters

Submission queue

• **queue_name**: orchestrate.reviews

- **auto_delete**: false
- **durable**: true

Top-level

Parameter	Required/ Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge Orchestrate that links incoming review requests with the appropriate source server.
		Example: "source_association_key": "31e8dd90-164f-0130-c4d0-406c8f04c05d"
request_data	Required	The raw review request data, as provided by the review server.

Objects

request_data - The following fields should be in this review request object.

Field	Required/ Optional	Description
remote_id	Required	A string to uniquely identify the review request.
link	Optional	A URL that points back to this individual review.
summary	Optional	A string to summarize the review request.
description	Optional	A longer string that can be used to further describe the review request. It will appear in the UI.
created_by	Required	A string containing the username of

Field	Required/ Optional	Description
		the person who submitted the request (e.g., initiated this request event, NOT necessarily the creator of the review; in some instances, this may consistently be an administrative user).
status	Required	An object that declares the status of the review request.

Accepted values

- **type** (Required) - A string representing the type of the status.

  - open - The review request is currently open for review.
  - submitted - The review request changes have been submitted.
  - rejected - The code in the request was rejected.
  - discarded - The review request has been discarded.
  - other - A status type that may not fit into the above list.
- **name** (Required) - A user-friendly display name for the status.

Field	Required/ Optional	Description
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
updated_by	Optional	A string containing the username of the person performing the update.
files	Optional	An array of objects

Field	Required/ Optional	Description
		with optional SCM revision information. If included, only the file name is required.

Accepted values

- **file** (Required) - A string with the full path to the file that was being reviewed.
- **revision** (Optional) - A string representing the revision number of the commit that includes this file.
- **repository_url** (Optional - A string with the URL to the SCM server. For associations to commits, this must match one defined as a source in the same pipeline.

Field	Required/ Optional	Description
reviewers	Optional	A array of user hashes. Optional reviewer information. If included, only the user name is required. For some systems (e.g., Crucible), reviewers are only reported when they have marked the review 'completed' or 'reviewed'; other systems (e.g., Review Board) report reviewers when they add comments or make other changes to a review.

Accepted values

- **username** (Required) - A string of the user's name.

Field	Required/ Optional	Description
		• **link** (Optional) - A string representing the URL to the users profile.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the "License"); you may not use this code except in compliance with the License. You may obtain a copy of the License at [*http://www.apache.org/licenses/LICENSE-2.0*]

Sample Submission JSON

```
{
    "api_version": "1",
    "source_association_key" :
 "31e8dd90-164f-0130-
c4d0-406c8f04c05d",
    "request_data" : {
      "remote_id": "42",
      "link": "http://reviewserver/
review/42/",
      "summary": "The summary of the
 request",
      "description": "An extended
 description of what the request
 covers",
      "status": {
        "type": "open",
        "name": "Review"
      },
      "event_time":
 "2012-10-02T17:15:32.320Z",
      "created_by": "username",
      "updated_by": "username",
      "files": [
        {
          "revision":
 "ef9f8fe661992f48c3539c05fc5c69c61918fa17",
          "file":
 "commit1_file1.txt",
          "repository_url": "ssh://
gitserver/project"
        },
        {
          "revision":
 "ef9f8fe661992f48c3539c05fc5c69c61918fa17",
          "file":
 "commit1_file2.txt",
          "repository_url": "ssh://
gitserver/project"
        },
        {
          "revision":
 "fb849a2440dda438f4c6ab25f8c3266ed82d8797",
          "file":
 "commit2_file1.txt",
          "repository_url": "ssh://
othergitserver/project"
        }
      ],
```

```
      "reviewers": [
         { "username": "adent",
 "link": "http://reviewserver/user/
adent" },
         { "username": "fprefect",
 "link": "http://reviewserver/user/
fprefect" }
         ]
      }
}
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the review format specified on this page.

```ruby
require 'amqp'

    QUEUE = 'orchestrate.reviews'
    REVIEW = '{
      "api_version": "1",
      "source_association_key" :
 "31e8dd90-164f-0130-
c4d0-406c8f04c05d",
      "request_data" : {
        "remote_id": "42",
        "link": "http://
reviewserver/review/42/",
        "summary": "The summary of
 the request",
        "description": "An extended
 description of what the request
 covers",
        "status": {
          "type": "open",
          "name": "Review"
        },
        "event_time":
 "2012-10-02T17:15:32.320Z",
        "created_by": "username",
        "files": [
          {
            "revision":
 "ef9f8fe661992f48c3539c05fc5c69c61918fa17",
            "file":
 "commit1_file1.txt",
            "repository_url":
 "ssh://gitserver/project"
          },
          {
            "revision":
 "ef9f8fe661992f48c3539c05fc5c69c61918fa17",
            "file":
 "commit1_file2.txt",
            "repository_url":
 "ssh://gitserver/project"
          },
          {
            "revision":
 "fb849a2440dda438f4c6ab25f8c3266ed82d8797",
            "file":
 "commit2_file1.txt",
```

```
            "repository_url":
"ssh://othergitserver/project"
        }
      ],
      "reviewers": [
        { "username": "adent",
 "link": "http://reviewserver/user/
adent" },
        { "username": "fprefect",
 "link": "http://reviewserver/user/
fprefect" }
      ]
    }
  }'

  # Event loop
  EventMachine.run do
    connection =
 AMQP.connect('amqp://
guest:guest@example-mq')

    # Set up our RabbitMQ
 information
    channel =
AMQP::Channel.new(connection)
    queue =
channel.queue(QUEUE, :auto_delete
=> false, durable: true)
    exchange = channel.direct('')

    # Publish the activity and
exit the loop
    exchange.publish
REVIEW, :routing_key => queue.name
do
      connection.disconnect
{EventMachine.stop}
    end
  end
```

TeamForge Orchestrate Work Item API

Standards

Integrating TeamForge Orchestrate with the Work Item systems requires for the adapters to be able to send the configuration details of the underlying systems (trackers configuration etc. (see "TODO put a link on configuring work item sources")) as well as the actual data (work items) generated by these systems. These are two distinct types of messages that use different formats.

TeamForge Orchestrate Work Item Configuration API

Submission parameters

Submission queue	• **queue_name**: orchestrate.work_items • **auto_delete**: false • **durable**: true

Top-level

Parameter	Required/ Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge Orchestrate that links incoming work items with the appropriate source server.
		Example: "source_association_key": "6t5qM5AuLLWLkmqFJeKp"
workitem_settings	Required	Tracker configuration details.

Objects

workitem_settings - The following fields should be in this workitem_settings object.

Field	Required/ Optional	Description
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
url	Required	A string with the browse URL for the system. Orchestrate will append an individual work item's id to that url to link back to the work item
tracker	Required	Tracker configuration information. A JSON object describing the structure of the tracker.

Accepted values

- **name** (Required) - A name for the tracker.
- **id** (Required) - A string containing a unique (on the

Field	Required/ Optional	Description

reporting system) identifier for the tracker.

- **key** (Optional) - A string containing the tracker key (usually used to prefix the work-item ids).
- **regex** (Required) - A string containing a regular expression that Orchestrate will use to associate work-items with commits using the commit message.
- **icon** (Optional) - A string containing a URL for this tracker's icon.
- **statuses** (Required) - An array of status objects associated with the tracker.

  - **id** (Required) - A string containing a unique(on the underlying system) identifier for the status.
  - **name** (Required) - A string containing the name of the status.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the "License"); you may not use this code except in compliance with the License. You may obtain a copy of the License at [*http://www.apache.org/licenses/LICENSE-2.0*]

Sample Submission JSON

```
{
      "api_version": "1",
      "source_association_key" :
 "150e8951-19e1-4066-8b74-601026dd6cde",
      "workitem_settings" : {
      "url": "http://
www.example.com/jira",
      "event_time":
 "2012-10-02T17:15:32.320Z",
      "tracker": {
                    "name":
 "Orchestrate/Stories",
                    "id":
 "01234567abcdef01234567",
                    "key":"ORC",
                    "regex": "ORC-\d
+",
                    "icon": "http://
www.example.com/jira/browse/
bug.jpig",
                    "statuses": [
```

```
                            {"name":
"Open", "id": "abcdef"},
                            {"name": "In
progress", "id": "123abc"},
                            {"name":
"Closed", "id": "456123"}
                    ]
                }
        }
}
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the build format specified on this page.

```ruby
require 'amqp'

    QUEUE =
'orchestrate.work_items'
    WORK_ITEM_CONFIGURATION = '
    {
      "api_version": "1",
      "source_association_key" :
"150e8951-19e1-4066-8b74-601026dd6cde",
      "workitem_settings" : {
          "url": "http://
www.example.com/jira",
          "event_time":
"2012-10-02T17:15:32.320Z",
          "tracker":  {
                            "name":
"Orchestrate/Stories",
                            "id":
"01234567abcdef01234567",

"key":"ORC",
                            "regex":
"ORC-\d+",
                            "icon":
"http://www.example.com/jira/
browse/bug.jpig",

"statuses": [

{"name": "Open", "id": "abcdef"},

  {"name": "In progress", "id":
"123abc"},

{"name": "Closed", "id": "456123"}
                            ]
                    }
        }
    }'
    # Event loop
    EventMachine.run do
      connection =
AMQP.connect('amqp://
guest:guest@example-mq')
```

```
        # Set up our RabbitMQ
information
        channel =
AMQP::Channel.new(connection)
        queue =
channel.queue(QUEUE, :auto_delete
=> false, durable: true)
        exchange =
channel.direct('')

        # Publish the work item
configuration and exit the loop
        exchange.publish
WORK_ITEM_CONFIGURATION, :routing_key
=> queue.name do
          connection.disconnect
{EventMachine.stop}
        end
      end
```

TeamForge EventQ Work Item Message API

Submission parameters

Submission queue

- **queue_name**: eventq.work_items
- **auto_delete**: false
- **durable**: true

Top-level

Parameter	Required/ Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge EventQ that links incoming work items with the appropriate source server.
		Example: "source_association_key": "6t5qM5AuLLWLkmqFJeKp"
workitem_settings	Required	The work item data.

Objects

work_item - The following fields should be in this work item object.

Field	Required/ Optional	Description
created_by	Optional	A string containing the username of the person who

Field	Required/ Optional	Description
		submitted the work item.
updated_by	Optional	A string containing the username of the person performing the update.
id	Required	A string to uniquely identify the work item.
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
summary	Required	A string to summarize the work item.
description	Optional	A longer string that can be used to further describe the work item.
status_id	Required	A string to uniquely identify the status. This value must match the identifier within the work item configuration message.
tracker_id	Required	A string to uniquely identify the tracker. This value must match the identifier within the work item configuration message.
closed	Optional	A boolean for the open or closed state of the work item. True if the work item is closed.

Field	Required/ Optional	Description
deleted	Optional	A boolean for the deleted state of the work item. True if the work item has been deleted.
priority	Optional	A string containing the priority of the work item.
assigned_to	Optional	A string containing the username of the person who is assigned to the work item.
creation_time	Optional	A string containing a timestamp in UTC timezone and RFC 3339 format. The time the item was first created in the source system.
tags	Optional	An array of strings containing the tags, or labels, for this work item.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the "License"); you may not use this code except in compliance with the License. You may obtain a copy of the License at [*http://www.apache.org/licenses/LICENSE-2.0*]

Sample Submission JSON

```
{
     "api_version": "1",
     "source_association_key" :
"150e8951-19e1-4066-8b74-601026dd6cde",
     "work_item": {
         "created_by": "admin",
         "updated_by": "username",
         "summary": "title",
         "id": "KDA-59",
         "description":
"description",
         "status_id": "10001",
         "event_time":
"2014-03-12T20:47:41.295Z",
```

```
        "closed": false,
        "deleted": false,
        "tracker_id": "10200-2",
        "priority": "Major",
        "assigned_to": "admin",
        "creation_time":
"2014-03-29T00:26:20.091Z",
        "tags": [
          "Tag1",
          "Tag2",
          "Tag3"
        ]
    }
}
```

Ruby Submission Example

You will need to change the following example before you can run it. Set the Rabbit MQ hostname and the association key, and paste a JSON block following the build format specified on this page.

```ruby
require 'amqp'

    QUEUE = 'eventq.work_items'
    WORK_ITEM = '
    {
      "api_version": "1",
      "source_association_key" :
"150e8951-19e1-4066-8b74-601026dd6cde",
      "work_item": {
          "created_by": "admin",
          "updated_by": "admin",
          "summary": "title",
          "id": "KDA-59",
          "description":
"description",
          "status_id": "10001",
          "event_time":
"2014-03-12T20:47:41.295Z",
          "closed": false,
          "deleted": false,
          "tracker_id": "10200-2",
          "priority": "Major",
          "assigned_to": "admin",
          "creation_time":
"2014-03-29T00:26:20.091Z",
          "tags": [
              "Tag1",
              "Tag2",
              "Tag3"
          ]
      }
    }'
    # Event loop
    EventMachine.run do
      connection =
AMQP.connect('amqp://
guest:guest@example-mq')

      # Set up our RabbitMQ
information
      channel =
AMQP::Channel.new(connection)
```

```
        queue =
channel.queue(QUEUE, :auto_delete
=> false, durable: true)
        exchange =
channel.direct('')

        # Publish the work item and
exit the loop
        exchange.publish
WORK_ITEM, :routing_key =>
queue.name do
          connection.disconnect
{EventMachine.stop}
        end
      end
```

Extensible Data Sources (XDS) walk-through

This walk-through is intended to provide a detailed step-by-step implementation example of the Extensible Data Sources (or XDS) feature. Please read the parent overview document first: *Extending TeamForge Orchestrate* on page 663

Setting the stage: Orchestrate does not ship with a stock activity API for deployment tools, so our fictional example lifecycle includes a process to deploy software into various environments. Orchestrate users would like to see these deployment activities bubble up to activity streams and any deployment associations in the context of other activities. So let's define an XDS Schema modeling the "deployment" domain and then send activities to the newly registered XDS Activity API for deploy.

Define the Schema Attributes - The first step is craft an XDS Schema to adequately describe our deployment domain space and the events we expect to send to Orchestrate. The submission of the data type schema is itself governed by an API, which is documented at *XDS Schema API*.

1. API version

   The version number of the XDS Schema API (not your new schema's version number).

2. Source association key

   This value is obtained by creating a new "XDS Source" in Orchestrate's web client. Supplying a source association key is optional during schema registration. If provided, the source will be associated with the schema when the schema is received. Otherwise, the XDS source will be associated to a registered schema when the first activity message is received. There is no "preferred" option; this flexibility exists purely to suit various types of integration models.

3. Schema name

   Should be a human-friendly identifier of the data source.

4. Schema ID

   Takes a UUID as the value and the value should be unique per schema. If UUID generation is not possible, characters are limited to alphanumerics and the dash symbol.

5. Schema version

   Allows for the same schema to be updated without sending a separate schema ID. This is an integer field, best practice is to start at 1 and increment.

6. Event time

   The time at which your newly crafted schema is submitted to Orchestrate. Requires UTC RFC3339 format.

   The schema ID is user-defined; we suggest a long "UUID" style string. The schema name is also user-defined. In our deploy example, we can add a schema ID and name as follows:

```
// Example Deploy XDS Schema
{
  "api_version": "1",
  "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
```

```
  "custom_schema": {
     "name": "Faux Schema",
     "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
     "schema_version": 2,
     "event_time": "2015-02-02T17:15:32.320Z",
  }
}
```

Note, the source association key value was obtained from our instance of Orchestrate. It is an attribute of the parent XDS Collection with which you intend to register the new schema.

Defining the fields expected by your schema - Since you are defining an API at this stage, the next step is to define the list of fields expected by your API. Fields are the attributes you wish to supply with each event message sent by an adapter. We'll address optional vs. required fields in the next section, but for now we want a comprehensive list of all fields represented in your new activity message format. For instance, in our deploy example you may want to supply deploy environment, time, duration of deploy activities, etc.

The XDS Activity API categorizes fields into three classes: mandatory fields, stock fields, and custom fields. "Mandatory fields" are required in all registered schemas and thus must be present in all activity messages. "Stock fields" are a set of non-mandatory fields that have special relevance to the Orchestrate application, such as appearing in specific places in the user interface. "Custom fields" are truly arbitrary and may be used to capture desired data not encompassed by mandatory or stock fields.

7. **Mandatory fields** - all of these fields are required in every XDS schema.

    a) schema_id

       Activity messages will need to send this to identify themselves as governed by a specific schema.

    b) schema_version

       Should match the same schema_version as sent by the schema. This makes sure that the message is displayed with the correct version of the schema.

    c) event_time

       The UTC RFC3339 time of the event.

    d) remote_id

       Represents the source tool's identifier for the activity. For instance, if we were modeling Subversion commits, the remote_id would be the revision number. Messages with the same remote_id are treated as part of the same activity and are aggregated as "sub-events".

8. **Stock fields** - Optional fields with special relevance in the Orchestrate application.

    a) associated_remote_id

       If your activity has an association this field is used to specify one or more IDs for purposes of association.

    b) created_by

       The user who instantiated the activity.

    c) summary

       The primary heading or title of the activity.

    d) description

       A detailed description of the activity.

    e) status_name

       The current status of the activity, corresponds to the "status_type" field.

    f) status_type

       The current status class, either "open", "success", or "fail" each of which maps to status icons displayed in the user interface.

    g) link_to

       Creates a link back to the activity on the source system. The link_to field can take an array with a link label along with the URL to customize the display text in the user interface. The default label is "See More". Following our example, one could display "Deployment Details" in the user interface in place of the default "See More".

Let's return to our deployment example and see how fields are defined in the Data Type Schema:

```
// Example Deploy XDS Schema
{
  "api_version": "1",
  "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
  "custom_schema": {
      "name": "Faux Schema",
      "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
      "schema_version": 2,
      "event_time": "2015-02-02T17:15:32.320Z",
      "fields": ["schema_id", "schema_version", "event_time",
  "remote_id",
        "associated_remote_id", "summary", "status_name", "status_type",
        "link_to", "rule", "deploy_url", "deploy_message"],
    }
}
```

Note that we've included the four mandatory fields, a couple of "stock fields", and some custom fields modeling our "deploy" specific domain data.

Requiring specific fields - You may impose required fields. Remember that some fields are always required, so this section is not optional for that reason. But you can extend the set of required fields beyond those required by the API.

9. required

Pass in an array of required fields. Any activity message without all of these fields will be discarded.

Let's return to our deploy example and see how fields are defined in the Data Type Schema:

```
// Example Deploy XDS Schema
{
  "api_version": "1",
  "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
  "custom_schema": {
      "name": "Faux Schema",
      "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
      "schema_version": 2,
      "event_time": "2015-02-02T17:15:32.320Z",
      "fields": ["schema_id", "schema_version", "event_time", "remote_id",
        "associated_remote_id", "summary", "status_name", "status_type",
        "link_to", "rule", "deploy_url", "deploy_message"],
      "required": ["schema_id", "schema_version", "event_time",
  "remote_id", "rule"],
    }
}
```

The four fields required by the API are represented, along with one additional field we added above named "rule". Any activity message that fails to supply all five of these fields will be discarded.

Select "important" fields that result in activity stream posts - Not every event that triggers an activity message is necessarily worthy of a post in the activity stream. List only those fields that you deem worth of an activity stream update in the "important" field.

10. important

Array of field names. Only changes to these fields will trigger a new entry in the activity stream.

In this case, we only want changes to the "summary" or "status_name" field to trigger activity stream posts:

```
// Example Deploy XDS Schema
{
  "api_version": "1",
  "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
  "custom_schema": {
```

```
        "name": "Faux Schema",
        "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
        "schema_version": 2,
        "event_time": "2015-02-02T17:15:32.320Z",
        "fields": ["schema_id", "schema_version", "event_time", "remote_id",
          "associated_remote_id", "summary", "status_name", "status_type",
          "link_to", "rule", "deploy_url", "deploy_message"],
        "required": ["schema_id", "schema_version", "event_time",
   "remote_id", "rule"],
        "important": ["summary", "status_name"],
    }
}
```

Parse certain fields for TeamForge object references - Say you have a field that often contains a reference to a tracker artifact and you want Orchestrate to automatically convert those into links. An association will also be created if Orchestrate is aware of the referenced object.

In addition, any URL reference will be converted to a HTML link in the Orchestrate user interface. Say, www.example.com is referenced in a "parse" field, then it will be converted to a HTML link, e.g. www.example.com

11. parse

Array of field names. Converts any TeamForge object references to links, converts URL references to links.

```
// Example Deploy XDS Schema
{
  "api_version": "1",
  "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
  "custom_schema": {
      "name": "Faux Schema",
      "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
      "schema_version": 2,
      "event_time": "2015-02-02T17:15:32.320Z",
      "fields": ["schema_id", "schema_version", "event_time", "remote_id",
        "associated_remote_id", "summary", "status_name", "status_type",
        "link_to", "rule", "deploy_url", "deploy_message"],
      "required": ["schema_id", "schema_version", "event_time",
   "remote_id", "rule"],
      "important": ["summary", "status_name"],
      "parse": ["deploy_message"],
  }
}
```

By adding "deploy_mesage" to the "parsed" field, any CTF object will be linked and possibly associated. Any web addresses referenced will also be converted to a full HTML link.

Register your schema with Orchestrate - Now that your new schema is formulated, register your schema with your Orchestrate instance using either:

- **orchestrate.custom** message queue, or
- **XDS Schema HTTP API** (see /api/1/docs/http_api_xds_schema)

There is no preferred method for schema registration; pick the method that works best for your integration scenario.

Registered schemas may be used site-wide. In our deploy example, perhaps we've written an extension to our deploy product and have registered its schema. The same schema may be reused by multiple sources (e.g., multiple instances of the deploy product) across the site.

Note also that successive submissions of the same schema ID and version will be ignored and it is a safe practice to send the same schema multiple times if it is programmatically convenient to do so.

To modify a schema, increment the schema_version value and re-register the schema with Orchestrate using the methods described above.

12. Success

You can use the **XDS Schema HTTP API** (see /api/1/docs/http_api_xds_schema) to inspect registered schemas and their versions.

13. Troubleshooting

Check the server log "custom_queue.log" to see if the schema was rejected and why it was rejected.

Send Activity messages - Now that you've defined an API for your tool/domain you can start supplying activity messages to Orchestrate that coincide with your new schema. For a technical reference on crafting activity messages, see the *XDS Schema API* documentation.

14. Submit XDS Activity messages to orchestrate.custom queue

Note that the activity stream will only reflect new entries when an "important" field is modified or set in an activity message.

```
// Example Deploy XDS Activity Message
{
  "api_version": "1",
  "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
  "custom_data": {
    //required fields
    "name": "Faux Schema",
    "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
    "schema_version": 2,
    "event_time": "2015-02-02T17:15:32.320Z",
    "remote_id": "deploy-1234",
    "rule": "1 comes before 2",
    // stock fields
    "summary": "Deploy at 4:52pm on a Friday",
    "deploy_message": "Deploying to fix [artf12345] and [artf12346] as
 critical bugs.",
    "associated_remote_id": "b4c8e50c908ade5c6f053e1ef6422f88",
    "link_to": ["Deploy Details", "http://www.example.com/deploy/
deploy-1234"],
    "status_type": "open",
    // other custom fields
    "environment": "Dev Stage 1"
  }
}
```

Optional Features - The XDS API offers some optional features to help pretty things up and make life a bit easier.

15. Interpolate TeamForge URLs

It may be desirable to create URL references to TeamForge pages or functions (like SSO authentication). There are two interpolation functions in XDS:

Reference TeamForge base URLs: `%CTF_BASE_URL%`

Reference TeamForge "Go" URLs: `%CTF_GO_URL%`

In our example, we may wish to reference a TeamForge Go URL like so:

```
// Example Deploy XDS Activity Message
...
"summary": "Deploy at 4:52pm on a Friday. See %CTF_GO_URL%/page1001",
```

16. Control field visibility

Say you wanted to load specific data into Orchestrate's database for reporting purposes, but you don't want to expose all of those fields in the user interface. The solution is to add those fields to the hidden array in the XDS Schema message. The hidden array takes only custom fields (optional, non-stock). Custom fields are visible by default.

Continuing our example, the newly introduced "environment" and "authorized_by" custom fields will be stored in the Orchestrate database, but not exposed in the user interface:

```
// Example Deploy XDS Schema
{
  "api_version": "1",
  "source_association_key" : "8578c900-f8df-0131-84ff-3c07547a48b0",
  "custom_schema": {
     "name": "Faux Schema",
     "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0",
     "schema_version": 2,
     "event_time": "2015-02-02T17:15:32.320Z",
     "fields": ["schema_id", "schema_version", "event_time", "remote_id",
 "associated_remote_id",
       "summary", "status_name", "status_type", "rule", "deploy_url",
 "deploy_message",
       "environment", "authorized_by"],
     "required": ["schema_id", "schema_version", "event_time",
 "remote_id", "rule"],
     "important": ["summary", "status_name"],
     "parse": ["deploy_message"],
     "link_to": ["deploy_url"],
     "hidden": ["environment","authorized_by"]
   }
}
```

TeamForge Orchestrate XDS Schema API

Submission parameters

Submission queue

- **queue_name**: orchestrate.custom
- **auto_delete**: false
- **durable**: true

Top-level

Parameter	Required/ Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Optional	A key generated by TeamForge Orchestrate that links incoming work items with the appropriate source server. (Optional)
		Example: "source_association_key": "kLM56uWtKAqpeLLm5qJF"
custom_schema	Required	a schema for the XDS events.

Objects

custom_schema - The fields specified below define the schema that activities will conform to.

Field	Required/ Optional	Description
name	Required	A visible name that will allow pipeline-creators to recognize this schema when populating a pipeline.
schema_id	Required	A unique id defining this schema. Every activity sent by a single adapter should reference the same schema id. If this changes, the data source is considered a new data source, and data will not populate into pipelines that listened for the old one. If the schema changes, you can use the version to show that it's intended to be parsed differently, but continue to deliver messages to existing pipelines.
		Example: "schema_id": "fc358dd0-11dc-0132-b9fe-3c07547a48b0"
schema_version	Required	A string to uniquely identify the work item.
		Example: "schema_version": 1
event_time	Required	A string containing a timestamp in UTC timezone and RFC 3339 format.
		Example: "event_time": "2012-10-02T17:15:32.320Z"

Field	Required/ Optional	Description
fields	Required	A string to summarize the work item.
		Example: "fields": ["schema_id", "schema_version", "event_time", "remote_id", "assigned_to", "hours_played", "severity", "canary"]
required	Required	A longer string that can be used to further describe the work item.
		Example: "required": ["schema_id", "schema_version", "event_time", "remote_id"]
important	Optional	A string to uniquely identify the status. This value must match the identifier within the work item configuration message.
		Example: "important": ["assigned_to", "severity"]
parsed	Optional	A string to uniquely identify the tracker. This value must match the identifier within the work item configuration message.
		Example: "parsed": ["description"]
hidden	Optional	A boolean for the open or closed state of the work item. True if the work item is closed.
		Example: "hidden": ["comment"]

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the "License"); you may not use this code except in compliance with the License. You may obtain a copy of the License at [*http://www.apache.org/licenses/LICENSE-2.0*]

Sample Submission JSON

```
{
    "api_version": "1",
    "source_association_key" :
 "8578c900-
f8df-0131-84ff-3c07547a48b0",
    "custom_schema": {
      "name": "Faux Schema",
      "schema_id":
 "fc358dd0-11dc-0132-
b9fe-3c07547a48b0",
      "schema_version": 2,
      "event_time":
 "2012-10-02T17:15:32.320Z",
      "fields": ["schema_id",
 "schema_version",
 "event_time", "remote_id",
 "associated_remote_id",
 "title", "rule", "deploy_url",
 "deploy_message", "comment"],
      "required": ["schema_id",
 "schema_version", "event_time",
 "remote_id", "rule"],
      "important": ["rule"],
      "parsed": ["deploy_message"],
      "hidden": ["comment"]
    }
}
```

TeamForge Orchestrate XDS Activity API

Submission parameters

Submission queue

- **queue_name**: orchestrate.custom
- **auto_delete**: false
- **durable**: true

Top-level

Parameter	Required/ Optional	Description
api_version	Required	A string that matches the API version this document is for.
source_association_key	Required	A key generated by TeamForge Orchestrate that links incoming work items with the appropriate source server.

Parameter	Required/ Optional	Description
		Example: "source_association_key": "kLM56uWtKAqpeLLm5qJF"
custom_data	Required	The object containing custom activity data conforming to the referenced schema/version.

Objects

custom_data - The fields specified below conform to the schema referenced, and provide activity information to Orchestrate:

Field	Required/ Optional	Description
schema_id	Required	The identifier of the XDS schema this message conforms to. The schema needs to be sent before an activity referencing that schema is sent.
schema_version	Required	The version of the referenced schema this message conforms to.
event_time	Required	The identifier of this activity; this is used to collapse related activities together.
remote_id	Required	A string to summarize the work item.
Stock Fields	Optional	
		The following field names are treated specially by Orchestrate, allowing a developer to show specific data in our UI where it can fit with our display patterns. All the fields are optional.
		• **associated_remote_id** - Specify the singular ID or array of IDs that your event should associate with, these IDs will be searched

Field	Required/Optional	Description
		for in the source chosen in the pipeline custom source and shown in the Association section of the Details page. • **created_by** - The username or email address of the person who took the action. This is shown in the Details page. • **updated_by** - The username or email address of the person who took an action. This is shown in the Details page. • **summary** - The main title of the event. Shown in bold at the top of the Details page and also in the Activity Stream. • **description** - Detailed description of the event. Shown in full in the Details page, a truncated version is shown in the Activity Stream. • **status_name** - The name of the current status. This is a text field that is displayed in the Details page and Activity Stream. The value should be representational of the current state of the activity and match with the • **status_type**. status_type - The type of status icon to show, containing values of "**open**", "**success**", "**failed**", each of which maps to a status icon displayed in the user interface. • **link_to** - The link back to the event on the source system. It take an array or single string. Array takes a URL along with the link label to customize the display text in the user interface. If only a URL string is passed without the label, the default label is "See More".
Custom Fields	Optional	These custom fields can only contain String data and will be displayed in a key value list in the Details page.

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the "License"); you may not use this code except in compliance with the License. You may obtain a copy of the License at [*http://www.apache.org/licenses/LICENSE-2.0*]

Sample Submission JSON 1: A deploy example with string value for link_to field

```
{
    "api_version": "1",
    "source_association_key" :
 "8578c900-
f8df-0131-84ff-3c07547a48b0",
    "custom_data": {
        //required fields
        "schema_id":
 "fc358dd0-11dc-0132-
b9fe-3c07547a48b0",
        "schema_version": 2,
        "event_time":
 "2012-10-02T17:15:32.320Z",
        "remote_id": "deploy-1234",
        // stock fields
        "created_by": "Build-bot",
        "updated_by": "Deploy-bot",
        "summary": "Deploy at 4:52pm
 on a Friday",
        "description": "Deploying
 to fix [FOO-245] and [BAR-900] as
 critical bugs.",
        "associated_remote_id":
 "b4c8e50c908ade5c6f053e1ef6422f88",
        "link_to": "http://
www.example.com/deploy/
b4c8e50c908ade5c6f053e1ef6422f88",
        "status_name": "Deploying",
        "status_type": "open",
        // custom fields
        "environment": "production",
        "branch": "master"
    }
}
```

Sample Submission JSON 2: A build example with array value for the link_to field

```
{
    "api_version": "1",
    "source_association_key" :
 "1ebbbef0-2025-0132-8f68-3c15c2d51622",
    "custom_data": {
        //required fields
        "schema_id":
 "5193ca40-2025-0132-8f68-3c15c2d51622",
        "schema_version": 1,
        "event_time":
 "2012-10-02T17:15:32.320Z",
        "remote_id": "build-1234",
        // stock fields
        "created_by": "Build-bot",
        "updated_by": "Deploy-bot",
        "summary": "Build
 successful!",
        "description": "Build
 containing commits 109 through
 112",
```

```
          "associated_remote_id":
 ["109","110","111","112"],
          "link_to": ["http://
www.example.com/build/build-1234",
 "Build-1234"],
          "status_name": "Success",
          "status_type": "success",
          // custom fields
          "tests_passed": "1400",
          "tests_failed": "0",
          "tests_pending": "23"
      }
}
```

TeamForge Orchestrate Reporting API

Overview

- TeamForge Orchestrate Reporting API provides a means to query activity data in TeamForge Orchestrate. This API is for data retrieval only.
- Submission URL: **https://<orchestrate_hostname>/orc/api/<version>/reporting**
- Method: **POST (only)**
- Authentication: Requires a valid TeamForge SOAP session id.
- Authorization: Requires the TeamForge Orchestrate "PIPELINE READ", "REPORTING API" permission, or site admin permissions.

Convention

API requests are constructed from two building-blocks: commands with parameter(s) that can be chained one after another, followed by a single return type with its parameter(s). As such, API requests take the form:

```
command parameter(s)
command parameter(s)
command parameter(s) ...
return type parameter(s)
```

- Each chained command is treated as an "AND" query filter
- Use separate queries to approximate "OR" requests
- Requests are limited to a single return type which must be supplied after the last command
- A query itself is a String constructed by chaining commands each separated from the next (or return) by a new line '\n'.

  - This '\n' is very important since the parser looks for this when tokenizing the query

Authentication and Authorization

☞ **Note:** In order to use the reporting api, your reporting client must have access to the endpoint of your TeamForge instance.

When sending query requests to the reporting api, a custom header named 'X-Orchestrate-Session' must be added to each http post request, the value of which is the 'session id' returned from a successful login call to the TeamForge endpoint. The TeamForge endpoint is found here: http(s)://TeamForgeHost/ce-soap60/services/CollabNet

To gain access to the Reporting API the user must have one either of the below permissions set in TeamForge:

- TeamForge SITE ADMINISTRATOR

  - This will provided access to all active projects in Orchestrate
- REPORTING API Orchestrate Permissions on at least one project

- This limits the projects to query against to only those which the user has REPORTING API permissions
- If the user is both SITE ADMIN and REPORTING API, SITE ADMIN will be used. There is no way to limit a SITE ADMIN's access

See the Examples below for a simple Java client.

Commands and Parameters

Command	Description
restrict_to *(string)*	Filter query results for certain classes of activity objects.
	Example: restrict_to reviews
	Accepted values: builds, commits, reviews, work_items, custom.
associated_to *(array)*	Takes one or more object IDs and returns a list of activity summaries with associations to the object IDs specified.
	Example: associated_to ["artf1002", "artf1002", "artf1003"]
associations *(none)*	Used in command chains to modify query results to return associations. No parameters.
	Example: associations
in_project *(array)*	Restrict query results to activities in a given project. Takes array of project IDs as a parameter.
	Example: in_project ["proj1123", "proj1134"]
sort_by *(array)*	Sort the result set by the fields of the objects which are supplied as a parameter, ascending or descending order.
	Example: sort_by ["status", "asc"]
	Accepted values: any object field; see MQ API docs for object descriptions or XDS schema for custom objects. For instance, builds may be sorted by remote_id, status, test_results, duration, created_by, or time_created.
	Accepted values: asc, desc
has_any *(string)*	Filter query results to those activities that have matching class associations.
	Example: has_any "builds"
	Accepted values: builds, commits, review, work_items, custom.
has_none *(string)*	Filter query results to those activities that do not have matching class associations.
	Example: has_none "reviews"
	Accepted values: builds, commits, review, work_items, custom.
limit *(integer)*	Limit query results to the specified integer for pagination of results and performance. May be used in conjunction with offset.
	Example: limit 100

Command	Description
	Accepted values: positive integers.
offset *(integer)*	Offset query results to specified integer for pagination of results and performance. May be used in conjunction with limit.
	Example: offset 10
	Accepted values: positive integers.
before *(date/time)*	Filter results set to those activity summaries with events that occurred prior to the supplied date and time. May be used in conjunction with after.
	• **Example**: before 2014-10-02T17:15:45.320Z • **Example**: before 2.weeks.ago
	Accepted values: RFC 3339 format date times in UTC and Ruby-style date references (minutes, hours, days, weeks, months, years).
after *(date/time)*	Filter results set to those activity summaries with events that occurred after the supplied date and time. May be used in conjunction with before.
	• **Example**: after 2014-11-02T12:45:45.320Z • **Example**: before 1.day.ago
	Accepted values: RFC 3339 format date times in UTC and Ruby-style date references (minutes, hours, days, weeks, months, years).
tagged_as *(array)*	Filter results set to those activities from sources labeled with the supplied tag. Takes multiple tags to create an OR condition. Two or more tagged_as commands may be chained which constructs an AND condition, returning only those activities with tag A and tag B.
	Example: tagged_as ["deploy","binary repository"] tagged_as ["production"]

Return types

Conclude your query with a single "return" method to transform the data and return the results. All results are returned wrapped in a top-level json hash containing the key items, which contains the results of the query. The return may also contain the optional keys source_display_data, containing a hash of display metadata for the associated sources keyed by source id, and work_item_urls, containing urls to external work items keyed by id (if applicable).

Return example

```
{
  "items": [
    {
      "_type": "WorkItemSummary",
      "assigned_to": "dev-bot",

 "associated_activity_summary_ids":
 [

      ],
      "associated_ctf_ids": [
```

```
    ],
    "closed": true,
    "comments": [

    ],
    "created_at":
"2015-04-22T20:54:26.063Z",
    "created_by": "itembot!",
    "creation_time":
"2014-03-29T00:26:20.091Z",
    "deleted": false,
    "description": "Autogenerated
workitem description",
    "id":
"55380a82109a01555d00000c",
    "opaque_id": "6kkd",
    "priority": "mecha-critical",
    "remote_id": "AUTO-6",
    "remote_status_id": "3",
    "settled": true,
    "source_id":
"5537ed65109a012cec000004",
    "summary": "Release the
integration",
    "tags": [
      "up",
      "strange"
    ],
    "time_created":
"2015-04-22T20:54:26.000Z",
    "updated_at":
"2015-04-23T16:49:19.657Z"
    }
  ],
  "source_display_data": {
    "5537ed65109a012cec000004": {
      "_type": "WorkItemSource",
      "_links": {
        "type_icon": {
          "href": "http://
ctf.example.com/orc/assets/icons/
work-item.png",
          "type": "image/png"
        },
        "source_icon": {
          "href": "http://
external.example.com/images/icons/
issuetypes/bug.png",
          "type": "image/png"
        }
      },
      "name": "External Bug
Tracker",
      "background_color": "#f2511b"
    }
  },
  "work_item_urls": {
    "55380a82109a01555d00000c":
"http://tracker.example.com/
a_tracker_source/AUTO-6"
  }
```

```
}
```

Return Keys

Return Key	Description
items *(array or hash)*	Query-dependent results.
source_display_data *(hash)*	Display metadata for sources references in results.
work_item_urls *(hash)*	URLs for referenced work items. This key is present in the return structure only if 'items' contains WorkItemSummary objects.

Query Return Data Types

Data item	Description
fields *(array or hash)*	Returns json formatted results set limited to the fields supplied as parameters.
	Example: fields ["time_created", "remote_id"]
	Accepted values: any set of object fields; see MQ API docs for object descriptions or XDS schema for custom objects. For instance, builds may take: remote_id, status, test_results, duration, created_by, or time_created.
field_contains *(nested array)*	Returns json formatted results set limited to exact value supplied as parameters.
	Example: field_contains ["remote_id", ["artf1001", "art1002"]]
	Accepted values: any object field with matching value set; see MQ API docs for object descriptions or XDS schema for custom objects. For instance, builds may take: remote_id, status, test_results, duration, created_by, or time_created.

Data item	Description
group_by_field *(string)*	Returns json formatted results set grouped by the field supplied as parameter. **Example**: group_by_field "status" **Accepted values**: created_at, remote_id, status
count *(none)*	Returns a json structure with a count of all matching activities. **Example**: count
timeseries *(date/time)*	Useful for charting applications, the output will be a hash, where the keys are the starting points of the time series, and the value is the array of objects that fall into that series. It takes a Ruby-style date reference parameter to specify the results granularity. Requires before and after in the same query. **Example**: timeseries 5.minutes **Accepted values**: Ruby-style date references (minutes, hours, days, weeks, months, years).

Examples

The following code examples are Copyright 2015 CollabNet, Inc., licensed under the Apache License, Version 2.0 (the "License"); you may not use this code except in compliance with the License. You may obtain a copy of the License at [*http://www.apache.org/licenses/LICENSE-2.0*]

Use Cases and Query Examples	Use case: "Find the number of all failed builds across all projects"

	``
	#Query:
	  restrict_to builds
	  field_contains ["status_type",
	["FAILURE"]]
	  count
	``

Use case: "Given a set of artfs as inputs, give me all the associated activities"

```
#Query:
  associated_to: ["artf1234",
"artf1235", "artf1236"]
```

Use case: "Across two specific projects, give me the builds that pass vs fail"

```
#Query:
  in_project ["proj1001",
"proj1002"]
  restrict_to builds
  group_by_field "status_type"
```

Use case: "Given a set of artfs as inputs (say, in a planning folder) give me build stats (total #, pass, fail, etc)"

```
#Query:
  associated_to ["artf1001",
"artf1002", "artf1003"]
  associations
  restrict_to builds
  group_by_field "status_type"
```

Use case: "For a given user, give me a list of commits over X timeline, associated build stats"

```
#Query:
  restrict_to commits
  after 2.weeks.ago
  before 1.week.ago
  field_contains [ "created_by",
["jdoe"] ]
  associations
  restrict_to builds
  group_by_field "status_type"
```

Use case: "Give me code review stats on a whole project (total #, total reviewed commits, total un-reviewed)"

```
#Query 1 to find un-reviewed commits
  in_project ["proj1706"]
  restrict_to commits
  has_none "reviews"
  count

  #Query 2 to find reviewed commits:
  in_project ["proj1706"]
  restrict_to commits
  has_any "reviews"
  count
```

Use case: "Given a set of artfs, give me associated build results starting 1 week ago ending 5 minutes ago, grouped into 4 hour intervals."

```
#Query:
```

```
associated_to ["artf1001",
"artf1002", "artf1003"]
associations
restrict_to builds
after 1.week.ago
before 5.minutes.ago
timeseries 4.hours
```

Use case: "Give me all binary artifact activities grouped by status"

```
#To query XDS sources like binary
artifacts, restrict_to "custom"
source type, then
  #filter to narrow the results
using tagged_as
  #Query:
  restrict_to custom
  tagged_as ["binary artifact"]
  group_by_field "status_type"
```

Code samples

Create a Java client using Apache Axis & Apache HttpComponents

Get an X-Orchestrate-Session value from TeamForge.

```
/**
  * @param ctfEndpoint
  * @param ctfUserName
  * @param ctfUserPassword
  * @return soapSessionId (X-
Orchestrate-Session)
  */
 private static String
getSoapSessionIdFromCTF(String
ctfEndpoint, String ctfUserName,
String ctfUserPassword) {
 try {
  Service service = new Service();
  Call call = (Call)
service.createCall();
  call.setTargetEndpointAddress(new
java.net.URL(ctfEndpoint));
  call.setOperationName(new
QName("http://
schema.open.collab.net/sfee50/
soap60/service", "login"));
  String soapSessionId =
(String) call.invoke(new Object[]
{ctfUserName, ctfUserPassword});
  return soapSessionId;
 } catch (Exception e) {
  System.err.println(e.toString());
  return e.toString();
  }
  }
```

Post a request to the Reporting endpoint containing the custom header and a query.

```
/**
  * @param url
```

```
 * @param token
 * @param query
 * @return queryResults
 * @throws Exception
 */
String sendPostHttpClient(String
url, String token, String query)
throws Exception {
 String queryResults;
 CloseableHttpClient client =
HttpClients.custom().build();

 HttpUriRequest request =
RequestBuilder.post().setUri(url)

.setHeader(HttpHeaders.CONTENT_TYPE,
"application/x-www-form-
urlencoded")
 .setHeader("X-Orchestrate-
Session", token)
 .setEntity(new
StringEntity(query))
 .build();

 CloseableHttpResponse response =
client.execute(request);
 queryResults =
EntityUtils.toString(response.getEntity(),
"UTF-8");
 return queryResults;
 }
```

Build your client using the above methods.

```
public class ReportingClient {

 /* Orchestrate End Point
represents the Orchestrate
instance which you will be querying
against*/
 static String orchestrateEndPoint
= "http(s)://OrchestrateHost/orc/
api/1/reporting/";

 /* teamForgeEndPoint represents
the soap endpoint for the TeamForge
instance which your Orchestrate
 * server is bound to*/
 static String teamForgeEndPoint =
"http(s)://TeamforgeHost/ce-soap60/
services/CollabNet";

 /* orcReportingQuery is your
query in String format. Each LINE
represents a different command
 * so in this example every
distinct command must be separated
by a newline character '\n'*/
 static String orcReportingQuery
= "restrict_to reviews
\ngroup_by_field \"status_type\"";
```

```
 /* TeamForge username and password
for the account which the queries
will be run under. This account
 * must have REPORTING_API
permissions on at least one project
in Orchestrate or must be a SITE
ADMIN
 * in Teamforge. If the user is NOT
a TeamForge SITE ADMIN, only those
projects for which the user has
 * REPORTING_API permissions will
be considered in the query.*/
 static String teamForgeUserName =
"userName";
 static String
teamForgeUserPassword =
"userPassword";


 public static void main(String...
args) {
  OrchestrateReportingClient
sampleClient = new
OrchestrateReportingClient();
  String xOrchestrateSession =
sampleClient.getSoapSessionIdFromCTF(teamForge
  teamForgeUserName,
teamForgeUserPassword);
  try {
   String serverResponse =
sampleClient.sendPostHttpClient(orchestrateEndP
   xOrchestrateSession,
orcReportingQuery);

System.out.println(serverResponse);
  } catch (Exception e) {

System.out.println(e.toString());
  }

  }
 }
```

TeamForge Orchestrate HTTP API
TeamForge Orchestrate HTTP API for XDS Schema Creation and Viewing

Overview

- TeamForge Orchestrate HTTP API for XDS Schema provides a means to create an XDS schema or view a schema and its versions. The API supports both data retrieval and submission.
- Authentication: requires a valid TeamForge SOAP session id, passed with the 'X-Orchestrate-Session' header field. One way to acquire this session id is to programmatically log into TeamForge using TeamForge SOAP API.
- Authorization: requires the TeamForge Orchestrate "PIPELINE CREATE" permission, or site admin permissions for creating schema. No permission required for viewing schema.

Convention

API requests for viewing and creating schemas are constructed as detailed below.

- **List all schemas**

  - **Request Type**: GET

- **Request URL**: https://<orchestrate_hostname>/api/1/schemas
- **Return Format**: Array of JSON objects. Each of the JSON object is a schema.
- List all schemas with a same schema_id

  - **Request Type**: GET
  - **Request URL**: https://<orchestrate_hostname>/api/1/schemas/SCHEMA_ID

    ☞ **Note:** SCHEMA_ID needs to be encoded (escaped).

  - **Requires Parameter**: SCHEMA_ID
  - **Return Format**: Array of one or more JSON object(s). Each of the JSON object is a schema with the given schema_id and all the versions.
- Get one schema

  - **Request Type**: GET
  - **Request URL**: https://<orchestrate_hostname>/api/1/schemas/SCHEMA_ID/VERSION

    ☞ **Note:** SCHEMA_ID needs to be encoded (escaped).

  - **Requires Parameter**: SCHEMA_ID, VERSION
  - **Return Format**: Array with one JSON object. The JSON object is a schema with the given schema_id and given version.
- Create a schema

  - **Request Type**: POST
  - **Request URL**: https://<orchestrate_hostname>/api/1/schemas
  - **Return Format**: An encoded (escaped) URL. This URL can be used to view the schema once it has been processed and saved.

TeamForge Orchestrate HTTP API for Tool Creation and Viewing

Overview

- TeamForge Orchestrate HTTP API for Tool provides a means to create an Tool or view a tool. The API supports both data retrieval and submission.
- Authentication: requires a valid TeamForge SOAP session id, passed with the 'X-Orchestrate-Session' header field. One way to acquire this session id is to programmatically log into TeamForge using TeamForge SOAP API.
- Authorization: requires the TeamForge Orchestrate "PIPELINE CREATE" permission, project admin, or site admin permissions.

☞ **Note:** PROJECT_URL_NAME is auto generated by TeamForge with project name initially. This should be found in last path of URL at project home page.

Convention

API requests for viewing and creating tools are constructed as detailed below.

- **List all tools**

  - **Request Type:**
  - **Request URL:** https://<orchestrate_hostname>/api/stable/projects/PROJECT_URL_NAME/tools
  - **Return Format:**

```
{
  "tools": [
    {
      "id": "unique_tool_id_1",
      "name": "Sample tool name One",
      "service_name": "Sample service name One",
      "project_id": "project id ",
      "active": true,
```

```
      "source_ids": [

      ]
    },
    {
      "id": "unique_tool_id_2",
      "name": "Sample tool name two",
      "service_name": "Sample service name two",
      "project_id": "project id ",
      "active": true,
      "source_ids": [
        "SourceID1",
        "SourceID2"
      ]
    }
  ]
}
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503
- **Get one tool**

  - **Request Type:** GET
  - **Request URL:** https://<orchestrate_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID
  - **Return Format:**

```
{
  "tool": {
    "id": "TOOL_ID",
    "name": "Sample tool name two",
    "service_name": "Sample service name two",
    "project_id": "project id ",
    "active": true,
    "source_ids": [
      "SourceID1",
      "SourceID2"
    ]
  }
}
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503
- **Create a tool**

  - **Request Type:** POST
  - **Request URL:** https://<orchestrate_hostname>/api/stable/projects/PROJECT_URL_NAME/tools
  - **Requires Parameter:** PROJECT_URL_NAME
  - **Request body:**

```
{
  "tool": {
    "name": "Sample tool name two",
    "source_ids": [
      "SourceID1",
      "SourceID2"
    ]
  }
```

```
  }
```

- **Return Format:**

```
{
  "tool": {
    "id": "unique_tool_id_x",
    "name": "Sample tool name two",
    "project_id": "project id ",
    "active": true,
    "source_ids": [
      "SourceID1",
      "SourceID2"
    ]
  },
  "errors": {
    "base": [
      "error msg"
    ],
    "attribute_key": [
      "error msg"
    ]
  }
}
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503, 422
- **Update a tool**

  - **Request Type:** PUT
  - **Request URL:** https://<orchestrate_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID
  - **Requires Parameter:** PROJECT_URL_NAME, TOOL_ID
  - **Request body:**

```
{
  "name": "Tool New Name",
  "source_ids": [
    "sourceID3"
  ]
}
```

- **Return Format**:

```
{
  "errors": {
    "base": [
      "error msg"
    ],
    "attribute_key": [
      "error msg"
    ]
  }
}
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503, 422
- **Activate a tool**

  - **Request Type:** PUT

- **Request URL:** https://<orchestrate_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID/ activate
- **Requires Parameter:** PROJECT_URL_NAME, TOOL_ID
- **Return Format:**

```
{
  "errors": {
    "base": [
      "error msg"
    ],
    "attribute_key": [
      "error msg"
    ]
  }
}
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503, 422

- **Deactivate a tool**

  - **Request Type:** PUT
  - **Request URL:** https://<orchestrate_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID/ deactivate
  - **Requires Parameter:** PROJECT_URL_NAME, TOOL_ID
  - **Return Format:**

```
{
  "errors": {
    "base": [
      "error msg"
    ],
    "attribute_key": [
      "error msg"
    ]
  }
}
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503, 422

TeamForge Orchestrate HTTP API for Source CRUD

Overview

- TeamForge Orchestrate HTTP API for Sources provides a means to create or view a source. The API supports both data retrieval and submission.
- Authentication: requires a valid TeamForge SOAP session id, passed with the 'X-Orchestrate-Session' header field. One way to acquire this session id is to programmatically log into TeamForge using TeamForge SOAP API.
- Authorization: requires the Orchestrate "PIPELINE CREATE" permission or TeamForge's project-admin / site-admin permissions.

☞ **Note:** PROJECT_URL_NAME is auto generated by TeamForge with project name initially. This should be the last value in the URL at project home page.

Convention

API requests for viewing and creating sources are constructed as detailed below.

- **List all sources by Project**

    - **Request Type:** GET
    - **Request URL:** https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources
    - **Return Format:**

```json
{
  "sources": [
    {
      "id": "5578cd1c0fbb6ad771000001",
      "_type": "Build",
      "tool_id": "552c3736c69d3b3dfb000004",
      "active": true,
      "display_name": "Sample_Build_Source",
      "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
      "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
      "queue_server": "amqp://rabbitmqserver.example.com:5672",
      "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
      "associated_commit_server_id": "551d8560c69d3b474b000102"
    }
  ]
}
```

    - **Return Codes**

        - **Success :** 200
        - **Error :** 404, 503

- **List all sources by Tool**

    - **Request Type:** GET
    - **Request URL:** https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/tools/TOOL_ID/sources
    - **Return Format:**

```json
{
  "sources": [
    {
      "id": "5578cd1c0fbb6ad771000001",
      "_type": "Build",
      "tool_id": "552c3736c69d3b3dfb000004",
      "active": true,
      "display_name": "Sample_Build_Source",
      "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
      "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
      "queue_server": "amqp://rabbitmqserver.example.com:5672",
      "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
      "associated_commit_server_id": "551d8560c69d3b474b000102"
    }
  ]
}
```

- **Create a source**

    - **Request Type:** POST
    - **Request URL:** https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources
    - **Request body:**

        Required: _type, display_name

```json
{
  "source": {
    "_type": "Build",
```

```
      "tool_id": "552c3736c69d3b3dfb000004",
      "display_name": "Sample_Build_Source",
      "associated_commit_server_id": "551d8560c69d3b474b000102"
    }
  }
```

- **Return Format:**

```
{
  "source": {
    "id": "5578cd1c0fbb6ad771000001",
    "_type": "Build",
    "tool_id": "552c3736c69d3b3dfb000004",
    "active": true,
    "display_name": "Sample_Build_Source",
    "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
    "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
    "queue_server": "amqp://rabbitmqserver.example.com:5672",
    "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
    "associated_commit_server_id": "551d8560c69d3b474b000102"
  },
  "errors": {
    "base": [
      "error msg"
    ],
    "attribute_key": [
      "error msg"
    ]
  }
}
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503, 422

- **Get one source**

  - **Request Type:** GET
  - **Request URL:** https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources/SOURCE_ID
  - **Return Format:**

```
{
  "source": {
    "id": "5578cd1c0fbb6ad771000001",
    "_type": "Build",
    "tool_id": "552c3736c69d3b3dfb000004",
    "active": true,
    "display_name": "Sample_Build_Source",
    "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
    "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
    "queue_server": "amqp://rabbitmqserver.example.com:5672",
    "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
    "associated_commit_server_id": "551d8560c69d3b474b000102"
  }
}
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503

- **Update one source**

- **Request Type:** PUT
- **Request URL:** https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources/SOURCE_ID
- **Request body:**

```
{
  "tool_id": "552c3736c69d3b3dfb000004",
  "display_name": "Sample_Build_Source",
  "associated_commit_server_id": "551d8560c69d3b474b000102"
}
```

- **Return Format:**

```
{
  "errors": {
    "base": [
      "error msg"
    ],
    "attribute_key": [
      "error msg"
    ]
  }
}
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503, 422
- **Activate one source**

  - **Request Type:** PUT
  - **Request URL:** https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources/SOURCE_ID/
    activate
  - **Return Format:**

```
{
  "errors": {
    "base": [
      "error msg"
    ],
    "attribute_key": [
      "error msg"
    ]
  }
}
```

  - **Return Codes**

    - **Success :** 200
    - **Error :** 404, 503, 422
- **Deactivate one source**

  - **Request Type:** PUT
  - **Request URL:** https://<ctf_hostname>/api/stable/projects/PROJECT_URL_NAME/sources/SOURCE_ID/
    deactivate
  - **Return Format:**

```
{
  "errors": {
    "base": [
      "error msg"
    ],
```

```
      "attribute_key": [
        "error msg"
      ]
    }
  }
```

- **Return Codes**

  - **Success :** 200
  - **Error :** 404, 503, 422

Return Types

The source objects have a set of common fields, while some of the fields depend on the _type of the source. The examples below show the type of source and the expected structure of the returned objects.

- **Source Type:** Build

```
{
      "source": {
        "id": "5578cd1c0fbb6ad771000001",
        "_type": "Build",
        "tool_id": "552c3736c69d3b3dfb000004",
        "active": true,
        "display_name": "Sample_Build_Source",
        "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
        "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
        "queue_server": "amqp://rabbitmqserver.example.com:5672",
        "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
        "associated_commit_server_id": "551d8560c69d3b474b000102"
      }
    }
```

- **Source Type:** Commit

```
{
      "source": {
        "id": "5578cd1c0fbb6ad771000001",
        "_type": "Commit",
        "tool_id": "552c3736c69d3b3dfb000004",
        "active": true,
        "display_name": "Sample_ExternalRepository",
        "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
        "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
        "queue_server": "amqp://rabbitmqserver.example.com:5672",
        "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
        "repository_uri": "ssh://sample_gitserver/sample_project"
      }
    }
```

- **Source Type:** CommitCTF

```
{
      "source": {
        "id": "5578cd1c0fbb6ad771000001",
        "_type": "CommitCTF",
        "tool_id": "552c3736c69d3b3dfb000004",
        "active": true,
        "ctf_id": "cmmt1234",
        "display_name": "Sample_ExternalRepository",
        "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
```

```
        "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
        "queue_server": "amqp://rabbitmqserver.example.com:5672",
        "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
        "repository_uri": "ssh://sample_gitserver/sample_project"
      }
    }
```

- **Source Type:** XDS

```
{
      "source": {
        "id": "5578cd1c0fbb6ad771000001",
        "_type": "XDS",
        "tool_id": "552c3736c69d3b3dfb000004",
        "active": true,
        "display_name": "Sample Custom Activity Name One",
        "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
        "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
        "queue_server": "amqp://rabbitmqserver.example.com:5672",
        "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
        "associated_activity_source_id": "55429d045667ba223e000010",
        "activity_source_tag_names": ["Exclusive", "Limited", "Sample"],
        "background_color_value": "#61c0e5",
        "icon_uri": "/sample_icons/custom_source/
Sample_Icon_White_12.png"
        "custom_schema_store_id": object.custom_schema_store_id
      }
    }
```

- **Source Type:** Review

```
{
      "source": {
        "id": "5578cd1c0fbb6ad771000001",
        "_type": "Review",
        "tool_id": "552c3736c69d3b3dfb000004",
        "active": true,
        "display_name": "Sample Review server Name One",
        "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
        "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
        "queue_server": "amqp://rabbitmqserver.example.com:5672",
        "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
        "commit_association_prefix": "Review"
      }
    }
```

- **Source Type:** WorkItemSource

```
{
      "source": {
        "id": "5578cd1c0fbb6ad771000001",
        "_type": "Review",
        "tool_id": "552c3736c69d3b3dfb000004",
        "active": true,
        "display_name": "Sample Work Item Name One",
        "queue_username": "4d4eaf60-f1f9-0132-4687-185aeb8b25b8",
        "queue_password":
 "d70bc954b64e368f1f650bfeaba2e2ff6e5525774ae42f9a959f4ccf2314e9cb",
        "queue_server": "amqp://rabbitmqserver.example.com:5672",
        "source_association_key": "29bf0c90-3b40-0130-ae2d-dddd5893",
        "associated_commit_server_id": "551d8560c69d3b474b000102"
```

```
                    }
                }
```

Validations

The following constitute the validations that are performed for the API request-response parameters.

- **_type**

  - type: string
  - present for all sources
  - must be one of: *Commit, CommitCTF, WorkItem, CustomActivity, Review, Build, XDS*

- **active**

  - type: boolean
  - present in all sources
  - must be true or false
  - default value: true

- **associated_activity_source_id**

  - type: string
  - attribute in XDS, need not be present
  - references an existing activity source if present

- **commit_association_prefix**

  - type: string
  - attribute in Review, must be present
  - must be unique
  - key length must be less than 101

- **activity_source_tag_names**

  - type: array with string elements
  - attribute in XDS, need not be present

- **background_color_value**

  - type: string
  - attribute of XDS, must be present
  - must be a valid RGB value or a HTML5 color name

- **associated_commit_server_id**

  - type: string
  - attribute in Build and WorkItem, need not be present
  - must be associated with a valid commit server if present

- **custom_schema_source_id**

  - type: string
  - attribute in XDS, need not be present
  - must be associated to an existing custom_schema_source if present

- **display_type**

  - type: string
  - must be present in all sources
  - maximum length less than 101

- **icon_uri**

  - type: string
  - attribute of XDS, need not be present
  - has the default value: "/assets/icons/custom_source/PNG_White/CN_Icon_White_12.png"

- must be a valid uri if present
- **repository_uri**

  - type: string
  - must be present in Commit and CommitCTF
  - must be a valid uri if present
  - must be a valid protocol (*http, https, ssh, git*) if present

- **ctf_id**

  - type: string
  - must be present in CommitCTF
  - must be a valid ctf id if present

- **source_association_key**

  - type: string in all sources
  - must be present

- **tool_id**

  - type: string
  - need not be present or present in any source
  - must be associated with a valid tool if present

Uninstall

Uninstall TeamForge Orchestrate by removing the various components

Follow the manual steps below to disassociate TeamForge Orchestrate from TeamForge, and then ultimately uninstall the components.

☞ **Important:** It is important that each step be followed in sequential order. The Orchestrate App server must remain up until the IAF application is removed.

Remove Orchestrate adapters from TeamForge

1. In TeamForge, as a site admin user navigate to **Admin** > **System Tools** > **Customizations**
2. Select the `orchestrate-commit-adapter-[version].jar` and (if installed) `orchestrate-tracker-adapter-[version].jar` check boxes
3. Click Delete

Delete the Orchestrate IAF application TeamForge

4. In TeamForge, as a site admin user navigate to **Admin** > **Integrated Apps**
5. Select Orchestrate application
6. Click Delete

Delete the Orchestrate site-wide linked app in TeamForge

7. In TeamForge, as a site admin user navigate to **Admin** > **Integrations** > **Site-wide Linked Applications**
8. Select Orchestrate application
9. Click Delete

Remove TeamForge Orchestrate server side components

10. It is now safe to shutdown Orchestrate services

    Shutdown each service in the *System Administration* on page 657

11. Remove components from the App server

    ```
    yum remove collabnet-orchestrate-core collabnet-redis
    ```

    This command will leave your TeamForge Orchestrate configuration and log files intact. If you wish to remove those as well, run the following command:

    ```
    rm -rf /opt/collabnet/orchestrate /var/log/orchestrate /var/log/collabnet-*
    ```

12. Remove components from the DB server

```
yum remove collabnet-mongodb
```

This command will leave your database configuration and log files intact. If you wish to remove those as well, run the following command:

```
rm -rf /opt/collabnet/mongodb /var/log/collabnet-mongodb
```

13. Remove components from the MQ server

```
yum remove collabnet-erlang collabnet-rabbitmq
```

This command will leave your message queue configuration and log files intact. If you wish to remove those as well, run the following command:

```
rm -rf /opt/collabnet/rabbitmq /var/log/collabnet-rabbitmq
```

Remove adapters

14. Uninstall adapters from integrated services

For most adapters, reverse installation steps to remove adapters. For plug-in based adapters, uninstall using the service's administrative user interface.

Set up Git

TeamForge 8.2 supports Git integration based on Gerrit 2.11.x (version 8.5.x of the integration).

TeamForge—Git integration: an overview

TeamForge 8.2 supports an integration with the Git distributed version control tool powered by Gerrit.

Although Git is the world's leading distributed version control system, the enterprise has been slow and tentative in its adoption. Concerned with security breaches, compliance violations and lack of governance, many organizations have chosen to take a "wait and see" approach. With TeamForge, Git is ready for the enterprise. TeamForge lets you realize all the benefits of Git while ensuring the security, governance and manageability your business demands. With TeamForge, you can even manage Git and Subversion together, within each individual project.

Gerrit is an open source code review system designed to work with Git. Gerrit supports various access control mechanisms. The TeamForge Git integration uses Gerrit as a vehicle to bring TeamForge project roles and permissions into Git.

Highlights

- Easy Git repository management from TeamForge using TeamForge's role based access control.
- Authentication using SSH keys stored in TeamForge and `http` using Gerrit's `http` passwords.

- TeamForge artifact association with Git commits.
- Git source code browsing and code search.
- Single sign-on between TeamForge and the Gerrit web console.
- Full support of Gerrit's APIs to connect with Continuous Integration systems like Jenkins.

Set up the TeamForge Git integration

The TeamForge-Git Integration 8.5.x is based on Gerrit 2.11.x and is supported for TeamForge 7.1 and later on Red Hat and CentOS version 6.7 or later. With TeamForge 8.2, you can install the Git integration as a part of installing TeamForge.

You can install Git integration on a separate server and other services on the main application server. For more information, see:

- *Install Git on a separate server* on page 431
- *TeamForge Git integration*

Set up Code Search for TeamForge-Git integration

To use TeamForge Code Search functionality for Git, manually grant the TeamForge Code Search user permissions to access all Git repositories.

1. You need the root user's public key for SSH authentication on the Code Search server. Do the following:
   a) On the Code Search server, check whether the key is present at `/root/.ssh/id_rsa.pub`. If not, generate it by running the `ssh-keygen` command.
   b) Copy it to a temporary location (`/tmp`) on the TeamForge application server.
   c) Run the *set_auth_key.py* script for the `scmviewer` user on the TeamForge application server.

   ```
   cd /opt/collabnet/teamforge/runtime/scripts/codesearch/

   ./set_auth_key.py --authkey-file=/tmp/id_rsa.pub
   ```
2. Sync the `scmviewer` user to Gerrit by running the following command in a shell on the host where you installed the Git integration.

   ```
   curl http://localhost:9081/api/gerrit/users/scmviewer/sshkeys
   ```
3. Log into the Gerrit console as the 'scmadmin' user and create an internal Gerrit group, for example, 'CodeSight Group'.
   a) Select **People** > **Create New Group** from the menu.
      The **Create Group** page appears.
   b) Type a name for the group, for example, 'CodeSight Group', and click **Create Group**.
      The group is created.
4. Add the `scmviewer` user to the group.

   Type `Code Search <scmviewer@localhost>` user name in the **Members** text box and click **Add**.
5. Grant read access to the group.
   a) Select **Projects** > **List** > **All Projects**.
   b) Select **Access** and click **Edit**.
   c) Click **Add Group** in the **Reference: refs/*** > **Read** section, type the newly created group name, for example, 'CodeSight Group', and click **Add.**
   d) Click **Save Changes**.
6. Log out from Gerrit.
7. Restart the Code Search server.

   ```
   /etc/init.d/collabnet restart tomcatcs
   ```

Set up Review Board

Review Board is a popular code review tool available with TeamForge 8.2 as a fully-integrated add-on.

☞ **Important:** TeamForge 8.2 supports Review Board 2.0.11 on RHEL/CentOS 7.2.

Install Review Board

You must install the Review Board before you can make it available as an integrated application to project managers on your TeamForge site.

☞ **Important:**

- Install Review Board on the TeamForge Application Server and it needs PostgreSQL database.
- To install Review Board successfully, ensure that other repositories similar to EPEL (Extra Packages for Enterprise Linux) are disabled apart from the CollabNet and Operating System repositories.

- This procedure is for those who are installing the Review Board for the first time.
- In this scenario, both TeamForge and Review Board use PostgreSQL.
- TeamForge 8.2 supports Review Board 2.0.11 on RHEL/CentOS 6.7 and 7.2.

Log on to the server as root user always

Do this on the main TeamForge Application Server. We'll call this `my.app.box`

1. Download the Review Board installer zip file from *http://collab.net/downloads/integrations#tab-1* and save it in the `/opt/collabnet` folder

   ☞ **Note:** Download the `RBInstaller-1.8.0.0.12.zip` file.

2. Unzip the installer zip file.

   ```
   cd /opt/collabnet
   unzip RBInstaller-1.8.0.0.12.zip
   ```

3. Modify these values in the *`install.conf`* file to suit your installation environment.

   ```
   vi /opt/collabnet/RBInstaller-1.8.0.0.12/installer/install.conf
   ```

Option	Description
rb_dir=/opt/collabnet/reviewboard	The path of the directory where the Review Board files and libraries are installed.
rb_data_dir=/opt/collabnet/reviewboard/data	The path of the directory where Review Board's database file, review request files and attachments are stored.
domain=<domain name or host name>	The Review Board site information. For example, cu064.cloud.maa.collab.net.
rb_database_type=postgresql	The Review Board database type.
rb_database_host=<reviewboard_db_hostname>	The Review Board database host name.
rb_database_port=<reviewboard_db_port>	The Review Board database port.
ctf_base_url=https://myapp.collab.net	The absolute URL of the TeamForge site that you want to associate to.
ctf_site_var_dir=/opt/collabnet/teamforge/var	The location of the rbctfevents.jar file.

4. Start the TeamForge application.

   ```
   /etc/init.d/collabnet start all
   ```

5. Run the following grep commands to get the value of HTTPD_USER, HTTPD_GROUP and HOME_DIR_BASE.

   ```
   grep HTTPD_USER= /opt/collabnet/teamforge/runtime/conf/runtime-
   options.conf
   grep HTTPD_GROUP= /opt/collabnet/teamforge/runtime/conf/runtime-
   options.conf
   ```

```
grep HOME_DIR_BASE= /opt/collabnet/teamforge/runtime/conf/runtime-
options.conf
chown -R <HTTPD_USER>:<HTTPD_GROUP> <HOME_DIR_BASE>/<HTTPD_USER>
```

6. Run the `install.py` script available in the Review Board installer directory.

```
cd /opt/collabnet/RBInstaller-1.8.0.0.12
python ./install.py -i -r --auth-scmuser
```

7. Perform this step on the TeamForge PostgreSQL Database Server on sites with database running on a separate server.

a) Create the Review Board database and user name.

```
su - postgres
createuser -P -S --createdb --no-createrole ctfrbuser
createdb -E UTF8 -O ctfrbuser ctfrbdb
exit
```

8. Set up the initial Review Board data. Run the `bootstrap-data.py` script available in the Review Board installer directory.

```
cd /opt/collabnet/RBInstaller-1.8.0.0.12
python ./bootstrap-data.py
```

9. Run the grep command to get the value of httpd_user and httpd_group.

```
grep httpd_user /etc/reviewboard.properties
grep httpd_group /etc/reviewboard.properties
chown -R <httpd_user>:<httpd_group> /opt/collabnet/reviewboard/data
```

10. Restart the TeamForge application.

```
/etc/init.d/collabnet stop all
```

```
/etc/init.d/collabnet start all
```

11. If SCM is installed on a separate box, run the following script to authenticate a scmviewer user against a TeamForge Subversion repository for creating a new review request.

```
python ./svn-auth.py --repo-path=https://<scm_domain>/svn/repos/
<repo_dir_name>
```

You should now have a Review Board instance ready to work with TeamForge. The installer has created two configuration files: `installer/conf/rb-application.xml` and `installer/conf/rb-deploy.xml`. See *Integrate Review Board with your TeamForge site* on page 724 for more information.

☞ **Note:** Run the following scripts available in the Review Board installer directory.

```
cd /opt/collabnet/RBInstaller-1.8.0.0.12
```

- To recreate the runtime, run the `python ./install.py -r` command.
- To update the revised password for a scmviewer account, run the `python ./install.py --auth-scmuser` command.
- To change the scmviewer password, see *SCM_USER_ENCRYPTED_PASSWORD* on page 811.
- To uninstall Review Board, run the `python ./install.py -u` command.

Upgrade Review Board

Use these instructions to upgrade Review Board to a latest build.

- TeamForge 8.2 supports Review Board 2.0.11 on RHEL/CentOS 6.7 and 7.2.
- This procedure is for those who have Review Board already and are upgrading Review Board to a latest build on RHEL/CentOS 6.7 or 7.2.
- You may choose to upgrade Review Board on the same server or on a new server.
- In this scenario, both TeamForge and Review Board use PostgreSQL.

Log on to the server as root user always

Do this on the main TeamForge Application Server. We'll call this `my.app.box`.

1. Download the Review Board installer zip file from *http://collab.net/downloads/integrations#tab-1* and save it in the `/opt/collabnet` folder

2. Uninstall the existing Review Board application if you are upgrading on the same server.

   ```
   cd /opt/collabnet/RBInstaller-1.7.0.0.7
   python ./install.py -u
   ```

3. Back up your Review Board data directory if you are upgrading TeamForge and Review Board on a new server. The Review Board database is backed up already when you have upgraded TeamForge. So, it is not necessary to take a back up of the Review Board database again.

   a) Back up the Review Board data directory.

   ```
   cd /opt/collabnet
   tar -zcvf /tmp/reviewboard_data.tgz reviewboard
   ```

   b) Copy the `/tmp/reviewboard_data.tgz` file to the `/tmp` directory of the new server.

4. Unzip the `RBInstaller-1.8.0.0.12.zip` file.

   ```
   cd /opt/collabnet
   unzip RBInstaller-1.8.0.0.12.zip
   ```

5. Modify these values in the *`install.conf`* file to suit your installation environment.

   ```
   vi /opt/collabnet/RBInstaller-1.8.0.0.12/installer/install.conf
   ```

Option	Description
rb_dir=/opt/collabnet/reviewboard	The path of the directory where the Review Board files and libraries are installed.
rb_data_dir=/opt/collabnet/reviewboard/data	The path of the directory where Review Board's database file, review request files and attachments are stored.
domain=<domain name or host name>	The Review Board site information. For example, cu064.cloud.maa.collab.net.
rb_database_type=postgresql	The Review Board database type.
rb_database_host=<reviewboard_db_hostname>	The Review Board database host name.
rb_database_port=<reviewboard_db_port>	The Review Board database port.
ctf_base_url=https://myapp.collab.net	The absolute URL of the TeamForge site that you want to associate to.
ctf_site_var_dir=/opt/collabnet/teamforge/var	The location of the rbctfevents.jar file.

6. Start the TeamForge application.

   ```
   /etc/init.d/collabnet start all
   ```

7. Run the following grep commands to get the value of HTTPD_USER, HTTPD_GROUP and HOME_DIR_BASE.

   ```
   grep HTTPD_USER= /opt/collabnet/teamforge/runtime/conf/runtime-
   options.conf
   grep HTTPD_GROUP= /opt/collabnet/teamforge/runtime/conf/runtime-
   options.conf
   grep HOME_DIR_BASE= /opt/collabnet/teamforge/runtime/conf/runtime-
   options.conf
   sudo chown -R <HTTPD_USER>:<HTTPD_GROUP> <HOME_DIR_BASE>/<HTTPD_USER>
   ```

8. Restore the Review Board data if you are upgrading TeamForge and Review Board on a new server.

> ☞ **Note:** Ensure that you have already copied the backup of the Review Board data directory to the `/tmp` directory of the new server.

a) Restore the Review Board data directory.

```
cd /opt/collabnet
tar -zxvf /tmp/reviewboard_data.tgz
```

9. Run the `install.py` script available in the Review Board installer directory.

```
cd /opt/collabnet/RBInstaller-1.8.0.0.12
python ./install.py -i -r --auth-scmuser
```

10. Run the `migrate.py` script available in the Review Board installer directory.

```
cd /opt/collabnet/RBInstaller-1.8.0.0.12
python ./migrate.py
```

11. Restart the TeamForge application.

```
/etc/init.d/collabnet stop all
```

```
/etc/init.d/collabnet start all
```

12. If SCM is installed on a separate box, run the following script to authenticate a scmviewer user against a TeamForge Subversion repository for creating a new review request.

```
python ./svn-auth.py --repo-path=https://<scm_domain>/svn/repos/
<repo_dir_name>
```

Integrate Review Board with your TeamForge site

When the Review Board application has been installed on your site, you can make it available for projects on your TeamForge site.

Review Board must be installed and configured before you can integrate it with your TeamForge site. See *Install Review Board* on page 721.

> ☞ **Important:** You must not create, edit or delete new user accounts while installing the Review Board application and integrating it with TeamForge.

When you have installed Review Board, projects on your site can add Review Board to their set of collaboration tools.

1. Log into TeamForge as an administrator.

2. Go to **My Workspace > Admin**.

3. Click **INTEGRATED APPS** from the **Projects** menu.

4. Click **Create**.

5. Use the **Browse** window to select the application and deployment configuration files that enable the Review Board application to work as a part of TeamForge:

- `rb-application.xml` (Application configuration file): Contains the text strings for the Review Board user interface. This file can be found in `/opt/collabnet/RBInstaller-1.8.0.0.12/installer/conf/`.
- `rb-deploy.xml` (Deployment configuration file): Contains the data that Review Board needs to interact with TeamForge. This file can be found in `/opt/collabnet/RBInstaller-1.8.0.0.12/installer/conf/`.

Click **Next**.

6. On the **Preview** screen, review the parameters you set in the configuration files.

> ☞ **Note:** You may have to revise one or more values to ensure they are valid.

7. Click **Save**.

The Review Board application is now available for all projects on your site. Project administrators can refer to the topic, *Add Review Board to a TeamForge project* on page 725, for instructions on adding Review Board to their own project toolbars.

☞ **Note:** You may need to adjust your site's look and feel to support your integrated application. See *the site admin help* for details.

Add Review Board to a TeamForge project

When the site administrator has made the Review Board application available, project administrators can add it as one of their project tools.

When you add Review Board to your project, it works just like the other TeamForge tools, authorization, authentication, go-urls, association, linkification and source code management support.

For this example, we'll call your project "testproject" and we'll assume you have Project Admin rights in that project.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tools** to see the list of integrated applications available in the site.
3. Click **Add Tool**.
4. Select **Review Board** from the list of tools displayed.

   Set the values that make sense for your TeamForge installation, then click **Save**.

Option	Description
Prefix	Specify a unique alphanumeric string that will identify this tool throughout the site. For example, suppose you set your prefix to ZZ. Entering ZZ_1 in the **Jump to ID** redirect you to review request id 1 of the "testproject" project. Another may also add Review Board as a tool, with a prefix of YY.Jumping to YY_1 will redirect to review request id 1 of that project. ☞ **Note:** The tool prefix cannot be changed after you have set it.
Exclude Repositories	Use a comma to separate the directory names of the repositories that you want to exclude from syncing with Review Board. To include all repositories, enter "None". ☞ **Important:** You must sync repositories with Review Board (select **Synchronize Repositories** check box) to have the repositories listed in this **Exclude Repositories** text box excluded.
Synchronize Repositories	Synchronize TeamForge Subversion repositories with Review Board. ☞ **Important:** Select this check box every time you want to sync repositories to Review Board. When you add Review Board to your project with this check box selected, the SVN repositories are synced once and the check box is cleared. For any new repositories you create, you must edit the Review Board (**Project Admin** > **Project Toolbar** > **Edit Integrated Application**) and sync new repositories with Review Board by selecting this check box.

If a **Review Board** button appears along with the prefix as a tooltip when you mouse over the button, your job is done.

TeamForge - Review Board integration: FAQs
Frequently asked questions about TeamForge - Review Board integration.

TeamForge - Review Board integration: Install FAQs
Questions about TeamForge - Review Board integration.

What are the software requirements for installing Review Board as an integrated application in TeamForge 8.2?	Review Board can run on: • RHEL/CentOS 7.2 In addition, Review Board needs PostgreSQL 9.3.6. See *Software requirements for CollabNet TeamForge 8.2* on page 748 topic for more information.
Can I install TeamForge and Review Board on separate servers?	No. Review Board must be installed on the same server where TeamForge runs.

TeamForge - Review Board integration: General usage FAQ
General usage questions about the TeamForge - Review Board integration.

Which version of Review Board does TeamForge 8.2 support?	TeamForge 8.2 supports Review Board 2.0.11.
Which repositories does Review Board support?	Review Board supports only Subversion repository in TeamForge 8.2.
How do I manage users in Review Board?	You can manage Review Board users from TeamForge. Whenever you create or edit users in TeamForge, they are synchronized automatically in Review Board.
Can I specify 'RB' as a prefix in my project?	No. You cannot specify 'RB' as a prefix in your project. The prefix for Review Board must be unique for every project.
Is it possible to grant TeamForge specific-permissions as part of the system generated Review Board administrator?	No. It is not possible to grant TeamForge specific-permissions as part of the system generated Review Board administrator (integrated application specific role).
Can I use the Review Board 'Search' feature after integrating Review Board with TeamForge?	No. TeamForge does not support the 'Search' feature of Review Board.
What are the additional features available in Review Board after you integrate it with TeamForge?	Review Board uses some of the TeamForge features like object IDs, links, GO URLs, and SVN integration and associations. For more information, see *How does an integrated application interact with other TeamForge tools?* on page 994
What are the other TeamForge features which Review Board does not support after you integrate Review Board with TeamForge?	Global search, page component, recent history and project template features of TeamForge are not supported in Review Board.
Where can I find the documentation for Review Board?	You can find the documentation for Review Board *here*.

How do I control user access for Review Board?
You can control user access for Review Board based on user roles.

The following roles are defined in the XML application configuration file in TeamForge:

Submit / Edit / View	Using this permission, you can submit new review request, modify your existing review requests, and view the existing requests in Review Board.
Change Status as another user	Using this permission, you can change the review request status of another user.
Submit as another user	Using this permission, you can submit a review request as another user through the post-review command line tool.
Edit as another user	Using this permission, you can modify a review request of another user through the user interface or the post-review command line tool.
Admin staff	When you have this permission along with the Submit/Edit/View permission through a site-wide role, you can access Review Board administration user interfaces.

☞ **Note:** Site administrators have all permissions on all projects on the site, by default. Site administrators grant these permissions to other users based on their roles.

Set up Black Duck Code Sight

TeamForge 8.2 supports the Black Duck Code Sight source code search engine.

To install Black Duck Code Sight, see the instructions specific to your platform.

Black Duck Code Sight

With the integration of Black Duck Code Sight, TeamForge 8.2 enables developers to easily find code snippets in source code repositories.

You can search repositories in all TeamForge projects or specific ones from your "My Projects" list. Besides code snippets, you can search for things such as file names, and classes or methods in C#, Java, and Python files. For more information on what you can search for, how search results are ranked, and advanced searches, please have a look at the *Black Duck Code Sight User Guide.*

You have the ability to control whether source code is included in search results. For example, you may want to exclude source code from searches that have sensitive information. You may also want to omit some repositories, such as the branding and publishing repositories, from search results. The **Available in Search Results** setting in **Create Repository** or **Edit Repository** page allows you to configure this. By default, this option is selected for all repositories, except for the **branding** repository in the **look** project and for **Publishing** repositories.

TeamForge site administrators can check the status of the Code Sight server in the **Admin** > **System Tools** > **Server Status** page.

☞ **Note:** Code search can be done only through global search.

Install TeamForge 8.2: Black Duck Code Sight on a separate server on RHEL/CentOS

In this option, we install Black Duck Code Sight on a separate server and other services on the main application server.

☞ **Important:** See *Plan your installation* on page 370 and *Software requirements for CollabNet TeamForge 8.2* on page 748 before you proceed.

☞ **Note:** For the ETL service to run as expected in a distributed TeamForge installation, all servers must have the same time zone.

In this option, the following services run on the application server (we call this my.app.host).

• TeamForge Application Server
• Database Server (Operational DB and Reports DB)

- ETL Server
- GIT Integration Server
- SCM Integration Server (Subversion and CVS)
- Search Server (Indexer).

☞ **Important:** Black Duck Code Sight is not supported on RHEL/CentOS 7.2. CollabNet recommends that you install Black Duck Code Sight on a separate RHEL/CentOS 6.7 server.

The following service runs on the code sight server (we call this my.codesight.host).

- Black Duck Code Sight Server

Log on to the server as root user always.

Do this on the main TeamForge application server. We'll call this my.app.host.

1. Install Red Hat Enterprise Linux/CentOS 7.2 and log in as root.

   - Red Hat Enterprise Linux/CentOS 6.7 is also supported. See *Software requirements for CollabNet TeamForge 8.2* on page 748.
   - The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
   - See *the Red Hat installation guide* for help.

   ☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Create a file, `/etc/security/limits.d/teamforge.conf,` and make sure it has the following "nproc" and "nofile" settings.

   ```
   vim /etc/security/limits.d/teamforge.conf
   sf-admin soft nproc 10240
   sf-admin hard nproc 10240
   sf-admin hard nofile 65536
   sf-admin soft nofile 65536
   ```
   Save the file and reboot the server.

3. Check your basic networking setup.

   See *Set up networking for your TeamForge server* on page 372 for details.

4. If the TeamForge server has SELinux enabled, run it in 'Permissive' mode temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```
   b) If the output of the `getenforce` command is "Permissive", continue with the next step. If not, run the following command to bring it to 'Permissive' mode.

   ```
   setenforce 0
   ```

   See *Set up SELinux* on page 580 to have TeamForge run in SELinux mode after completing the installation or upgrade.

5. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.
   b) Install the repository package.

   ```
   yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
   ```
   c) Refresh your repository cache.

   ```
   yum clean all
   ```

   TeamForge installation repository configuration for sites without internet access

a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in /tmp.

- Red Hat Enterprise Linux/CentOS 7.2 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel7.x86_64.rpm**

  ☞ **Note:** In addition to the above CentOS 7.2 64 bit RPM package, you must get the following CentOS 7.2 compatibility RPM, which is required for TeamForge 8.2 disconnected media installation on CentOS 7.2 profile: **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm**.

b) Unpack the disconnected installation package.

```
rpm -ivh <package-name>
```

c) Unpack the **compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm** package if you are installing TeamForge 8.2 on CentOS 7.2.

```
rpm -ivh compat-ctf-dc-media-1.0-1.el7.centos.noarch.rpm
```

d) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

e) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

f) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

6. Install the following application packages.

a) TeamForge: To install the TeamForge application packages run the following command:

```
yum install teamforge
```

b) GIT: To install the GIT packages run the following command.

```
yum install teamforge-git
```

7. Set up your site's master configuration file.

```
vi /opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options.conf
```

a) Configure the HOST token.

-
  ```
  HOST_localhost=app database datamart etl indexer subversion cvs
  ```

-
  ```
  DOMAIN_localhost=my.app.domain.com
  ```

-
  ```
  HOST_my.codesight.domain.com=codesearch
  ```

b) Add 'gerrit' to the HOST_localhost token if you are installing Git.

```
HOST_localhost=app database datamart etl indexer subversion cvs gerrit
```

c) Add 'binary' to the HOST_localhost token if you are installing Nexus.

```
HOST_localhost=app database datamart etl indexer subversion cvs
 gerrit binary
```

d) Configure the database and datamart settings.

☞ **Note:** For more information about configuring variables, see *site-options.conf* on page 775

-
  ```
  DATABASE_TYPE=postgresql
  ```

-
  ```
  DATABASE_USERNAME=ctfuser
  ```

-
  ```
  DATABASE_NAME=ctfdb
  ```

-
  ```
  DATABASE_READ_ONLY_USER=ctfrouser
  ```

-
  ```
  REPORTS_DATABASE_USERNAME=ctfrptuser
  ```

-
  ```
  REPORTS_DATABASE_NAME=ctfrptdb
  ```

-
  ```
  REPORTS_DATABASE_READ_ONLY_USER=ctfrptrouser
  ```

-
  ```
  REPORTS_DATABASE_MAX_POOL_SIZE=30
  ```

☞ **Note:** The database name and username values are arbitrary alphanumeric strings.

e) TeamForge 7.1 and later support automatic password creation. See *AUTO_DATA* on page 777 for more information.

f) **Password obfuscation**: The password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

☞ **Restriction:** The password-related tokens cannot contain the following characters in the `site-options.conf` file: $<>/\'"`

- To disable password obfuscation, set *OBFUSCATION_ENABLED=false*.
- To configure the obfuscation key, set *OBFUSCATION_KEY=<Any AlphaNumeric value with length >= 8 bytes>*. The default value of *OBFUSCATION_KEY* token is XSJt43wN.

g) Turn on the SSL for your site by editing the relevant variables in the `site-options.conf` file. To generate the SSL certificates, see *Generate SSL certificates* on page 582.

```
SSL=on
```

```
SSL_CERT_FILE
```

```
SSL_KEY_FILE
```

```
SSL_CHAIN_FILE
```

☞ **Note:** The SSL_CERT_FILE and SSL_KEY_FILE tokens need an absolute path. The SSL_CHAIN_FILE token is optional.

h) If the token *REQUIRE_PASSWORD_SECURITY* is enabled, then set a value for the token, *PASSWORD_CONTROL_EFFECTIVE_DATE*.

⚠ **Caution:** The Password Control Kit (PCK) disables, deletes or expires user accounts that don't meet the password security requirements starting from the date set for the *PASSWORD_CONTROL_EFFECTIVE_DATE* token. If a date is not set, the PCK disables, deletes or expires user accounts immediately. See *PASSWORD_CONTROL_EFFECTIVE_DATE* for more information.

i) Include the *SCM_DEFAULT_SHARED_SECRET* token in the `site-options.conf` file of the primary TeamForge server and provide it with a value of 16-24 characters.

☞ **Important:** Remember to use the same key in the external SCM integration server also.

j) If the token *REQUIRE_RANDOM_ADMIN_PASSWORD* is already set to true, then set the token *ADMIN_EMAIL* with a valid email address.

```
ADMIN_EMAIL=root@{__APPLICATION_HOST__}
```

k) If you have LDAP set up for external authentication, you must set the "*REQUIRE_USER_PASSWORD_CHANGE*" site options token to false.

l) Ensure to set the token *DEDICATED_INSTALL=true*. This makes the installation process very simple as the TeamForge installer takes care of configuring the Apache and PostgreSQL automatically.

m) Make sure that the following tokens have a value if ETL is enabled.

```
SOAP_ANONYMOUS_SHARED_SECRET
```

```
ETL_SOAP_SHARED_SECRET
```

n) Configure Black Duck Code Sight tokens. See *Black Duck Code Sight site-option tokens*.

o) To enable the history protection feature of TeamForge Git integration, set the **GERRIT_FORCE_HISTORY_PROTECTION=true**. For more information, see *GERRIT_FORCE_HISTORY_PROTECTION* on page 789.

p) Ensure to set the token *SELINUX_SETUP=true*.

q) Make sure the PostgreSQL tokens in the `site-options.conf` file are set as recommended in the following topic: *What are the right PostgreSQL settings for my site?* on page 909

r) Save the `site-options.conf` file.

8. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do this on the Black Duck Code Sight Server. We'll call this my.codesight.host.

9. Install Red Hat Enterprise Linux/CentOS 6.7 and log in as root.

- The host must be registered with the Red Hat Network if you are using Red Hat Enterprise Linux. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
- See *the Red Hat installation guide* for help.

☞ **Important:** Don't customize your installation. Select only the default packages list.

10. If the TeamForge server has SELinux enabled, disable it temporarily while installing or upgrading TeamForge.

   a) Verify if SELinux is running in enforcing mode.

   ```
   getenforce
   ```

   b) If the output of the `getenforce` command is either "Disabled" or "Permissive", SELinux is already disabled.

   c) If not disabled, run the following command to disable SELinux.

   ```
   setenforce 0
   ```

11. Configure your TeamForge installation repository.

   TeamForge installation repository configuration for sites with internet access

   a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

   b) Install the repository package.

   ```
   yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
   ```

   c) Refresh your repository cache.

   ```
   yum clean all
   ```

   TeamForge installation repository configuration for sites without internet access

   a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

   - Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: **CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm**

   b) Unpack the disconnected installation package.

   ```
   rpm -ivh <package-name>
   ```

   c) ☞ **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

   Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

   In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

   ```
   cd /media/
   mkdir cdrom
   mount /dev/cdrom ./cdrom/
   ```

   If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

   d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

   ```
   vi /etc/yum.repos.d/cdrom.repo
   ```

   Here's a sample yum configuration file.

   ```
   [RHEL-CDROM]
   name=RHEL CDRom
   baseurl=file:///media/cdrom/Server/
   gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
   enabled=1
   gpgcheck=0
   ```

   e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

12. Run the following command to install the Black Duck Code Sight packages.

```
yum install teamforge-codesearch
```

13. Copy the `site-options.conf` file from the application server to the `/opt/collabnet/teamforge-installer/8.2.0.0/conf` directory of the Black Duck Code Sight server.

14. Modify the host token settings on the `site-options.conf` file.

> ☞ **Important:** If you choose not to use the application server's `site-options.conf` file, then don't forget to copy the value of AUTO_DATA token from the application server.

- ```
  HOST_my.codesight.host=codesearch
  ```

- ```
  DOMAIN_my.codesight.host=my.codesight.domain.com
  ```

- ```
  HOST_my.app.domain.com=app database datamart etl indexer subversion cvs
   gerrit
  ```

15. Recreate the runtime environment.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

Do the following on the application server - my.app.host

16. Set up the initial site data (bootstrap).

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./bootstrap-data.sh
```

17. Start TeamForge.

```
/etc/init.d/collabnet start
```

> ☞ **Note:** Gerrit is configured as part of the post installation tasks included in the `post-install.py` script. As a result, Gerrit's status would be `Not configured` during startup, which you can safely ignore.

18. ☞ **Important:** If the token *REQUIRE_USER_PASSWORD_CHANGE* is set to true, login to TeamForge user interface, change the admin password and then run the `post-install.py` script.

Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

> ☞ **Note:** In case you face any GIT-related issues while running the `post-install.py` script, see *Post install fails for GIT. What should I do?* on page 921.

19. Revoke the user permissions of the database and datamart users.

```
/opt/collabnet/teamforge/runtime/scripts/revoke-superuser-permission.py
```

Do this on my.codesight.host

20. Install the Black Duck Code Sight license on the server where Black Duck Code Sight is installed. For more information, see *Install the Black Duck Code Sight license* on page 736.

21. Run the TeamForge post installation script. For more information, see *post-install.py*.

```
/opt/collabnet/teamforge/runtime/scripts/post-install.py
```

22. Restart the Black Duck Code Sight service.

```
/etc/init.d/collabnet restart tomcatcs
```

Do this on my.app.host

23. Restart the collabnet services.

```
/etc/init.d/collabnet restart
```

24. Apply some finishing touches and make sure everything is running smoothly.

    a) Reboot the server and make sure all services come up automatically at startup.

    b) Log into your site as the administrator.

        The value of the *DOMAIN* variable in the `site-options.conf` file is the URL to log into.

    c) Create a sample project.

        See *Create a TeamForge project*.

    d) Write a welcome message to your site's users.

        See *Create a site-wide broadcast*.

- **Review Board**: To install Review Board, see: *Set up Review Board* on page 720.
- **SELinux**: To run TeamForge in SELinux enabled mode, see *Set up SELinux* on page 580.
- **Orchestrate**: To install TeamForge Orchestrate, see *TeamForge Orchestrate installation* and *Proxy upgrade overview* on page 632.
- **Caching with Memcached**: *Install Memcached* on page 571

Upgrade Black Duck Code Sight on the same server

Follow these instructions to upgrade Black Duck Code Sight on the same server.

Log on to the server as root user always.

1. Upgrade to Red Hat Enterprise Linux/CentOS 6.7.

> ☞ **Important:** Black Duck Code Sight is not supported on RHEL/CentOS 7.x. CollabNet recommends that you have Black Duck Code Sight run on a separate RHEL/CentOS 6.7 server if you upgrade to TeamForge 8.2 on RHEL/CentOS 7.2.

- The host must be registered with the Red Hat Network. See *Platform specification for TeamForge 8.2* on page 747 for the full platform requirements.
- See *the Red Hat installation guide* for help.

> ☞ **Important:** Don't customize your installation. Select only the default packages list.

2. Check your basic networking setup.

See *Set up networking for your TeamForge server* on page 372 for details.

3. Configure your TeamForge installation repository.

    TeamForge installation repository configuration for sites with internet access

    a) Contact the *CollabNet Support* and download the TeamForge 8.2 installation repository package to `/tmp`.

    b) Install the repository package.

```
yum install -y /tmp/collabnet-teamforge-repo-8.2.0.0-1.noarch.rpm
```

    c) Refresh your repository cache.

```
yum clean all
```

    TeamForge installation repository configuration for sites without internet access

    a) Contact the *CollabNet Support* to get the auxiliary installer package for TeamForge 8.2 disconnected installation and save it in `/tmp`.

- Red Hat Enterprise Linux/CentOS 6.7 64 bit RPM package: CTF-Disconnected-media-8.2.0.0-4729.rhel6.x86_64.rpm

    b) Unpack the disconnected installation package.

```
rpm -Uvh <package-name>
```

c) 👉 **Note:** If the Red Hat Enterprise Linux/CentOS installation DVD is mounted already, skip the following instructions. If not, mount the DVD.

Mount the Red Hat Enterprise Linux/CentOS installation DVD. The DVD contains the necessary software and utilities required for installing TeamForge without internet access.

In the following commands, replace "cdrom" with the identifier for your server's CD/DVD drive, if necessary.

```
cd /media/
mkdir cdrom
mount /dev/cdrom ./cdrom/
```

If there are any spaces in the automount, unmount it first and mount it as a filepath, with no spaces.

d) Create a yum configuration file that points to the Red Hat Enterprise Linux/CentOS installation DVD.

```
vi /etc/yum.repos.d/cdrom.repo
```

Here's a sample yum configuration file.

```
[RHEL-CDROM]
name=RHEL CDRom
baseurl=file:///media/cdrom/Server/
gpgfile=file:///media/cdrom/RPM-GPG-KEY-redhat-release
enabled=1
gpgcheck=0
```

e) Verify your yum configuration files.

```
yum list httpd
yum list apr
```

f) Download the code search installation package, `CodeSearch-installer-2.1.3-1.noarch.rpm` from *https://ctf.open.collab.net/sf/go/rel4193* and save it in the `/tmp` directory. To install the Black Duck Code Sight installer, run the following command.

```
cd /tmp
rpm -ivh CodeSearch-installer-2.1.3-1.noarch.rpm
```

4. Install Black Duck Code Sight.

```
yum install teamforge-codesearch
```

5. Copy the master `site-options.conf` file from the `app` server and modify these tokens:

```
HOST_my.host.name=codesearch
```

```
DOMAIN_my.host.name=<mycodesearchbox.domain.com>
```

• If you have Black Duck Code Sight on SSL and the following external certificate tokens are not provided, the installer will generate a self-signed certificate.

For valid SSL certificates, configure the following tokens:

•
```
BDCS_SSL_CERT_FILE
```

•
```
BDCS_SSL_KEY_FILE
```

The ca.crt and chain files are optional -- leave out the tokens if you don't use the files.

•
```
BDCS_SSL_CA_CERT_FILE
```

•
```
BDCS_SSL_CHAIN_FILE
```

Advanced Black Duck Code Sight configuration settings

☞ **Note:** The following are the advanced configuration tokens which can be set once during the time of installation.

- The path where the repositories are enabled for codesearch to check out.

```
BDCS_SCAN_SOURCE_DIR_ROOT=/opt/collabnet/blackduck/scan
```

- The path where the codesearch software is installed.

```
BDCS_INSTALL_PATH=/opt/collabnet/blackduck
```

- The path where codesearch database is installed.

```
BDCS_PGSQL_HOME_DIR_ROOT=/opt/collabnet/blackduck/postgres
```

- The port number for the codesearch db server.

```
BDCS_PGSQL_PORT=55435
```

- The tomcat maximum heap memory size in megabytes.

```
BDCS_TOMCAT_MX_IN_MB=1024
```

- The shutdown port number for codesearch tomcat server.

```
BDCS_TOMCAT_SHUTDOWN_PORT=9189
```

6. Review the variables you've changed, then save the `site-options.conf` file.
7. Run the installer.

```
cd /opt/collabnet/teamforge-installer/8.2.0.0
./install.sh -r -I -V
```

8. To start the Black Duck Code Sight service, use:

```
/etc/init.d/collabnet start tomcatcs
```

- To install the license for Black Duck Code Sight, follow *these instructions*.
- To migrate TeamForge repositories follow *these instructions*.

Install the Black Duck Code Sight license

To set up Black Duck Code Sight for TeamForge 8.2, you need to install a license.

1. Get the MAC Address of your Black Duck Code Sight server.

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./license-util.sh --listmacid
```

2. To receive the license, contact your CollabNet account manager or send an email request to info@collab.net.

☞ **Note:** A commercial TeamForge license is required; the TeamForge "free option license" does not qualify.

a) Send the MAC Address of your Black Duck Code Sight server.
   You will receive the license file (in XML format).
b) Preserve the license file for future use.

3. To install the license for Black Duck Code Sight, run these commands:

```
cd /opt/collabnet/teamforge/runtime/scripts/codesearch/
./license-util.sh --install <license_file_path>
```

Type "Y" when prompted.

Bootstrap the Black Duck Code Sight instance

Follow these instructions to bootstrap the Black Duck Code Sight instance.

To bootstrap the Black Duck Code Sight instance:

1. Run the script in the command prompt.

   ```
   cd $RUNTIME_DIR/scripts/codesearch
   ./bootstrap.sh
   ```

   ☞ **Note:** Type "Y" to continue with the bootstrap process.

2. Run the TeamForge installer.

   ```
   ./install.sh -r -d /opt/collabnet/teamforge -V
   ```

3. Start the Black Duck Code Sight service.

   ```
   /etc/init.d/collabnet start tomcatcs
   ```

If the Black Duck Code Sight is running on a separate server, then run the TeamForge installer in the Black Duck Code Sight box.

Install Nexus

The following instructions are for installing Nexus Open Source version 2.9.0 or later as a stand-alone server and integrating it with TeamForge 8.2.

Nexus comes bundled with a Jetty instance that listens to all configured IP addresses on a host (0.0.0.0) and runs on port 8081 by default.

Installing Nexus is straightforward. Unzip the Nexus bundle in a directory and start Nexus.

☞ **Note:** Though Nexus can be installed in Mac OS, CollabNet did not test Nexus integration in Mac OS.

1. Log on to the Nexus server.
2. Download the Nexus 2.9.0 or later zip file and unzip the content to a directory of your choice. See *Installing Nexus*. You can find two directories, a directory that contains Nexus installation files and folders (hereinafter referred to as `<nexus-install-directory>`) and a Nexus work directory (hereinafter referred to as `<nexus-work-directory>`).

   ☞ **Important:**
   - `<sonatype-work>` is the default Nexus work directory. As a notation, `<nexus-work-directory>` is used in place of `<sonatype-work>` in this document.
   - Make sure you have full access permissions on all Nexus folders.

3. Open the command prompt and start Nexus.

   Linux:

   ```
   cd <nexus-install-directory>
   ./bin/nexus start
   ```

   Windows:

   ```
   cd <nexus-install-directory>
   \bin\nexus start
   ```

   Verify if Nexus is running by accessing the URL: `<nexus host name>:port/nexus/index.html`.

   ☞ **Tip:** Default port is 8081. In case you have multiple Nexus instances, modify the *application-port* token in `/<nexus-install-directory>/conf/nexus.properties` file.

4. Stop Nexus.

   Linux:

```
./bin/nexus stop
```

Windows:

```
\bin\nexus stop
```

Install the TeamForge-Nexus integration plugin

Once you have your Nexus server set up, install the TeamForge-Nexus integration plugin.

You must keep the following information handy before installing the TeamForge-Nexus integration plugin:

- **Nexus absolute path**: This is the path to the directory where you have Nexus installed. In other words, the path to the directory where you have your Nexus files unzipped. For example, `/u1/nexus/nexus-2.11.4-01/`.
- **TeamForge host URL**: The URL to access TeamForge. For example, `http://ctf.cloud.collab.net`.
- **TeamForge administrator user name and password**.
- **Nexus Application Name**: The name given to your Nexus integration. In other words, the name found in the Nexus integrated application configuration file. For example, see the pebble application's *pebble-app.xml* on page 818 for more information.
- **Nexus Application Prefix**: The prefix chosen for Nexus. In other words, the prefix found in the Nexus integrated application configuration file. For example, see the pebble application's *pebble-app.xml* on page 818 for more information.
- **Nexus URL**: The fully qualified Nexus URL.

1. Log on to the Nexus server.
2. Stop Nexus if it's running.

   Linux:

   ```
   ./bin/nexus stop
   ```

   Windows:

   ```
   \bin\nexus stop
   ```

3. *Download* the `CTF-Nexus-Integration-Plugin-2.0.zip` file.
4. Unzip the `CTF-Nexus-Integration-Plugin-2.0.zip` file.

   Linux:

   ```
   cd <nexus-work-directory>/nexus/plugin-repository
   unzip CTF-Nexus-Integration-Plugin-2.0.zip
   ```

   Windows: Use a utility such as WinRAR.

5. Install the TeamForge-Nexus integration plug-in.

   ```
   sudo java -jar <nexus-work-directory>/nexus/plugin-repository/CTF-Nexus-
   Integration-Plugin-2.0/util/installer.jar -enable
   ```

   Enter the Nexus absolute path and TeamForge host URL when prompted.

6. Start Nexus.

   Linux:

   ```
   cd <nexus-install-directory>
   ./bin/nexus start
   ```

   Windows:

   ```
   cd <nexus-install-directory>
   \bin\nexus start
   ```

7. Once Nexus is up and running, upload the Nexus IAF descriptors to TeamForge.

```
java -jar <nexus-work-directory>/nexus/plugin-repository/CTF-Nexus-
Integration-Plugin-2.0/util/installer.jar -installxml
```

Enter the TeamForge Host URL, TeamForge admin user name and password, Nexus application name, Nexus application prefix and Nexus URL when prompted.

Upgrade the TeamForge-Nexus integration plugin

CollabNet releases new versions of the TeamForge-Nexus integration plugin. It is recommended to upgrade your TeamForge-Nexus integration plugin to the available latest version.

1. Log on to the Nexus server.
2. Stop Nexus if it's running.

   Linux:

   ```
   ./bin/nexus stop
   ```

   Windows:

   ```
   \bin\nexus stop
   ```

3. Modify the `ctf_nexus.properties` file.

   ```
   vim sonatype-work/nexus/conf/ctf_nexus.properties
   ```

   a) Replace `TIME_TO_HOLD_CACHE=60` with the following two properties:

   - `TIME_TO_HOLD_USER_CACHE=1800`
   - `TIME_TO_HOLD_PERMISSION_CACHE=1200`

   ☞ **Note:** `TIME_TO_HOLD_USER_CACHE` and `TIME_TO_HOLD_PERMISSION_CACHE` are in seconds.

4. Back up the existing TeamForge-Nexus integration plugin directory: `<nexus-work-directory>/nexus/plugin-repository/CTF-Nexus-Integration-Plugin-1.0.2`.

5. *Download* the `CTF-Nexus-Integration-Plugin-2.1.1.zip` file.

6. Unzip the `CTF-Nexus-Integration-Plugin-2.1.1.zip` file you downloaded into the `<nexus-work-directory>/nexus/plugin-repository/` directory.

   Linux:

   ```
   cd <nexus-work-directory>/nexus/plugin-repository
   unzip CTF-Nexus-Integration-Plugin-2.1.1.zip
   ```

   Windows: Use a utility such as WinRAR.

7. Start Nexus.

   Linux:

   ```
   cd <nexus-install-directory>
   ./bin/nexus start
   ```

   Windows:

   ```
   cd <nexus-install-directory>
   \bin\nexus start
   ```

TeamForge-Nexus release notes

Look here for information about the release highlights and bug fixes of various Nexus integration plugin versions supported by all the TeamForge releases starting from TeamForge 8.0.

Latest Nexus plugin: CTF-Nexus-Integration-Plugin-2.1.1.

Download the latest Nexus plugin: *CTF-Nexus-Integration-Plugin-2.1.1*

To upgrade to Nexus plugin 2.1.1, check this page: *Upgrade the TeamForge-Nexus integration plugin* on page 739
.

Release highlights of Nexus 2.1.1 plugin

- Available from TeamForge 17.4 release.
- Improved peformance while loading Nexus UI.

Bug fixes in Nexus 2.1.1 plugin

Fixed the issue in which the '500: Internal Server Error' was shown instead of '401: Unauthorized Error' for any user activities in Nexus after a session timeout for the Nexus versions 2.9, 2.10, and 2.11.

Previous releases

- *Nexus plugin 2.0.1 release notes*
- *Nexus plugin 2.0 release notes*

Known Limitations with TeamForge-Nexus integration

Here's a list of known limitations of TeamForge-Nexus integration.

- While it is possible for users (all TeamForge users but site administrators) with 'create repository' permission to create binary repositories via the TeamForge **Binaries** application, such users cannot create repositories directly on the Nexus server (using the Nexus UI) as there is no TeamForge project mapping available in Nexus.
- A TeamForge user cannot be configured as an anonymous user in Nexus as TeamForge users are not available in the Nexus database.
- TeamForge broadcast messages and license notifications are not visible in Sonatype Nexus pages in TeamForge. This is due to a limitation with the TeamForge-Nexus integration plugin.
- When the TeamForge Project administrator changes the existing binary permissions for a user, the changes will not immediately take place due to the cache implementation in Nexus to improve the performance of session handling. Due to this limitation, when a user tries to create a new Nexus repository, he will get the permission denied error. Hence the user must wait for the changes to take effect. However, the user can view the changes immediately on a standalone Nexus application by restarting his session on it.

Nexus plugin 2.0 release notes

This release notes provides the release highlights and bug fixes of Nexus 2.0 plugin.

Name of the plugin: CTF-Nexus-Integration-Plugin-2.0.

☞ **Note:** You must upgrade to latest Nexus plugin version regardless of whichever TeamForge version you have installed or upgraded. To upgrade to latest Nexus version, check this page: *Upgrade the TeamForge-Nexus integration plugin* on page 739 .

Release highlights of Nexus 2.0 plugin

- Usability of the Nexus installer (used to integrate Nexus with TeamForge) has been enhanced.
- Improved peformance while loading Nexus UI.

Bug fixes in Nexus 2.0 plugin

- When users log on to Nexus using CTF user credentials, the Nexus UI showed poor performance. This is fixed.
- Fixed the time delay due to the presence of Nexus plugin in Maven builds.

Known Limitations with TeamForge-Nexus integration

Here's a list of known limitations of TeamForge-Nexus integration.

- While it is possible for users (all TeamForge users but site administrators) with 'create repository' permission to create binary repositories via the TeamForge **Binaries** application, such users cannot create repositories directly on the Nexus server (using the Nexus UI) as there is no TeamForge project mapping available in Nexus.
- A TeamForge user cannot be configured as an anonymous user in Nexus as TeamForge users are not available in the Nexus database.
- TeamForge broadcast messages and license notifications are not visible in Sonatype Nexus pages in TeamForge. This is due to a limitation with the TeamForge-Nexus integration plugin.
- When the TeamForge Project administrator changes the existing binary permissions for a user, the changes will not immediately take place due to the cache implementation in Nexus to improve the performance of session handling. Due to this limitation, when a user tries to create a new Nexus repository, he will get the permission denied error. Hence the user must wait for the changes to take effect. However, the user can view the changes immediately on a standalone Nexus application by restarting his session on it.

TeamForge Orchestrate JIRA Integration: an overview

The TeamForge Orchestrate JIRA integration enhances further the extensible toolchain model of TeamForge by offering association and traceability between JIRA, TeamForge, and various orchestrated third-party tools. The TeamForge Orchestrate JIRA adapter enables JIRA as an alternative tracker for use with TeamForge by detecting and storing associations between JIRA issues and TeamForge managed SCM activities, like commits, builds and so on.

For example, the graphical representation of associations helps you perform a root cause analysis of an event, say, a build failure. Similarly, the traceability feature provides data useful for the purpose of audit or governance. For detailed information, see *JIRA adapter*.

Set up Pebble

Pebble is a blogging application that has been enhanced to support quick and easy integration with TeamForge.

Install Pebble

To see what you can do with an integrated application in TeamForge, start by installing Pebble, a blogging application that you can configure to work as part of your TeamForge site.

1. Get the Pebble installer package from *open.collab.net* and unzip it.
2. Modify these values in the `installer/install.conf` file to suit your installation environment.

Option	Description
pebble.base.dir	Path where you want Pebble to be installed on this host.
ctf.baseurl	Absolute URL of the TeamForge site that you want to associate to, such as `https://my.ctf.instance/`
tomcat.port	Pebble runs on Tomcat. This token indicates which port you want Tomcat to be running on. Make sure there are no other services running on that port.
domain	The base url to be used for Pebble, such as `my.ctf.instance`. (You don't need `http:` or `https:` here.)
timezone	The time zone Pebble will use to timestamp blog entries.
java_home	Path to a JDK 1.6.x instance.
protocol	`http` if SSL is not being used; `https` if SSL is being used.
data.dir	Path in the file system where Pebble blogs will be stored.

3. Run the installer.

```
sudo python install.py -i -r
```

4. Set up the initial blog data.

```
sudo python bootstrap-data.py
```

This is known as "bootstrapping" the application.

☞ **Tip:** You can bootstrap again if you want to start from scratch, but any existing blogs will be deleted if you do.

5. Restart the Pebble application.

```
/etc/init.d/pebbled stop
/etc/init.d/pebbled start
```

You should now have a working Pebble instance ready to work with TeamForge. The installer has created two configuration files: `installer/conf/pebble-app.xml` and `installer/conf/pebble-dep.xml`. See *Integrate Pebble into your TeamForge site* on page 742 for how to use them.

If you have installed Pebble with SSL, restart the TeamForge server before integrating Pebble with TeamForge. You can restart the TeamForge server using `/etc/init.d/collabnet restart all`

Integrate Pebble into your TeamForge site

When the sample Pebble blogging application has been installed on your site, you can make it available for projects on your TeamForge site.

Pebble must be installed and configured before you can integrate it into your TeamForge site. See *Install Pebble* on page 741.

When you have integrated Pebble, projects on your site can add Pebble to their set of collaboration tools. The blogs they create will share many of the core TeamForge features, such as authorization, authentication, go-urls, association, linkification, templating, Project Pages components, and source code management support.

1. Log into TeamForge as an admin user.
2. Go to **My Workspace > Admin**.
3. Click **INTEGRATED APPS** from the **Projects** menu.
4. Click **Create**.
5. Use the **Browse** window to find the two configuration files that enable the Pebble application to work as a part of TeamForge:

   - `pebble-app.xml` (Application configuration file): Contains the text strings for the Pebble user interface.
   - `pebble-dep.xml` (Deployment configuration file): Contains the data that Pebble needs to interact with TeamForge.

   Click **Next**.

6. On the **Preview** screen, review the parameters you set in the configuration files.

   ☞ **Note:** You may have to revise one or more values to ensure they are valid.

7. Click **Save**.

The Pebble application is now available for all projects on your site. You can direct project administrators to *the project admin help* for instructions on adding it to their own project toolbars.

☞ **Note:** You may need to adjust your site's look and feel to support your integrated application. See *the site admin help* for details.

Add Pebble to a TeamForge project

When the site administrator has made the sample Pebble blogging application available, project administrators can add it as one of their project tools.

When you add Pebble to your project, it works just like the other TeamForge tools, with features such as authorization, authentication, go-urls, association, linkification, templating, Project Page components, and source code management support.

For this example, we'll call your project "testproject" and we'll assume you have Project Admin rights in that project.

1. Click **PROJECT ADMIN** from the **Project Home** menu.
2. Click **Tools** to see the list of integrated applications available in the site.
3. Click **Add Tool**.
4. Select **Pebble Blog** from the list of tools displayed.

   Set the values that make sense for your TeamForge installation, then click **Save**.

Option	Description
Prefix	Specify a unique alphanumeric string that will identify this tool throughout the site. For example, suppose you set your prefix to `ZZ`. Entering `ZZ_20100612` in the **Jump to ID** box will get you all blog posts on June 12, 2010 in the "testproject" project. Another may also add Pebble as a tool, with a prefix of YY. Jumping to `YY_20100101` will get all the blog posts on January 1, 2010 from that project.
	👉 **Note:** The tool prefix cannot be changed after you have set it. Your site administrator may have elected to make other parameters permanent too, such as the blog name.
Blog Name	This will appear as a banner in the blog for this project. You can enter any name.
Blog Description	This will appear under the Blog Name in the blog for this project. You can enter any description.
Rich Text Editor Enabled	Select this if your users want word-processor-like writing support when they create blog posts.
Number of Entries in Blog Home	You can choose to have 3, 5, 7 or 9 entries in the blog homepage.

   If a **Pebble Blog** button appears along with the prefix as a tooltip when you mouse over the button, your job is done.

TeamForge–TestLink integration: an overview

TestLink is a web-based test management system that supports all the various components and processes involved in a testing process. Using TestLink, you can create test suites, execute test cases, create custom reports, generate test plan metrics and so on.

TeamForge 7.0, 7.1, 7.2, 8.0 and 8.2 (CentOS 6.7) supports an integration with TestLink to track and synchronize requirement and defect tracker artifacts between these two systems.

The following is a summary of the workflow between TeamForge and TestLink integration:

- Install and configure TestLink in your TeamForge site.
- Create a project in TeamForge.
- Create all your requirement tracker artifacts such as epics, stories and tasks, and a single defect tracker in your TeamForge project.
- Associate the TeamForge requirement tracker artifact with TestLink. A test suite is created automatically in TestLink.

- In TestLink, create test cases for the test suite, which is associated with the requirement tracker artifact in TeamForge.
- Create a test plan that contains multiple test cases for a specific build.
- Execute the test cases.
- Failed test cases are updated as defects in the associated TeamForge defect tracker.

TestLink features

TestLink provides the following additional features catering to your testing needs:

- **Reports and charts:** You can generate multiple types of reports based on a build, an assignment or on the test case status. You can also generate reports of test cases that were not assigned to any test plan. In addition, you can view the test result matrix at the test plan level and assess the number of times a test case has been executed. You can also analyze test case results with the help of pie charts or bar charts. You can create custom reports using the framework available in TestLink.
- **Dashboard:** At the project level, you can create a dashboard with information about the overall progress of all the test cases available within a project.
- **Import/Export:** If there are test cases that can be re-used across projects, you can import or export them as `.xml` files at the test suite or test case level from one project to another.
- **E-mail notification:** You can set this option to send e-mail notifications to appropriate users when a test case is assigned to them or send generated reports to specific users automatically.
- **Assign keyword tags:** You can group test cases based on your needs and assign keywords to them. Then you can perform a search on these keyword tags to locate the entire set of related test cases.
- **Custom field:** You can customize your test cases, your test plan tracker based on your business requirements.
- **Other languages support:** TestLink supports Chinese, Japanese and Korean languages.
- **Versioning test cases:** You can maintain different versions of test cases using which you can compare and track the changes you have made to each version of a test case.

For detailed information about TeamForge and TestLink integration, see *TeamForge test management with TestLink v2.1.2* .

Authenticate your integrated application with TeamForge

To help third party developers write integrated applications, CollabNet provides an SDK. At the heart of the SDK is the IntegratedAppSupport helper class. You use IntegratedAppSupport to authenticate integrated application requests with TeamForge, and provide context information for subsequent processing of forms and links within the application.

You can download the SDK from *here*.

You must call the *IntegratedAppSupport* class for every request made by the integrated application. This class takes HttpServletRequest and HttpServletResponse for each request and determines whether the user is already authenticated. It also provides project- and user-related information that can be used throughout the request.

☞ **Tip:** It is a good idea to store this as a ThreadLocal so that it can be used from anywhere in the application.

Here's what you need to do to call IntegratedAppSupport from the application:

* To create an object, pass in these three parameters:

    * TeamForge Base URL -- this is where TeamForge is installed
    * Integrated application URL -- this is where the application is installed
    * Application Name -- the name of the application as defined in the integrated application's descriptor

    Typically these parameters are stored as web.xml initialization parameters so that they are passed to IntegratedAppSupport's constructor.
* When you have a validated IntegratedAppSupport object, store it in a ThreadLocal object so that it can be used in other places. In the example, when constructing the blog URLs in the Pebble blogging application, we get IntegratedAppSupport from ThreadLocal and retrieve the IntegratedAppId from there.

    (Refer to /src/net/sourceforge/pebble/domain/Blog.java's getUrl method)

    Once the object is created, another method called processRequest is called on it. This takes an HttpServletRequest and HttpServletResponse object that the servlet or a filter gets.
* For each request that enters your system and needs to be validated, create an IntegratedAppRequest object.

    Subsequently, you also need to call the processRequest method to set the required parameters internally.

Where you construct the IntegratedAppSupport and call ProcessRequest subsequently depends on the architecture of your application. We've provided two cases -- for servlet filters and httpservlets.

* If your application has servlet filters, use them as shown in the example Pebble application. (Refer to src/net/sourceforge/pebble/web/filter/BlogLookupFilter.java and src/com/collabnet/pebble/filter/CTFAuthenticationFilter.java.)
* If your application does not use servlet filters, but directly calls an HttpServlet as the point of entry, place the call to IntegratedAppRequest where the application enters the HttpServlet code.

ProcessRequest does the following tasks:

* If the URL is coming in for the first time from the TeamForge, it will have the singlesignon token. The method validates the token and stores a cookie which identifies the soap session for the user as well as the project for which this request is being made. This will also be the case for a Go URL coming in from TeamForge.
* If it is a subsequent request (from a form submission or a click from a previous page), then the cookie information is picked up and used in IntegratedAppSupport. For this, the integrated application expects that the linkid (the id linking the integrated application and the project, obtained through IntegratedAppSupport.getIntegratedAppId()) is provided either as a request attribute or as a request parameter or in the request path -- it can be any one of these:

    ```
    http://my.integrated.application/reach/linkid/prplxxxx/me
    http://my.integrated.application/reach/me?linkId=prplxxxx
    ```
    a request parameter obtained through any other means and set as a Request Attribute.

It is advisable to set it so that all originating URLs (forms and other links to the application) have the /linkid/ prplxxxx in their request URL. For example, in the Pebble application (refer to ./src/net/sourceforge/pebble/ domain/Blog.java's getUrl method), the URL for each link that goes out of the blog is added a /linkid/prplxxxx to its request URL. This helps subsequent URLs to be validated correctly. You can retrieve the linkid using IntegratedAppSupport.getIntegratedAppId.

Add the TeamForge button bar to your integrated application

To help users navigate easily between the integrated application and the TeamForge site, you can make the TeamForge button bar available from the integrated application user interface.

CollabNet provides a tag library for JSP-based applications to plug the TeamForge button bar into their pages. The ButtonBar tag library is packaged in the SDK.

👉 **Note:** Only JSP is supported in this release. More helpers for other view technologies are on the way.

👉 **Tip:** Reference the `buttonbar` tag in a default layout page so that all pages can use the same tag library.

1. Copy the taglib definition to your `WEB-INF` folder.
   a) The `integratedapp-taglib.tld` file is available in the SDK Package.
   b) The `.jar` file containing the `IntegratedApp` taglib is available in the `integratedapps.jar` (which also contains the other helper classes).
2. Add the following to your `web.xml` file to define the tag library:

```
<taglib>
     <taglib-uri>/tags/iapp</taglib-uri>
     <taglib-location>/WEB-INF/integratedapp-taglib.tld</taglib-location>
</taglib>
```

3. Set up the Layout JSP page.
   a) Get an instance of IntegratedAppSupport and add it to the Request attribute.
   b) Set the "linkid" in the Request attribute.

Now you can draw the button bar on all integrated app pages so that the user can navigate between TeamForge and the integrated application.

> For example, this is how we integrated Pebble, our example blogging application, with the TeamForge button bar. (For more detail, see `iapp.jsp` in the Pebble source.)
>
> ```
>         <%@ page
>   import="com.collabnet.ce.soap50.integratedapps.util.IntegratedAppSupport"
>   %>
>         <%@ taglib prefix="iapp" uri="http://jakarta.apache.org/
>   tomcat/pluggableApp-taglib" %>
>         <%
>         IntegratedAppSupport support = (IntegratedAppSupport)
>         request.getSession().getAttribute("intAppSupport");
>         request.setAttribute("intAppSupport", support);
>         request.setAttribute("linkId",
>   support.getIntegratedAppId());
>         %>
>         <iapp:viewbuttonbar />
> ```

Internationalize your integrated application

You can configure your integrated application to display languages based on the TeamForge site user's browser locale.

These parameters can be internationalized:

- The description of the integrated application.
- The title and description of the integrated application when it appears as a component on a project page.
- The name and description of any configuration parameter.

1. In the *XML application configuration file*, select the values for which you want to provide localized content.
2. Inside the `bundles` tag, create a `bundle` tag identical to the default English `bundle` tag.

   (Keep the `en` bundle.)
3. Change the value of the `locale` attribute of the new `bundle` tag to the language you are going to provide.

   The value for each of the internationalized tags must start with `l10n`.

   For example:

```
    <bundles>
            <bundle locale="en">
                <key name="l10n.application.description">My Test App</key>
                <key name="l10n.blogname.title">Blog Name</key>
                <key name="l10n.blogname.description">Please provide a name
 for the Blog. This appears on all blog pages</key>
                ........
            </bundle>
            <bundle locale="ja">    <!-- Japanese Language Translations -->
                <key name="l10n.application.description">My Test App</key>
                <key name="l10n.blogname.title">Blog Name</key>
                <key name="l10n.blogname.description">Please provide a name
 for the Blog. This appears on all blog pages</key>
                ........
            </bundle>
    </bundles>
```

4. Save the application configuration file.

Reference information about TeamForge

Use this reference information to get deeper detail on configuration files, logs, scripts and other resources you use to administrate TeamForge.

Platform specification for TeamForge 8.2

This is the hardware and software platform that TeamForge runs on.

Hardware requirements for CollabNet TeamForge 8.2
This topic lists the CPU, RAM and JVM Heap Size recommendations for Small, Medium, Large and Extra-large sites.

	Small	**Medium**	**Large**	**X-Large**
Users	100	500	1000-5000	10000+
CPU	Octa-core	12-core	> 12-core	> 16-core
RAM	16GB	24GB	32GB	32GB
JVM Heap Size	1.5GB	3GB	6GB	> 8GB

200 GB (or more) hard drive [*]

The following table highlights the factors that can impact TeamForge performance. Numbers are indicative. Anything more than the prescribed numbers may impact the performance.

---

[*] The required hard drive capacity depends on the estimated amount of document and file release uploads.

	Small	Medium	Large	X-Large
Artifacts	15000	70000	100000	100000
Flex Fields	25	50	100	100
Projects	20	80	500	500
Integrations	0-1	0-2	0-2	0-2
Integrated Applications	0-2	0-3	3+	3+

☞ **Important:** On Medium, Large and X-Large sites, it is highly recommended that you install the TeamForge Application, Database and SCM services on separate 64-bit servers based on the usage pattern.

Software requirements for CollabNet TeamForge 8.2

This is the official list of software tested for compatibility with CollabNet TeamForge 8.2.

☞ **Important:** Do not customize your operating system installation. Select only the default packages list.

Software version requirements for 64 bit platform

		Red Hat Enterprise Linux 6.7 or 7.2[1] | **CentOS 6.7 or 7.2**
Database	PostgreSQL[3]	9.3.6
	Oracle DB**	12c
	Oracle Client	12c
Java[4]	JDK	1.8.0_45
	Wildfly/Jboss	8.2.0.Final
	Tomcat	8.0.22
Lucene		4.4
Open LDAP		2.4.23
Quartz Job Scheduler		1.8.3
Browser	Google Chrome	46
	Mozilla Firefox	40, 41 and 42
	Microsoft IE	9, 10 and 11
	TeamForge user interfaces are best viewed at screen resolution of at least 1280 x 800 (or more) pixels.	
yum package manager		3.2.29 or earlier versions

---

[3] PGTurant tool is available for quick pg_upgrade. See *Upgrade PostgreSQL using PGTurant* on page 573 for more information.

[4] For larger installs, the max JVM heap size should be increased to 1024MB (or larger), depending on available resources on the server.

Red Hat Enterprise Linux 6.7 or 7.2[1] | CentOS 6.7 or 7.2
CollabNet TeamForge 8.2 supports the following software versions for integration

Black Duck Code Sight[5]	2.1.3
Review Board	2.0.11
GIT	2.4.2 or later
Subversion	1.8.14
Subversion Edge	5.0
CVS	1.11.x
ViewVC	1.1.24
Nexus	2.9.0 or later
Jira	6.3-7.0

Orchestrate

Orchestrate *Installation requirements* on page 618

CollabNet TeamForge 8.2 on Safari

CollabNet TeamForge 8.2 is tested for compatibility with Safari 6.0.5 (8536.30.1).

CollabNet TeamForge 8.2 on VMware and ESXi

The CollabNet TeamForge 8.2 is tested for compatibility with VMware Player 5.0 or later. For Windows and Linux versions supported by VMware Player 5.0 or later, see *VMware Player Documentation*.

The CollabNet TeamForge 8.2 is tested for compatibility with ESXi 5.1 or later.

PostgreSQL or Oracle?

PostgreSQL 9.3.6 is installed automatically when you install TeamForge 8.2. If you intend to use Oracle, CollabNet recommends that you let the installer run its course, make sure things work normally, and then set up your Oracle database and switch over to it.

If you want to use Oracle as your database, consider the following points:

- TeamForge 8.2 supports Oracle server 12c and Oracle client 12c.
- Oracle express edition is not supported for both client and server.
- Black Duck Code Sight (BDCS) 2.1.3 was tested with PostgreSQL 9.3.6 only. BDCS with Oracle was not tested. If you want BDCS, you may install BDCS on the PostgreSQL database that is installed by default during TeamForge installation.
- Review Board 2.0.11 was tested with PostgreSQL 9.3.6 only. Review Board with Oracle was not tested. Note that Review Board must be installed on the TeamForge application server and therefore uses the PostgreSQL database that is installed by default during TeamForge installation.

---

[5] ☞ **Important:** Black Duck Code Sight has no support for RHEL/CentOS 7.x. CollabNet recommends that you install or upgrade Black Duck Code Sight on a separate RHEL/CentOS 6.7 server if you choose to install or upgrade TeamForge 8.0 or later on a RHEL/CentOS 7.2 server.

- GIT integration works only with PostgreSQL. The Git integration will use PostgreSQL even if your TeamForge site uses Oracle.

The efficiency of your database can have an impact on your users' perception of the site's usability. If your site uses a PostgreSQL database (which is the default), you may want to consider tuning it to fit your specific circumstances. The default settings are intended for a small-to-medium site running on a single server. See *What are the right PostgreSQL settings for my site?* on page 909 for recommendations from CollabNet's performance team on optimizing PostgreSQL for different conditions.

Microsoft applications

The following Microsoft applications have been tested with CollabNet TeamForge:

- Microsoft Office XP (with Service Pack 3) on Windows XP Service Pack 2 and Windows 2000 Service Pack 4.
- Microsoft Office 2003 (with Service Pack 1) on Windows XP Service Pack 2 and Windows 2000 Service Pack 4.

Hardware and software requirements for CollabNet TeamForge 8.2 on a virtual machine

This is the hardware and software platform requirements if you run TeamForge on VMware Player.

- An operating system that VMware Player can run on. For Windows and Linux versions supported by VMware Player, see the *VMware Player documentation*.
- 4 GB system RAM
- 20 GB available disk space
- 2 Ghz Pentium 4 or equivalent processor

> ☞ **Note:** CollabNet TeamForge may also run on some other VMware products, such as VMware Workstation 5.5. However, these instructions are only for running TeamForge on VMware Player.

Scripts installed with TeamForge

System administrators can use these utilities to control the behavior of the application.

add_orchestrate_to_projects.py

Run the `add_orchestrate_to_projects.py` script to add the Orchestrate integrated application to existing projects.

Usage

Use the following command to run the this script.

```
[RUNTIME_DIR]/scripts/add_orchestrate_to_projects.py [[--tfuser=<username>]
[--tfpasswd=<password>]]
```

Parameters

Paramater	Description
-u | --tfuser	The TeamForge site administrator user name.
-p | --tfpasswd	The password for TeamForge user.
-s | --silent	To prevent printing progress message on standard-out.
-h | --help	To view help information for the script.

backup-data.py

The `backup-data.py` script makes a compressed copy of the site's data. The script deletes any backup existing in the destination directory. By default, the Teamforge and the reporting database are backed up to the destination directory. If reporting is disabled, only the Teamforge database is backed up.

Overview

`backup-data.py` makes compressed copies of these data resources:

- Subversion repositories
- CVS repositories
- The data directory (`/var`)
- The SourceForge database

Usage

```
./backup-data.py --destination_dir=<directory name>
```

Location

`/opt/collabnet/teamforge/runtime/scripts/`

Options
--destination_dir

> Directory where the compressed copy of the site data is to be stored.

--help | -h

> Print this usage message and exit.

backup-rb-data.py
The `backup-rb-data.py` script is used to back up the Review Board application data.

Overview

The Review Board application data includes the database and files. If there are any files in the backup directory, the script overwrites these files.

Usage

Run this script as follows:

```
python ./backup-rb-data.py --backupdir={dir}
```

Options

The following options are available for the `backup-rb-data.py` script:

-b | --both	Back up both the database and the filesystem. This is the default option.
-d | --database	Back up the database.
-f | --files	Back up the filesystem.
-h | --help	Provides a list of all available options for this script.

bootstrap-data.sh
The `bootstrap-data.sh` script prepares application and database data for new installations. Preparing application and database data is referred to as "bootstrapping" the data.

Overview

☞ **Important:** This script is only for new installations. If you run it on a site that already has data, all data will be wiped.

This script resides in the `<installation_source>` directory and calls the `wrapper-bootstrap-data.py` script when run. The `[log_file_directory]/runtime/bootstrap.log` file is created when this script is run. All success and error messages from this script are written to this log file.

Usage

Run this script as follows:

```
cd <installation_source>
  ./bootstrap-data.sh -d /opt/collabnet/teamforge
```

Example

The following command forces a bootstrap of the data, showing all actions on the screen:

```
  ./bootstrap-data.sh -n -F -V -d /opt/collabnet/teamforge
```

Options

The following options are available for the `bootstrap-data.sh` script:

-d | --directory	Specify installation directory. This argument is required.
-h | --help	Provides a list of all available options for this script.
-n | --non-interactive	Runs the script in a non-interactive mode. The script will fail with an error message when used with this option if an existing `[DATA_DIR]` is located. You can use the -F option to force bootstrapping on sites that have an existing `[DATA_DIR]`.
-F | --force	This option is only valid when the -n option is used. This option forces the bootstrapping of data when a `[DATA_DIR]` exists.
-V | --verbose	Writes all script actions to the screen. Without this option the script runs silently and logs messages to the `[log_file_directory]/runtime/bootstrap.log` file.
-q | --quiet	Do not show script output.
-f | --site-options=[filename]	Points to the `site-options.conf` configuration file for the site. This argument is optional.

Cluster location

This script runs on the application server machine.

bootstrap-reporting-data.sh

The `bootstrap-reporting-data.sh` script prepares the datamart data for new installations.

Overview

☞ **Important:** This script is only for new datamart installations. If you run it on a site that already has reporting data, all data will be wiped.

The success and error messages from this script are written to the `[log_file_directory]/runtime/bootstrap.log` file.

Usage

Run this script as follows:

```
cd /opt/collabnet/teamforge/runtime/scripts
./bootstrap-reporting-data.sh
```

Options

The following options are available for the `bootstrap-data.sh` script:

-h | --help	Provides a list of all available options for this script.
-n | --non-interactive	Runs the script in a non-interactive mode. The script will fail with an error message when used with this option if an existing `[REPORTS_DATA_DIR]` is located. You can use the -F option to force bootstrapping on sites that have an existing `[REPORTS_DATA_DIR]`.
-F | --force	This option is only valid when the -n option is used. This option forces the bootstrapping of data when a `[DATA_DIR]` exists.
-V | --verbose	Writes all script actions to the screen. Without this option the script runs silently and logs messages to the `[log_file_directory]/runtime/bootstrap.log` file.
-q | --quiet	Do not show script output.

Cluster location

This script runs on the application server machine.

CodeSightMigration.sh

The `CodeSightMigration.sh` script is used to create Black Duck Code Sight projects for existing TeamForge repositories. After migration, the code search functionality is enabled for migrated repositories.

Location

`runtime/scripts/codesearch/CodeSightMigration.sh`

Usage

Before running this script, do the following:

- Turn on the CODESEARCH_ENABLED flag on the TeamForge server.
- Make sure the tokens specific to Black Duck Code Sight, such as the server name and port, are valid.

☞ **Note:** Log on with site administrator credentials and run this script on the TeamForge application server.

Example

The following command enables Code Search for the given repositories.

```
[root@cuXX codesearch]# ./codesightmigration.sh --filename=/tmp/repository-
ids.txt
```

Targets

The following targets are available for this script:

--target=create	This is the default target and is used for creating Black Duck Code Sight projects for existing TeamForge repositories.	
	[-- filename=<repositories-input-filename>]	Default: /tmp/repository-ids; specify any other filename with the --filename parameter.
	--username=<admin-username>	Site admin username
	--password=<admin-password>	Site admin password
	[--totalthreads=<number of threads to spawn>]	Default: 25; the total number of threads needed to run Black Duck Code Sight migration.
--target=delete	Delete Black Duck Code Sight projects for the corresponding TeamForge repositories.	
	[-- filename=<repositories-input-filename>]	Default: /tmp/repository-ids; specify any other filename with the --filename parameter.
	--username=<admin-username>	Site admin username
	--password=<admin-password>	Site admin password
	[--repoids=<repo-id1,repo-id2,repo-id3>]	Default: delete all repositories as specified in the input filename or delete the repositories specified in the command line
--target=status	Status of Black Duck Code Sight projects created for TeamForge repositories.	
	[-- filename=<repositories-input-filename>]	Default: /tmp/repository-ids; specify any other filename with the --filename parameter.
	--username=<admin-username>	Site admin username
	--password=<admin-password>	Site admin password
	[--repoids=<repo-id1,repo-id2,repo-id3>]	Default: status of all repositories as specified in the input filename or the status of the repositories specified in the command line

Logs

Three types of logs are available. They are located at `site/log/codesearch`.

- `codesightdebugmessage.log`: used for debugging Black Duck Code Sight issues
- `codesightprojectstatus.log`: shows project status for repositories; created when target = status is run
- `codesightfailedrepositories.log`: shows project status for failed project repositories; created when target = status is run

Comments

There is a another script that generates the final SQL to fetch repository IDs for a given list of repository paths. See *repository-sql.sh* for more information.

The collabnet script

Run this script to start or stop TeamForge, or to get the status of the application or a component.

Overview

You can use this script to start or stop the application as a whole or to start and stop an individual service. You can also use it to determine the status of an individual service.

☞ **Important:** On production sites, this script must be invoked by the root user.

Usage

Run this script as follows:

```
/opt/collabnet/teamforge/runtime/scripts/collabnet [--verbose|-V] [--
service|-s serviceName] <command>
```

For example, the following command checks the status of the jboss component:

```
<SITE_DIR]/runtime/scripts/collabnet -s jboss status
```

Parameters

Command	Action to perform. The supported commands are:	
	start	Starts the application / service
	stop	Stops the application / service
	status	Provides the status of the service(s)
	restart	Restarts the application / service

help	Prints this message and exits.
-s | service serviceName	Perform the command for the service serviceName
verbose	Print debug messages
-q | --quiet	
-F | --force	Force option to perform the specifed operation forcefully
-S | --silent	To perform the operation silently without providing the output

Logging

`collabnet` writes entries to the following logs:

- `log/runtime/service.log`: The master service log.
- `log/{service}/service.log`: Log entries from starting up individual services end up in the `service.log` file of the corresponding service log folder (e.g. `log/apps/service.log`)

datamart-oracle-setup.sh

This script is used for setting up the Oracle datamart in CTF advanced mode installation.

Usage

You can run this script from the command line as follows:

```
[RUNTIME_DIR]/scripts/datamart-oracle-setup
```

The datamart-oracle-setup.sh is an interactive script and does not take any parameters.

Comment

[SCRIPTS_DIR]/set-reports-readonly-user-permission.py script needs to be executed after bootstrap or migration to complete the datamart setup.

datamart-pgsql-setup.sh

This script is used for setting up the PostgreSQL datamart in CTF advanced mode installation.

Usage

You can run this script from the command line as follows:

```
sudo [RUNTIME_DIR]/scripts/datamart-pgsql-setup
```

The datamart-pgsql-setup.sh is an interactive script and does not take any parameters.

Comment

[SCRIPTS_DIR]/set-reports-readonly-user-permission.py script needs to be executed after bootstrap or migration to complete the datamart setup.

db.py

The `db.py` script can be used to dump and restore a PostgreSQL database.

Overview

This script can be used only for the PostgreSQL service. Don't run this script on a remote database. Execute the script only when the database is up and running.

Usage

Run this script as follows:

```
./db.py --action=<action> --path=<destination directory>
```

Options

Required options:

`-a	--action`	Values: dump or restore

`-f	--path`	Path where the database backup file will be created. Must be a directory owned by the postgresql user (usually `/var/lib/pgsql/9.0/`). Can be a new directory.

Optional options:

`-t	--type`	Specifies the type of database (ctf or reporting).
`-h	--help`	Print this usage message and exit.

☞ **Note:** The options -a and -f are mandatory.

domain_change_db.py

The `domain_change_db.py` script handles all the steps required to change the domain name in the site database. It does not change anything in the file system.

☞ **Note:** Changing the domain through any other mechanism may cause problems.

Usage

Execute this script with a command like this:

```
[RUNTIME_DIR]/domain_change_db.py [--debug] [--dir] --old={domain_name} --
new={domain_name}
```

☞ **Note:** The new domain name must match the value defined for the DOMAIN token in the `site-options.conf` file.

Options

The `domain_change_db.py` script provides the following parameters:

--help	Show this message
--debug	Include debugging output
--old	Old domain
--new	New domain
--dir	Run domain change in this directory only. You must specify the full path.
	Use this feature to do a subset of the data directory. This instructs the script to do a recurse in the specified directory looking for the old `domain_name` and replacing it with the new `domain_name`..
	☞ **Note:** Without this option, only HTML, text, and VM files are modified.
--threadlimit	Defines the maximum number of simultaneous threads that can be invoked by this program. The default value is '50'

domain_change_fs.pl

The `domain_change_fs.pl` script handles all the steps required to change the domain name under the file system data directory. It does not change database contents.

Overview

By default, only .s?html?(.[a-z][a-z])?(,v)?, .txt(,v)?, and .vm(,v)? files are modified, unless `--dir` is specified.

☞ **Note:** Changing the domain through any other mechanism may cause problems.

Usage

Execute this script with a command like this:

```
[RUNTIME_DIR]/domain_change_fs.pl [--debug] [--dir] --old={domain_name} --new={domain_name}
```

☞ **Note:** The new domain name must match the value defined for the DOMAIN token in the `site-options.conf` file.

Options

The `domain_change_fs.pl` script provides the following parameters:

--help	Show this message
--debug	Include debugging output
--old	Old domain
--new	New domain
--dir	Run domain change in this directory only. You must specify the full path.
	Use this feature to do a subset of the data directory. This instructs the script to do a recurse in the specified directory looking for the old `domain_name` and replacing it with the new `domain_name.`.
	☞ **Note:** Without this option, only HTML, text, and VM files are modified.
--threadlimit	Defines the maximum number of simultaneous threads that can be invoked by this program. The default value is '50'

environment_check.sh

The `environment_check.sh` script verifies whether all environment packages required for installing CollabNet TeamForge are present.

Overview

Use the `environment_check.sh` script to verify that you have all required environment packages for installing SourceForge.

You will be prompted to run the `install-missing-packages.sh` script if one or more required packages are missing from your system.

☞ **Note:** You may be prompted to manually remove any older packages identified on your system before running the `install-missing-packages.sh` script.

Usage

From the <INSTALLATION_SOURCE> directory, run this script as follows:

```
./environment_check.sh
```

etl-client.py

The etl-client.py script allows you to access the Extract, Transform and Load (ETL) scheduler and check the status of the jobs configured. The script also supports triggering jobs manually.

Parameters

The following parameters are available for the etl-client.py script:

**-s	--status**	Prints the status of all the jobs configured in the ETL service.
**-a	--status-all**	Prints the status of incremental and historical jobs configured in the ETL service.
-v| --verbose	Chronicles the process of requested operation a bit more.
-r| --run=	Triggers a job manually for a given job.

Data collection (ETL) jobs supported by the ETL service

While some ETL jobs are scheduled to run automatically, some must be triggered manually. The following table lists the available ETL jobs in TeamForge.

Job Category	Job Name	Description
Historical Data Collection Historical data collection jobs must be triggered manually. As a best practice, these jobs are run as part of post migration activities. Refer to the "Related Links" for more information.	SCMCommitInitialJob	Collects the historical commit data from TeamForge.
	TrackerInitialJob	Collects the historical data of artifacts from TeamForge.
	LoadFlexFields ☞ **Note:** This job must be executed if and only if you are upgrading from TeamForge 8.0 and earlier versions.	Collects the historical data of custom-defined artifacts from TeamForge.
Incremental Data Collection Incremental data collection jobs collect data that are added or modified incrementally on an ETL run-to-run basis. These jobs are scheduled to run automatically on a regular basis.	SCMCommitActivityJob	Collects the SCM commit data incrementally on an ETL run-to-run basis.
	TrackerIncrementalJob	Collects the tracker artifacts data incrementally on an ETL run-to-run basis.
	UserLoginActivityJob	Collects the user logon activity data incrementally on an ETL run-to-run basis.
Imported Data Collection (Simbel) TeamForge supports bulk data import through Simbel. This job collects Simbel-imported data. This job must be triggered manually post data import into TeamForge.	SimbelImportJob	Collects all Simbel-imported data such as the user logon activity, SCM commit and tracker artifacts data.

fix_data_permission.sh

The `fix_data_permission.sh` script is used to fix the TeamForge filesystem permissions when you backup and restore the TeamForge data in a new box during TeamForge upgrade.

Usage

Run this script as follows:

```
fix_data_permissons.sh [-r <repo_thread_count>>] [-d <dir_thread_count>] [-t
<fix_permission item>]
```

Example

Run the following command to fix the permission of GIT repositories.

```
./fix_data_permissons.sh -t git
```

Run the following command to fix the permission of GIT repositories and app directories.

```
./fix_data_permissons.sh -t git -t app
```

Run the following command to fix the permission of 15 Subversion repositories simultaneously.

```
./fix_data_permissons.sh -r 15 -t svn
```

Options

The following options are available for the `fix_data_permission.sh` script:

-r	Number of repos to process at a time. Default=5
-d	Number of directories to process at a time. Default=<No of processors>
-t	Item to fix. Possible values are [app, http, svn, cvs, git]. Default=<all items>
-h | --help	Prints the help information.

install.sh

The `install.sh` script handles all operations related to installing and removing the application.

Options
-a | --all

> Performs these operations in sequence - Install, create runtime and setup initscripts (equivalent to: -I -r -i)

-b | --bootstrap

> Sets up initial site data.

-C | --cleanup

> Stop/Kill the application processes, wipes out application packages and site directory.

> ⚠ **Caution:** THIS OPTION WILL WIPE OUT THE SITE DATA.

-c | --conf-file

> The environment configuration file (usually of the form environment-<platform>.conf), used to identify the platform.

> 👉 **Note:** This is not to be confused with the the `site-options.conf` file.

-d | --directory type='str', argname='installation_base_dir',

Path where the site would get installed.

-E | --check_environment

Check if the system environment is suitable for this installation.

-F | --force

Force the operation that is performed wherever appropriate (e.g. install/uninstall).

-f | --siteoptionsfile

Path to the site configuration file (default: ./conf/site-options.conf).

-I | --initscript

Start application services on reboot.

-i | --install

Install application packages.

-n | --noninteractive

Used to run the installer in non-interactive mode.

-R | --internalruntime

Create the internal runtime instance using the site configuration file.

-r | --runtime

Create the runtime instance using the site configuration file. (Also does configuration for Apache and PostgreSQL.).

-S | --startnow

Start application services after completing the other operations (if possible).

-s | --startup

Start application services on reboot and start it now: equivalent to -I -S.

-u | --uninstall

Uninstall application packages.

-V | --verbose

Show all output in noninteractive mode.

pbl.py

The `pbl.py` utility enables you to upload files to the Project Build Library and perform various operations on them.

Options

`--help|-h`	Print out a help message and exit.
`--api-user|-u username`	Your TeamForge Lab Management login name. Required for all upload operations.
`--api-key|-k key`	Your TeamForge Lab Management API key. Required for all upload operations.
`--api-url|-l url`	The URL to the TeamForge Lab Management API's. Will generally be `https://$external_host/cubit_api/1`. Required for all upload operations.
`--comment|-c "your comment"`	Print out a comment on this operation. The comment is always optional. The comment string will be logged in the audit log, but is not recorded in the PBL. For example, if you are deleting some files, you might want to use a comment to explain why you were deleting those files, for future auditing purposes.

--verbose|-v	Print out more detail on what the pbl.py is doing.
--xml-server-output|--xml	If this option is not specified, the pbl.py client reads in the XML returned from the server and presents the results to you in nicely formatted text. If you'd like to instead see the raw XML returned from the server, select this option.
--no-auth-cache	As a convenience, the pbl.py function caches the value of the --api-user and --api-key parameters in your home directory, in a subdirectory named .cubit, the first time a successful authentication is performed against the server. This is analogous to the Subversion client's use of the .subversion directory to store authentication credentials. Selecting the --no-auth-cache option turns off this caching.
--project|p projname	The TeamForge Lab Management project in which the file you are operating on is located.
--type|t {pub|priv}	The visibility type of the file, either pub (the file is in the public area of the PBL) or priv (the file is in the private area of the PBL).
--remotepath|r path	The remote path on the server, excluding the base directory, the project, and the visibility type. Examples are below.

password_util.sh

The password_util.sh script is used to get the encrypted or decrypted password value for the user scmviewer.

Usage

To encrypt:

```
sudo /opt/collabnet/teamforge/runtime/scripts/password_util.sh -encrypt
'teamforge'
```

```
[root@xx scripts]# ./password_util.sh -encrypt 'teamforge'
Input String:teamforge
Encrypted password:VBxJJvzbXb5tNx2SxR26egA==
```

To decrypt:

```
sudo /opt/collabnet/teamforge/runtime/scripts/password_util.sh -decrypt
'VBxJJvzbXb5tNx2SxR26egA=='
```

```
[root@xx scripts]# ./password_util.sh -decrypt 'VBxJJvzbXb5tNx2SxR26egA=='
Input String:VBxJJvzbXb5tNx2SxR26egA==
Decrypted password:teamforge
```

post-install.py

Run the post-install.py script to perform TeamForge post installation/upgrade actions. For example, run this script for setting up CLI reports, integrating Gerrit SCM with TeamForge, integrating SVN and CVS, and so on. This script is intelligent enough to identify the list of post installation actions required based on the TeamForge applications that are configured and running locally.

post-install.py [-D|--debug] [-c|--component <teamforge component>] [-r|--reconfigure] [-g |--group]

Option	Description
-D|--debug	Use this option to enable debug mode. Enabling the debug mode displays all the back-end debug messages on the console.
-c|--component	Use this option to run the post installation action for a specific TeamForge component. By default, if this parameter is not specified, all the post installation/upgrade actions for all relevant TeamForge components running on the server are performed. Valid values: app, etl, database, datamart, subversion, cvs, gerrit and codesearch.
-g |--group	Run post installation for a specific application group. Valid values: clireports, reporting, codesearch, scm, projecttemplates and fixpermissions.
-r|--reconfigure	To reconfigure the post installation with default credentials.

-O |--option	The `post-install.py` script prompts for input for items such as TeamForge admin user name, password, etc. Use this option to pass such parameters as command line arguments so that you are not prompted for the same later.
	
	Valid values:
	
	`teamforge-tf.siteadmin.username=<USERNAME>`
	`teamforge-tf.siteadmin.passwd.secure=<PASSWORD>`
	`gerrit-tf.gerritdb.passwd.secure=<PASSWORD>`

-h|--help	Use this option to display help information about post-install.py.

Usage examples

- Run the following command with no arguments to run the all post installation actions based on TeamForge applications that are configured in the current server.

  ```
  ./post-install.py
  ```
- Run the following command to run post installation actions specific to Black Duck Code Sight.

  ```
  ./post-install.py -c codesearch
  ```
- Run the following command to reconfigure Gerrit integration.

  ```
  ./post-install.py -c gerrit -r
  ```
- Run the following command to run post installation actions specific to SCM integrations such as Subversion, CVS or GIT relevant to the context.

  ```
  ./post-install.py -g scm
  ```

Manually create SCM integration in case your integration server uses self-signed (or wildcard) certificates

If your integration server uses self-signed (or wildcard) certificates, the *post-install.py* script may fail to create SCM integration. In such case, you may have to manually create the SCM integration using the TeamForge **Admin** > **Integrations** > **SCM Integrations** UI. The following message is shown when you run the post-install.py script on integration servers that use self-signed (or wildcard) certificates.

```
Please create SVN integration through TeamForge Admin UI if your integration
server uses self signed certificate. Failed.
```

Troubleshooting

See *Post install fails for GIT. What should I do?* on page 921 for more information.

psql-wrapper
The psql-wrapper script is used to connect to the TeamForge application.

Usage

Run this script as below:

```
sudo [RUNTIME_DIR]/scripts/psql-wrapper
```

Comments

• Run this script as a sudo user.

• Run this script with the postgres backend.

• You have full write access to the database for executing queries.

☞   **Note:** This script is not supported in the Oracle backend.

psql-reporting-wrapper
The psql-reporting-wrapper script is used to connect to the datamart.

Usage

Run this script as below:

```
sudo [RUNTIME_DIR]/scripts/psql-reporting-wrapper
```

Comments

• Run this script as a sudo user.

• Run this script with the postgres backend.

• You have full write access to the datamart for executing queries.

☞   **Note:** This script is not supported in the Oracle backend.

restore-data.py
The restore-data.py script restores the compressed data from the named source directory and deletes any existing data. By default, the Teamforge and the reporting database are backed up to the destination directory. If reporting is disabled, only the Teamforge database is backed up.

Overview

restore-data.py finds and unpacks these data resources:

- Subversion repositories
- CVS repositories (if any)
- The data directory (/var)
- The SourceForge database.

Usage

Run this script as follows:

```
./restore-data.py --source=<directory name>
```

where <directory-name> is the directory to which you backed up the data with the backup-data.py script.

Location

```
/opt/collabnet/teamforge/runtime/scripts/
```

Options
--source

>Directory where the compressed copy of the site's data is available.

--help | -h

>Print this usage message and exit.

restore-rb-data.py
The restore-rb-data.py script is used to restore the Review Board application data from the backup directory.

Overview

This script removes the existing Review Board application data present in the system and restores data from the backup directory.

Usage

Run this script as follows:

```
python ./restore-rb-data.py -backupdir={dir}
```

Options

The following options are available for the restore-rb-data.py script:

-b | --both	Restore both the database and the filesystem. This is the default option.
-d | --database	Restore the database.
-f | --files	Restore the filesystem.
-h | --help	Provides a list of all available options for this script.

SearchReindex.py
The SearchReindex.py script allows you to reindex the entire TeamForge data.

Overview

You can use this script to reindex the entire TeamForge data or you can choose to reindex the subset of data types.

Usage

Run this script as follows:

```
./SearchReindex.py --<component name>
```

Example

To perform a search reindex for the tracker, run this command:

```
./SearchReindex.py --trackers-only
```

To perform a search reindex for the wiki, run this command:

```
./SearchReindex.py --wiki-only
```

To perform a search reindex for documents run this command:

```
./SearchReindex.py --documents-only
```

Options

--single-item itemId, | -i	Schedules a re-index for just the given item. If the item id is for a project the scheduling results in the server re-indexing all of the project data.
---force-index | -f	Force indexing (doesn't check if item is searchable already).
--artifacts only | -a	Reindex all artifacts on the site that are currently not searchable or all artifacts if option f is selected.
--documents only | -d	Reindex all documents on the site that are currently not searchable or all artifacts if option f is selected.
---posts only	Reindex all posts on the site that are currently not searchable or all artifacts if option f is selected.
---trackers only	Reindex all trackers on the site.
---document_folders-only	Reindex all document folders on the site.
---topics-only	Reindex all topics.
---forums-only	Reindex all forums on the site.
---news-only	Reindex all news.
---project_pages-only	Reindex all project pages.
---packages	Reindex all packages.
---commits-only	Reindex all commits.
---frs_files-only	Reindex all frs files.
---releases-only	Reindex all releases.
---wikis-only	Reindex all wikis.

`--project-id projectID | -p`	Limit the re-indexing to data for single project when re-indexing only artifacts and or documents.
`--verify | -x`	Searches for each item that is scheduled for re-indexing. There is a one minute wait limit for each item to be re-indexed by the server.
`--dryrun | -r`	Executes all the steps for scheduling a re-index without actually sending any re-index requests to the server. This provides a list of items that need re-indexing.
`output-file filePath, | -o`	Prints the output for the given file.
`--verbose | -v`	Chronicles the process of scheduling the re-index a bit more.

set_auth_key.py

The `set_auth_key.py` script sets the authorized key to the scmviewer user profile.

Overview

The script accepts the authorized key file, reads the key from the file and sets it to the scmviewer user profile.

```
$ sudo /opt/collabnet/teamfoge/runtime/scripts/codesearch/set_auth_key.py --
help
```

Usage

Run this script as follows:

```
./set_auth_key.py --authkey-file=<path_of_authkey_file>
```

When you run the script, you will be prompted to enter the TeamForge site-admin credentials to update the key for the scmviewer user.

set-reports-readonly-user-permission.py

This script grants read-only access to the datamart for those users specified by the REPORTS_DATABASE_READ_ONLY_USER token.

☞ **Note:** This script is executed automatically when runtime is created, after bootstrap or migration is completed.

Usage

You can run this script from the command line as follows:

```
sudo runtime/scripts/set-reports-readonly-user-permission.py
```

Comment

You can use this script only in advanced mode.

snapshot.py

Use this script as a debugging tool to troubleshoot system errors. It records a snapshot of the current state of the machine.

Overview

Run this script manually to generate debugging information before restarting the instance.

Usage

Run this script as follows:

```
/opt/collabnet/teamforge/runtime/scripts/snapshot.py
```

Options

The following options are available for the snapshot.py script:

-h|--help	Provides information on using the script.
--extra	An arbitrary command whose output should be placed in the generated log file. For example, you can have snapshot.py execute the lsof command like this:

```
--extra '/usr/sbin/lsof -n -P -b -i
 -U'
```

Enclose commands with options in quotes.

-v|--verbose	Provides output on the actions performed by the script.

☞ **Note:** The output from snapshot.py is written to a log file in the [LOG_DIR]/runtime directory. Use the output (snapshot.log) to troubleshoot any system or CollabNet related errors.

Cluster location

☞ **Important:** The snapshot.py script generates a log file for the node on which it is run. When a CollabNet site is deployed on a cluster and you need information to troubleshoot problems, it is recommended that you run this script on all the nodes.

sync-orchestrate.py

Run the sync-orchestrate.py script to synchronize a TeamForge project with Orchestrate by triggering the world-state message for the given project.

Usage

Use the following command to run the this script.

```
[RUNTIME_DIR]/scripts/sync-orchestrate.py --projectId=<proj1234> [[--
tfuser=<username>] [--tfpasswd=<password>]]
```

Parameters

Paramater	Description
--projectId	The TeamForge project ID.
-u | --tfuser	The TeamForge site administrator user name.
-p | --tfpasswd	The password for TeamForge user.
-h | --help	To view help information for the script.

upgrade-site.sh

With this script, you can perform a cumulative patch upgrade or downgrade on a running instance.

Overview

This is a wrapper for the upgrade.py script.

The script verifies the following:

- The user invoking the script is the equivalent of a root user.
- The specified directory has a valid SourceForge installation.

It performs the following actions depending on the options specified:

- Displays a summary of what would happen during the patch installation.
- Downgrades or upgrades the site to the specified patch level.
- Reverts the site to the previous patch level it was at, before the current patch was applied.
- Downgrades the patch level on the site by one.
- Starts SourceForge after successfully installing the patch.
- Allows a test "dry run" of the patch installation.

Usage

Run this script as follows:

```
./upgrade-site.sh -d <INSTALL_DIR> [-r] [-u] [-t] [-l level] [-f file] [-n]
[-h] [-V] [v]
```

Example

To perform a component upgrade from a base SourceForge installation (patch level 0) to patch level 2, use this command:

```
sudo ./upgrade-site.sh -t -d /opt/collabnet/teamforge -l 2
```

Options

-f [manifest] | --file [manifest]	The manifest file with the appropriate information for this upgrade.
-d [INSTALLATION_DIR] | --directory [INSTALLATION_DIR]	The directory where the application is installed.
	☞ **Note:** This option is required.
-r | --rollback	Rolls back the previous (most recently applied) patch. For example, if you upgrade the site from patch level 1 to patch level 4, and then run `upgrade-site.sh` with this option, the resulting patch level on the site is patch 1.
-l [level] | --level [level]	The patch level to which the SourceForge site must be upgraded (or downgraded).
-V | --verbose	Displays script output including traceback errors. If this option is not used, the script displays error messages but not the actual traceback errors.
-v | --version	Displays the script version.
-n | --noninteractive	Non-interactive mode.
-t | --testrun	Displays a summary of the actions that will be performed as part of the upgrade or downgrade. Use this option to view a description of what would take place during a patch upgrade (or downgrade) before you actually apply the patch.
	☞ **Note:** You must use this option along with the `l`, `r`, `u`, or `f` options.

**-u	--uninstall**	Decrements the patch level on the site by one. For example, if you upgrade the site from patch level 1 to patch level 4, and then run upgrade-site.sh with this option, the resulting patch level on the site is patch 3.
**-h	--help**	Prints usage information.

☞ **Note:** Do not use the following combinations of options in the same command:

- -u (uninstall) with -r (rollback)
- -f (manifest) with -l (level)
- any combination of -u, -r, -l, -f

If you do, the script exits with a corresponding error message.

projecttracker.py

The projecttracker.py utility provides a command line interface to control a Project Tracker integration on a TeamForge site.

Overview

You can run this script as the root user to start, stop and get the status of a Project Tracker instance that is integrated into your site.

Usage

Run this script as follows:

```
sudo /etc/init.d/projecttracker [command]
```

For example, use the following command to start the Project Tracker integration:

```
sudo ./projecttracker start
```

Commands

The following commands are available for the projecttracker.py script.

☞ **Note:** "Catalina" is the code name for the integrated Project Tracker feature.

debug -security	Debug Catalina with a security manager
jpda start	Start Catalina under JPDA debugger
run	Start Catalina in the current window
run -security	Start in the current window with security manager
start	Start Catalina in a separate window
start -security	Start in a separate window with security manager
stop	Stop Catalina
stop -force	Stop Catalina (followed by kill -KILL)
version	What version of tomcat are you running?
restart	Stop and Start Catalina

status	Indicate whether Catalina is running

wmt-wrapper.sh

The `wmt-wrapper.sh` script is used to invoke `wmt.java` by using `wmt.jar`. This script is a wrapper script for the `wmt.sh` script.

Overview

The wmt tool converts the CEE Moinmoin Wiki data to TeamForge JspWiki data. The `wmt-wrapper.sh` script is used to execute the wmt tool. For more information on WMTool, click *https://forge.collab.net/sf/wiki/do/viewPage/ projects.sf_engine/wiki/WMTool*

Usage

Run this script for all projects:

```
$ <WMT_BUILD_DIR>/wmt-wrapper.sh -a
```

Run this script for a particular project:

```
$ <WMT_BUILD_DIR>/wmt-wrapper.sh -s.
```

For detailed information on running this script, click *Run the wmt-wrapper.sh script*

Options

Options that are available for this script:

**-a	-- All projects**	Runs the script for all the projects.
**-s	-- Specific project**	Runs the script for specific projects by mentioning the project name.

Log files in TeamForge

System administrators can use logs to debug problems and ensure that the application is performing to expectations.

JBoss logs

The JBoss application server writes several different logs under the <SOURCEFORGE_INSTALL_DIR>/log directory.

boot.log	Lgs the JBoss startup and shut down notifications. This log is overwritten each time JBoss is (re)started.
localhost_access	The Records access to the application from a remote host, similar to the Apache access_log. This log is rotated each day, and the files have a date stamp appended to their name, such as localhost_access2004-11-26.log.
server.log	Logs all the activities of the application server, including any exceptions. This log is the best place to begin debugging CollabNet TeamForge server error exception ids (exid).
session-info.log	Records when new sessions are created. This log is overwritten each time JBoss is (re)started.
vamessages.log	Records CollabNet TeamForge -specific actions, including some SQL queries that are sent to the backend database. This log is rotated each time it reaches 100MB in size. When rotated the older files have a number

appended to the end, such as `vamessages.log.1` and `vamessages.log.2`.

Oracle logging

The most important Oracle log is the `alert` log, which is found in `$ORACLE_HOME/admin/$SID/bdump/alert_$SID.log`.

An Oracle database performs logging on a wide array of functionality. The majority of the logs that are generated are stored under `$ORACLE_HOME/admin/$SID/`. Many logs are stored under this directory hierarchy, but `alert` is the most important. This log records all database activity, including serious problems.

The `alert` log is not rotated or overwritten, and can become quite large over time, especially on an active database.

Additional logs are created under the same directory hierarchy, for specific incidents. If a problem is recorded in the alert log, the other logs should be inspected for additional details.

For more information, as well as support in the maintenance of an Oracle database, contact Oracle Support or Oracle's *Metalink* site.

SCM (CVS and Subversion) logs

Software configuration management (SCM) servers generate several logs from the CollabNet TeamForge ; however, in the interest of completeness they are all documented here.

catalina.out	This log contains information on the startup and runtime activities of the Tomcat server. This log is not rotated, nor is it overwritten, and is appended continuously over the lifetime of the server.
localhost_log	This log contains a record of CVS or Subversion browsing URL construction. When a user attempts to browse a CVS or Subversion repository in his or her web browser, the URL construction process is documented in this log. This log is rotated for each date that there is activity.
localhost_admin_log	This log contains a record of the initial startup and deployment of the managed integration server. A new date stamped log is generated each time the integration server is started.
vaexternalintegration.log	This log contains information on the operations that are being executed by the managed integration server. This log is stored in <SOURCEFORGE_INSTALL_DIR>/log.

Email logs

Both the CollabNet TeamForge email and search backends are managed from a parent daemon known as Phoenix. If the mail backend is not operating properly, the first troubleshooting step is to check the phoenix.log to see if it encountered difficulties starting up.

Overview

The Phoenix daemon logs its activities to the `phoenix.log` file, which is stored under `SOURCEFORGE_INSTALL_DIR/james/james-2.2.0/logs`. This log is overwritten each time Phoenix is (re)started. Phoenix is run as part of the CollabNet TeamForge standalone server init script.

CollabNet TeamForge email is handled by the `JAMES` server. `JAMES` logs all of its activities under `SOURCEFORGE_INSTALL_DIR/james/james-2.2.0/apps/james/logs`. A new log is created for each

date when there is activity, and additional logs are created if james is restarted on a date when there is activity. The date is embedded in the log name (such as james-2005-04-28-01-00.log).

Active logs

Sixteen different logs are created by `james` for different components of its functionality. This topic describes only the ones that are used actively by CollabNet TeamForge .

`james-$date.log`	The James log records the overall mail handling behavior of the James server.
`mailet-$date.log`	The mailet log records how each piece of email is handled. If there is a mail delivery problem, this log is the best place to begin investigation.
`mailstore-$date.log`	The mailstore log records the behavior of mail spools, and the storage of mail. This log should normally not contain errors unless James is unable to write or read mail to or from the file system.
`smtpserver-$date.log`	The smtpserver log records all inbound mail handling results. If email to discussion forums is not posting, or is getting rejected, this log would be the best place to begin investigation.
`spoolmanager-$date.log`	The spoolmanager log records the processing of mail spools. This log could be of value in troubleshooting mail delivery or handling problems.

Search logs

Both the CollabNet TeamForge search and email backends are managed from a parent daemon known as Phoenix. If the search backend is not operating properly, the first troubleshooting step is to check the `phoenix.log` file to see if it encountered difficulties starting up.

The Phoenix daemon logs its activities to the `phoenix.log` file, which is stored under `SOURCEFORGE_INSTALL_DIR/james/james-2.2.0/logs`. This log is overwritten each time Phoenix is (re)started.

Phoenix is run as part of the CollabNet TeamForge standalone server init script.

Once started successfully, the search server waits for new content to be indexed or searches to be performed. The search server logs its activities under `SOURCEFORGE_INSTALL_DIR/james/james-2.2.0/apps/search/logs`. The logs that are created are all named `default` with the date stamp appended to them (such as `default-20041126.log`). A new log is created for each date that there is indexing activity.

If the search server is not running, or expected search results are not being provided, the default log is the best place to investigate further.

Project Build Library audit log

You can use this screen to view the complete list of actions performed in the Project Build Library.

Contents

Information about the following types of actions is displayed in this screen:

- Change a File Description
- Create a Directory
- Delete a File or Directory
- Download a File
- Move a File or Directory
- Upload a File

☞ **Note:** The value displayed in the **Event** field is the value passed in the --comment parameter from the Project Build Library client.

Getting there

On the project home page, click **Build Library** in the left navigation bar and select the **Audit Log** tab.

Access

This screen is accessible for all users who have at least the view permission for the project.

Profile audit log

Use this screen to view the complete list of actions performed on a profile.

Getting there

On the **Profile Library** screen, click the **Audit Log** tab.

Access

This screen is accessible for all users who have at least the view permission for the project to which the profile is allowed.

x

Example

When a user updates any of the profile fields on the **Profile Admin** screen, the following details are displayed in this screen:

- The old value for the field.
- The new value for the field.
- The name of user who updated the field.
- The time when the change occurred.

User Audit Log

You can use this screen to view the list of actions performed by the user in the TeamForge Lab Management system.

For example, when a user logs into the web interface of the TeamForge Lab Management system, the event is displayed in this screen.

Access

This screen is accessible to all users who have at least the Domain Administrator role.

Getting there

On the **Administration** tab, click **User Audit Logs** in the left navigation bar.

Host audit log

You can use this screen to view the complete list of actions performed on a host.

Getting there

On the TeamForge Lab Management Host home page, click the **Audit log** tab.

Access

This screen is accessible for all users who have at least the view permission for the project to which the host is assigned.

Example

When the IP address for the host is changed, the following details are displayed in this screen:

- The old IP address.
- The new IP address.
- The name of user who changed the IP address.
- The time when the change occurred.

Project audit log

The **Project audit log** screen shows the complete list of changes applied to a project.

Getting there

On the TeamForge Lab Management Project home page, click **Audit Logs** in the left navigation bar.

Access

This screen is accessible for all users who have at least the view permission for the project.

Example

When a profile is added to the list of buildable profiles for this project, the following information appears on this screen:

- The action that was taken.
- The user who performed the change.
- The time when this change occurred.

etl.log

This file contains information from extract-transform-load runs, including data transformation warnings and errors.

> ☞ **Note:** Transformation errors do not constitute a failed ETL run. For example, if a corrupt row of data in one of the source tables causes transformation errors, this is treated as a "skipped record" and gets logged.

Configuration files in TeamForge

Edit these configuration files to get the behavior you want.

site-options.conf

CollabNet TeamForge is controlled by settings in a master configuration file called `site-options.conf`. Some of the most useful configuration settings you can specify in the `site-options.conf` file are described here.

The `site-options.conf` file resides in the installer directory. (The default installer directory is `/opt/collabnet/teamforge-installer/8.2.0.0/conf`.)

When TeamForge starts, it reads and implements the instructions provided as values to the variables in this file.

The basic procedure for configuring your TeamForge site, then, is to edit the `site-options.conf` file, supply a valid value for the variable of interest, save the file, and restart the TeamForge runtime environment.

TeamForge has a number of configuration files that can be used to improve site performance. You can configure the `site-options-small.conf`, `site-options-medium.conf` and `site-options-large.conf` files to suit the needs of small, medium and large systems. Each of these sample configuration files has information about the load it supports and the hardware (CPU and memory) required.

ACTIVITY_LINKS_CUSTOMIZATION

When the *ACTIVITY_LINKS_CUSTOMIZATION* variable is set to true, the TeamForge Activity Chart and the Most Active Projects List do not appear on the main page before the user logs in.

Values

true or false

Default

false

ADMIN_EMAIL

The *ADMIN_EMAIL* variable specifies a valid email address for the site administrator.

The mail account specified must be hosted on a separate server from the TeamForge site server.

The *SYSTEM_EMAIL*, *ADMIN_EMAIL*, and *JAMES_POSTMASTER_EMAIL* variables can specify the same address.

> ☞ **Important:** In TeamForge 6.x, the sender name and address for system-generated emails is taken from the value of the *SYSTEM_EMAIL* variable. Therefore, changing the admin user's full name or email address does not affect the sender details of system-generated emails. This is different from TeamForge 5.x, in which the sender name and address for system-generated emails is derived from the admin user's full name and email address.

Values

Email address specification

Default

root@{__APPLICATION_HOST__}

ALLOW_NO_PASSWORD_ON_USER_CREATION

The *ALLOW_NO_PASSWORD_ON_USER_CREATION*, when set to true, allows the admin to create users with null password through SOAP when external authentication is enabled.

Values

true, false

Default

false

ALLOW_USERNAME_IN_PASSWORD

The *ALLOW_USERNAME_IN_PASSWORD*, when set to true, allows users to set a password that includes the string that they use for their user name on the site.

Values

true, false

Default

true

APPLICATION_LOG_DIR

The *APPLICATION_LOG_DIR* specifies the directory to which the application writes its log files.

Values

Path specification

Default

/opt/collabnet/teamforge/log/apps

APPROVE_NEW_USER_ACCOUNTS

The *APPROVE_NEW_USER_ACCOUNTS* variable specifies whether a site administrator must approve the requests to join the site.

Values

true or false

Default

true

ARTIFACT_DESC_EDITOR

The *ARTIFACT_DESC_EDITOR* variable allows you to choose the type of text that can be used for artifact description using the editor tool.

Values

Plain Text

Default

Plain Text

ARTIFACT_LIST_LIMIT

The *ARTIFACT_LIST_LIMIT* variable specifies the maximum number of artifacts displayed in and exported from the **Planned Tracker Artifacts** tab available in **File Releases**.

Values

Integer

Default

5000

AUTO_DATA

TeamForge 7.1 and later support automatic password creation. Once you turn on automatic password creation, the *AUTO_DATA* token is auto-generated by the installer during runtime recreation.

☞ **Restriction:** The *AUTO_DATA* token is not a user-editable site options token. Do not modify this token manually.

The following password-related `site-options.conf` tokens can have the passwords automatically created if set to **$auto$**. When set to **$auto$**, the passwords for the tokens are randomly generated, encrypted and stored in the *AUTO_DATA* token that is added automatically to the `site-options.conf` file during runtime recreation.

```
DATABASE_PASSWORD=$auto$
```

```
DATABASE_READ_ONLY_PASSWORD=$auto$
```

```
REPORTS_DATABASE_PASSWORD=$auto$
```

```
REPORTS_DATABASE_READ_ONLY_PASSWORD=$auto$
```

```
ETL_SOAP_SHARED_SECRET=$auto$
```

```
JAMES_ADMIN_PASSWORD=$auto$
```

```
BDCS_ADMIN_PASSWORD=$auto$
```

```
MIRROR_DATABASE_PASSWORD=$auto$ (applicable only if you are mirroring your
 database)
```

⚠️ **Caution:**

As this *AUTO_DATA* token is auto-generated and managed by the TeamForge installer, do not modify this token manually during TeamForge upgrades. When you upgrade TeamForge, you must copy this token intact (along with its value) from the old `site-options.conf` file to the upgraded site's `site-options.conf` file, before recreating the runtime. This applies to all the servers in a distributed set up (copy the *AUTO_DATA* token and its value to all the servers).

This feature is enabled by default. You can, however, override any of the above password-related tokens with the password of your choice.

BCC_MAIL_BATCH_SIZE

Set ENABLE_BCC_MONITORING token to true and then depending on your site's requirements, you can set the value of this BCC_MAIL_BATCH_SIZE token to configure the number of monitoring emails delivered in a single delivery.

When a work item such as an artifact or discussion is updated, instead of sending separate monitoring emails to all monitoring users, you can now choose to have just one monitoring email sent with all monitoring users added to the BCC. This reduces the load on the email server and results in faster email delivery. You must enable this feature by setting *ENABLE_BCC_MONITORING* to true and then, depending on your site's requirements, you can optimize the value of this BCC_MAIL_BATCH_SIZE token to increase or decrease the number of emails delivered in a single delivery.

Values

Integer

Default

100

This token was added in TeamForge 7.2.

BDCS_ADMIN_PASSWORD

The *BDCS_ADMIN_PASSWORD* variable specifies the password to be set in the Black Duck Code Sight application.

Values

15 Bytes, case-sensitive, alphanumeric, '_!@&*%'

Default

blackduck

BDCS_ADMIN_USERNAME

The *BDCS_ADMIN_USERNAME* variable specifies the admin username to be created in the Black Duck Code Sight application.

Values

20 bytes, case-sensitive, alphanumeric, '_'. should begin with alpha

Default

sysadmin

BINARY_SETUP_TYPE

Set this token appropriately to have the binary application installed the way you want (as part of the TeamForge installation or upgrade).

TeamForge 8.0 and later support integration with Nexus OSS, an open source repository manager for binary artifacts. By default, the TeamForge installer installs a binary application (referred to as the 'binary app' hereinafter), which is essentially a launching pad for all binary integrated applications such as Nexus OSS. Once this binary app is installed as part of TeamForge 8.0 installation, you can integrate your Nexus servers and repositories with TeamForge.

Though the binary app is installed by default with TeamForge 8.0, you can change the value of this site option token, *BINARY_SETUP_TYPE*, to skip binary app installation altogether. You can also have this token configured to have the binary app installed and rolled out for all projects (both existing and new projects to be created) or only for new projects to be created or for select projects on a need basis.

Values

- all: Binary app is installed and available for all projects.
- new: Binary app is installed and available for new projects only.
- manual: Binary app is installed at a site level and project administrators can add it to select projects on a need basis.
- none: TeamForge installer skips the binary app installation altogether.

Default

new

This token was added in TeamForge 8.0.

DATABASE_HOST

The *DATABASE_HOST* variable is a special case of the *HOST* variable that specifies the host name of the server where the database is running.

Values

Hostname specification

Default

None

Comments

The database host is specified by adding `database` to an existing *HOST_localhost* property, or adding a *HOST_localhost* property if it is not already there.

Example

This site has the database running on the same box as all other services:

```
HOST_localhost=app cvs subversion database
```

This site has the database running on its own separate box:

```
HOST_localhost=app cvs subversion
HOST_<database_host>=database
```

DATABASE_NAME
The *DATABASE_NAME* variable specifies the name of the site's database.

Values

Alphanumeric string

Default

ctfdb

DATABASE_PASSWORD
The *DATABASE_PASSWORD* variable is the password for the Unix user that is authorized to read from and write to the site's database.

Values

Alphanumeric string

Default

ctfpwd

DATABASE_TYPE
The *DATABASE_TYPE* variable specifies the type of database in which the TeamForge site's data is stored.

Values

postgresql or oracle

Default

postgresql

DATABASE_USERNAME
The *DATABASE_USERNAME* variable specifies the Unix user that is authorized to read from and write to the site's database.

Values

Alphanumeric string

Default

ctfrptuser

Comments

For some advanced operations, you may need to log into the database as the database user. However, under normal conditions only the TeamForge site process itself needs to access the database.

DEDICATED_INSTALL

If the *DEDICATED_INSTALL* variable is set to true, the TeamForge site is installed automatically, with the default configuration and minimal user intervention.

Values

true or false

Default

true

Comment

The dedicated install option is appropriate for TeamForge sites where:

* All services (the TeamForge application, the database, and Subversion) run on a single box.
* No other services run on the TeamForge box.

DEFAULT_LOCALE

The *DEFAULT_LOCALE* variable specifies the language in which automated email messages from the site are generated.

Values

Default

en

DEFAULT_PROJECT_ACCESS

The *DEFAULT_PROJECT_ACCESS* variable specifies the type of access that is assigned to a project when it is created. A project can be private, public, or gated.

Values

private, gated, public

Default

private

DISABLE_CREATE_INTEGRATION_SERVERS

The *DISABLE_CREATE_INTEGRATION_SERVERS* token specifies whether the creation of new SCM integrations is allowed.

Values

true or false

Default

false

Comments

When this token is set to its default value of "false", you can add SCM integration servers to your TeamForge site. Also, the **Discover Subversion Edge Servers** option, which enables you to find and connect to Subversion Edge servers on your LAN, is available.

DISABLE_USER_SELF_CREATION

The *DISABLE_USER_SELF_CREATION*variable restricts users from creating their own accounts on the TeamForge home page.

Values

true or false

Default

true

DISCUSSION_ADD_HEADERS

The *DISCUSSION_ADD_HEADERS* variable allows you to add custom headers to the emails posted in the forum.

Values

You can choose to add or remove headers by specifying the particular information you want to be added or dropped from the header. For example, if you add <#d#> in the Add header field, the URL of that discussion will be added to the header of all the available messages in that discussion.

Default

None

Example

DISCUSSION_ADD_HEADERS=headername1:value1, name2: value2 , post-id:<#n#>, forum-url:<#d#>, message-url:<#m#>, domain:<#h#>, list-name:<#l#>, list-address:<#l#>@<#h#>

Comments

Add one or more header names. The match of any of these headers in an outgoing message (via email) causes its addition with appropriate notification to the posting user.

DISCUSSION_DROP_MIME_TYPES

The *DISCUSSION_DROP_MIME_TYPES* variable allows you to delete the mime types submitted by email that contain arbitrary strings.

Values

image/jpeg,image/jpg,text/xml

Default

Regular expression

Example

DISCUSSION_DROP_MIME_TYPES=image/jpeg,image/jpg,text/xml

Comments

Add one or more mime types to the Drop mime types filter. The presence of any of these mime types in an incoming message (via email)causes its deletion with appropriate notification to the posting user.

☞　**Note:** If a mime type is specified in both the Reject and Drop mime filters, then the Reject mime type filter must take higher precedence than the Drop mime type filter.

DISCUSSION_EMAIL_MONITORING

The *DISCUSSION_EMAIL_MONITORING* variable determines which users can monitor a forum on the site.

Values

Value	Description
0	Allow only forum admins.
1	Users with role permissions.
4	All logged in users.
5	Allow all site users and guests.

Default

1

Example

DISCUSSION_EMAIL_MONITORING=4

Comments

This setting applies to the site as a whole. Project owners can choose to be more restrictive in their own project by selecting a lower value on the project administration page.

DISCUSSION_EMAIL_POSTING

The *DISCUSSION_EMAIL_POSTING* variable determines which users on your site can post to forums by e-mail.

Values

Value	Description
0	Allow only forum admins.
1	Users with roles and permissions.
4	All logged in users.
5	Allow known email addresses only.
6	Allow all site users and guests.

Default

1

Example

DISCUSSION_EMAIL_POSTING=4

Comments

This setting applies to the site as a whole. Project owners can choose to be more restrictive in their own project by selecting a lower value on the project administration page.

DISCUSSION_FORUM_EDITOR

The *DISCUSSION_FORUM_EDITOR* variable allows you to choose the type of text that can be used in discussion forum description using the editor tool.

Values

Plain Text

Default

Plain Text

DISCUSSION_MAX_ATTACHMENT_SIZE

The *DISCUSSION_MAX_ATTACHMENT_SIZE* variable sets an upper limit to the size of files that users can attach to an email message sent to any discussion forum on the site.

Values

Integer (Megabytes)

Default

blank

Comment

A value of zero or less specifies that there is no limit, which is the same as the default behavior without the variable.

DISCUSSION_POST_EDITOR

The *DISCUSSION_POST_EDITOR* variable allows you to choose the type of text that can be used for posting in discussion forums using the editor tool.

Values

Plain Text

Default

Plain Text

DISCUSSION_REJECT_CONTENT

The *DISCUSSION_REJECT_CONTENT* variable allows you to block the discussion messages submitted by email that contain arbitrary strings.

Values

Regular expression

Default

None

Example
DISCUSSION_REJECT_CONTENT=(?s).*word.*,(?s).*spam.*

Comments

Add one or more entries. Each regular expression must match an entire entry. The match of any of these entries in the body or subject of an incoming message (via email) causes its rejection, with appropriate notification to the posting user.

☞ **Note:** The content entry is case sensitive.

DISCUSSION_REJECT_HEADERS

The *DISCUSSION_REJECT_HEADERS* variable allows you to block different headers submitted by email that contain arbitrary strings.

Values

Regular expression

Default

None

Example

DISCUSSION_REJECT_HEADERS=(?s).*headername1:value2.*,(?s).*name2:value2.*

Comments

Add one or more header names. Each regular expression must match an entire header name. The match of any of these headers in an incoming message (via email)causes its rejection, with appropriate notification to the posting user.

DISCUSSION_REJECT_MIME_TYPES

The *DISCUSSION_REJECT_MIME_TYPES* variable allows you to delete the mime types submitted by email that contain arbitrary strings.

Values

Application/PDF,text/xml

Default

Regular expression

Example

DISCUSSION_REJECT_MIME_TYPES=application/pdf,text/xml

Comments

Add one or more mime types to the Reject MIME types filter. The presence of any of these mime types in an incoming message (via email) will cause its deletion with appropriate notification to the posting user.

DISPLAY_TIMEZONE

The *DISPLAY_TIMEZONE* token, if set with a preferred time zone, takes precedence over the physical TeamForge server location's time zone and will be the default time zone that's displayed throughout the application. In other words, use this token to set a preferred time zone to display across the TeamForge application in case the TeamForge physical server and users are not on the same time zone.

Values

The ID for a time zone can be either a full name such as *America/Los_Angeles*, or a custom ID in the form GMT[+|-]hh[[:]mm] such as *GMT-08:00*. It can also be in the form of a three letter abbreviation such as *PST*.

Default

If set, this token overrides the default time zone of the TeamForge server. TeamForge uses the default time zone of the JVM otherwise.

DOCUMENT_MAX_FILE_UPLOAD_SIZE

The *DOCUMENT_MAX_FILE_UPLOAD_SIZE* variable sets an upper limit to the size of the documents that can be attached.

Values

Integer (Megabytes)

Default

blank

Comment

A value of zero specifies that there is no limit, which is the same as the default behavior without the variable.

DOCUMENT_TEXT_EDITOR

The *DOCUMENT_TEXT_EDITOR* variable allows you to choose the type of text that can be used for the document description using the editor tool.

Values

Plain Text

Default

Plain Text

DOMAIN_localhost

The *DOMAIN_localhost* variable links the URL at which users can access the TeamForge site to the hostname of the machine where the site is running.

Values

Host and domain specification

Default

None

Example

DOMAIN_appbox.supervillain.org=worlddomination.supervillain.org

Comments

The *DOMAIN_localhost* variable consists of two options. One option identifies the host name of the machine where the TeamForge application is running, and the other specifies the URL through which users will access the services running on that machine.

- To identify the host machine, replace the localhost portion of the *DOMAIN_localhost* variable with the hostname of the machine where the TeamForge application runs.
- To specify the URL, set the value of the variable to the publicly accessible domain name of the site.

☞ **Note:**

- The localhost portion of the *DOMAIN_localhost* variable must match the localhost portion of the *HOST_localhost* variable.
- If the site has its services distributed on multiple machines, the localhost portion of the variable must match the host to which the app option is assigned.

ENABLE_BCC_MONITORING

When a work item such as an artifact or discussion is updated, instead of sending separate monitoring emails to all monitoring users, you can now choose to have just one monitoring email sent with all monitoring users added to the BCC. This reduces the load on the email server and results in faster email delivery. Set ENABLE_BCC_MONITORING to true to enable this feature.

Values

true or false

Default

false

> ☞ **Tip:** Depending on your site's requirements, you can also optimize the value of *BCC_MAIL_BATCH_SIZE* to increase or decrease the number of emails delivered in a single delivery.

This token was added in TeamForge 7.2.

ENABLE_CACHING_WITH_MEMCACHED

Set the *ENABLE_CACHING_WITH_MEMCACHED* token to true if you want to cache SVN authentication and authorization information in Memcached.

Values

true or false

Default

Caching with Memcached is disabled by default in TeamForge.

You must also set up the following tokens if you set *ENABLE_CACHING_WITH_MEMCACHED* token to true:

- *MEMCACHED_SERVER_HOST* on page 797
- *MEMCACHED_SERVER_PORT* on page 798
- *MEMCACHED_SERVER_TTL* on page 798

See *Install Memcached* on page 571 for more information.

ENABLE_UI_FOR_CUSTOM_EVENT_HANDLERS

To support branding and customization changes, set the *ENABLE_UI_FOR_CUSTOM_EVENT_HANDLERS* variable to true.

Values

true or false

Default

true

ENFORCE_MINIMUM_USERNAME_LENGTH

The *ENFORCE_MINIMUM_USERNAME_LENGTH* variable determines the minimum length that can be set for usernames.

Values

0-31

Default

0

ETL_JAVA_OPTS

The *ETL_JAVA_OPTS* token specifies the memory settings for the Java virtual machine that supports the ETL (Extract Transform and Load) job.

Values

Java specifications

Default

-Xms160m -Xmx512m -server -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/tmp -verbose:gc -XX:+ PrintGCTimeStamps -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=600000 - Dsun.rmi.dgc.server.gcInterval=600000 - Djava.security.egd=file:/dev/urandom

ETL_JOB_THREAD_COUNT

The *ETL_JOB_THREAD_COUNT* variable specifies the number of Extract, Transform and Load (ETL) jobs that can be run simultaneously.

Values

1-100

Default

2

Comments

If you only have a few jobs to be triggered few times a day, then 1 thread is sufficient. If you have tens of thousands of jobs, that needs to be triggered every minute, then you probably require thread counts like 50 or 100 (this depends on the nature of the work that your jobs perform, and your systems resources).

ETL_JOB_TRIGGER_TIME

The *ETL_JOB_TRIGGER_TIME* variable specifies the time and date for recurrent Extract, Transform and Load (ETL)jobs.

Values

Cron expression.

Default

0 30 2 * * ?

Comments

This variable takes a cron expression for a value, and not an absolute time value. The default value evaluates to 2.30 a.m. local time. For help with cron expressions, see *Cron Trigger Tutorial*.

ETL_SOAP_SHARED_SECRET

The *ETL_SOAP_SHARED_SECRET* variable enables users to access site-wide reporting data via a SOAP client.

Values

String (possibly encrypted).

Default

mightyetlsoapsecret

FILTER_DROPDOWN_MAX_SELECTION

By default, the drop-down lists with multi-select feature let you select up to 10 filter values. However, you can set any value that suits your requirement for this *FILTER_DROPDOWN_MAX_SELECTION* token to increase or decrease the count.

Values

Any positive integer.

Default

10

This token was added in TeamForge 7.1

FORBIDDEN_PASSWORD

The *FORBIDDEN_PASSWORD* variable restricts specified words from being used as passwords.

Values

Comma-separated strings

Default

None

GERRIT_DATABASE_HOST

This is the Gerrit Postgres database host. The Gerrit configuration property, *database.hostname*, is derived from the value of *GERRIT_DATABASE_HOST* token in the `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of the post installation script.

Default

127.0.0.1

GERRIT_DATABASE_NAME

This refers the Gerrit database schema. The Gerrit configuration property, *database.database*, is derived from the value of *GERRIT_DATABASE_NAME* in `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of post installation script.

Default

reviewdb

GERRIT_DATABASE_USER

This is the PostgresDB role name that has access to the Gerrit database '*GERRIT_DATABASE_Name*'. The Gerrit configuration property, *database.username*, is derived from the value of *GERRIT_DATABASE_USER* in the `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of post installation script.

Default

gerrit

GERRIT_FORCE_HISTORY_PROTECTION

The *GERRIT_FORCE_HISTORY_PROTECTION* token determines whether the history protection feature of the TeamForge Git integration is enabled. If it is set to true, history protection is turned on for all repositories hosted on the Git integration server.

Values

true or false

Default

false

Comments

In TeamForge 7.0 (and later versions), this token is defined in the `runtime-option.conf` file to support the non-interactive installation of the Git integration. After Gerrit's post-install script is run, the value of this token is used to set the configuration property `forceHistoryProtection` in the `/opt/collabnet/gerrit/etc/gerrit.config` file.

GERRIT_GIT_PUSH_THRESHOLD

The *GERRIT_GIT_PUSH_THRESHOLD* token determines the maximum number of commits in a single Git push. If the limit exceeds, only a single commit object is created in the TeamForge.

Values

Any positive integer.

Default

30

GERRIT_GIT_REFRESH_PERIOD

The *GERRIT_GIT_REFRESH_PERIOD* token sets the interval in seconds after which Git Integration synchronizes all the repositories and all RBAC permission with TeamForge.

Values

Seconds.

Default

3600 Seconds.

GERRIT_REPLICATION_MODE

Use this site-options token to set the Git integration server as either 'master' or 'slave' server. In case you do not want replication (standalone mode) or you have only one primary source for repositories, set this token to master. On the other hand, if you have a master Git integration server and you want to replicate (mirror) its repositories on a secondary slave Git integration server, set this token to slave on the slave Git integration server.

Values

master or slave

Default

master

Comments

By default, in TeamForge 8.1 (and later), this token is set to master in the `runtime-options.conf` during runtime creation. As you cannot change the replication mode of a Git server after initial runtime creation, you have to set this to slave at the very beginning of your installation process in case you want to configure the server as a mirror of a 'master' Git server. It is not possible to have a Git master and slave configured on the same node, but you can have multiple masters and slaves in your TeamForge environment. Each slave belongs to exactly one master. Once

a replica server is set up, it is not possible to reassign it to a different master Git integration server at a later point in time.

GERRIT_REPLICATION_MASTER_EXTERNAL_SYSTEM_ID

If GERRIT_REPLICATION_MODE is set slave, this token specifies external system ID of the 'master' Git integration server.

Values

Alphanumeric string (exsy<number>) of a master Git integration server

Comments

This token is mandatory if GERRIT_REPLICATION_MODE is set to slave, without which the runtime creation shall fail. On the contrary, the runtime recreation will also fail if GERRIT_REPLICATION_MODE is set to master and this token is present in the `site-options.conf` file.

GERRIT_SMTP_SERVER

This is the hostname of the SMTP mail server for Gerrit. The Gerrit configuration property, *sendmail.smtpServer*, is derived from the value of *GERRIT_SMTP_SERVER* in the `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of post installation script.

Default

localhost

GERRIT_SYNCH_PORT

The Port over which TeamForgecommunicates to Gerrit. The Gerrit configuration property, *teamforge.apiPort*, is derived from the value of *GERRIT_SYNCH_PORT* in the `runtime-options.conf`. It can be overridden in `site-options.conf` and requires runtime creation with execution of post installation script.

Default

9081

HIGHCHARTS_EXPORT_REQUEST_MAX_WAIT

This token is used to set the number of milliseconds that the Highcharts web application has to wait for response from the phantomjs server before it times out.

TeamForge chart export feature requires a pool of phantomjs servers to be running in the application server that is managed by a Highcharts web application. The phantomjs server pool runs as a blocking queue.

This token is used to specify the number of milliseconds that the Highcharts web application has to wait before it times out. In other words, the phantomjs server is expected to respond to the Highcharts web application within the time limit (in milliseconds) set in HIGHCHARTS_EXPORT_REQUEST_MAX_WAIT.

Default value

The default value is 500 milliseconds.

This token was added in TeamForge 7.2.

HIGHCHARTS_EXPORT_REQUEST_POOL_SIZE

TeamForge chart export feature requires a pool of phantomjs servers to be running in the application server that is managed by a Highcharts web application. This token is used to specify the number of phantomjs servers to be running in the pool.

Default value

The default pool size is 10.

This token was added in TeamForge 7.2.

HOST_localhost

The *HOST* variable identifies which of the TeamForge site's services are to run on a given machine.

Values

One or more of: app, database, cvs, subversion, etl, datamart

Default

app database cvs subversion etl datamart

Comments

Multiple applications can be assigned to a host. List the applications in space-separated format.

☞ **Important:**
- app (the main TeamForge application) must be assigned to one host only.
- database (the PostgreSQL or Oracle application) must be assigned to one host only.

Examples

Here are some possible variations on the *HOST_* variable when configured on the box where the main TeamForge application is running.

- This host has been assigned the main application, the database, and the Subversion and CVS source control services:

```
HOST_localhost=app database subversion cvs
```
- This is a two-box setup with a database running on a separate machine:

```
HOST_localhost.example.com=app subversion cvs etl datamart
HOST_mydatabase.mydomain.net=database
```
- This is a three-box setup with the TeamForge application running on one machine, the database on another and the reporting services on a third:

```
HOST_localhost=app
HOST_mydatabase.mydomain.net=database
HOST_myreportingbox.mydomain.net=etl datamart
```
- This is a three-box setup with the TeamForge application and reporting engine on one machine, both databases on another machine, and the source control services on a third machine.

```
HOST_localhost=app etl
HOST_mydatabase.mydomain.net=database datamart
```
- This is a four-box setup:

```
HOST_localhost.com=app database
HOST_myreportingbox.mydomain.net=etl datamart
HOST_mycvs.mydomain.net=cvs
```

☞ **Note:** Observe that when the source control integrations run on a separate box from the main application, no *HOST_* variable is needed on the application box to point to the source control box. The source control machine needs to know how to find the main application box, but the application box does not need to know where the source code integration box is.

HTTPD_LOG_DIR

The *HTTPD_LOG_DIR* variable specifies the path where information about the activity of the TeamForge site's Apache service is written.

Values

Path specification

Default

{__LOG_DIR__}/httpd

INCLUDE_ORGANIZATION_USER_FIELD

The *INCLUDE_ORGANIZATION_USER_FIELD* variable controls whether the organization entry is displayed while creating a user account.

Values

true or false

Default

true

INDEXING_TIMEOUT

The *INDEXING_TIMEOUT* token allows you to configure the time limit for indexing a file.

Values

Integer (in min)

Default

5 min

INITIAL_PASSWORD_CHANGE_ACTIVATION_CODE_TIMEOUT

An administrator can optionally supply a password when creating a user. If the password is not specified while creating the user, the user is sent an email with a ticket to set the password. This *INITIAL_PASSWORD_CHANGE_ACTIVATION_CODE_TIMEOUT* token sets the duration (in hours) for which the password ticket is valid.

Values

Integer (hours)

Default

72

INTEGRATION_JAVA_OPTS

The *INTEGRATION_JAVA_OPTS* token specifies the memory settings for the Java virtual machine that supports the site's integrated source control services.

Values

Java specifications

Default

-Xms160m -Xmx160m -server -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/tmp -verbose:gc -XX: +PrintGCTimeStamps -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000 - Djava.security.egd=file:/dev/urandom

INTEGRATION_LOG_DIR

The *INTEGRATION_LOG_DIR* variable specifies the path where information about the activity of the TeamForge site's source code integrations is written.

Values

Path specification

Default

```
{__LOG_DIR__}/integration
```

JAMES_ACCEPTED_RELAYS

Set the *JAMES_ACCEPTED_RELAYS* token with the IP address of the Git server in case you have a distributed setup where Git runs on a separate server and you want Git notification emails to be sent to the TeamForge application server.

Values

Set this token as illustrated:

```
JAMES_ACCEPTED_RELAYS=127.0.0.1,{__CEE_DOMAIN__},<The IP address of the Git
  server>
```

Default

```
JAMES_ACCEPTED_RELAYS=127.0.0.1,{__CEE_DOMAIN__}
```

JAMES_GATEWAY_HOST

The *JAMES_GATEWAY_HOST* variable specifies a mail server with Internet access, separate from the TeamForge server.

Values

Email address specification

Default

None

Comments

Specifying a gateway host assures delivery of site email to users if your TeamForge server cannot connect to a DNS server or cannot get outside connections over port 25.

The mail account specified must be hosted on a separate server from the TeamForge site server.

The *SYSTEM_EMAIL*, *ADMIN_EMAIL*, and *JAMES_POSTMASTER_EMAIL* variables can specify the same address.

☞ **Note:** Specify the gateway host by its fully qualified domain name, not a host name.

JAMES_LOG_DIR

The *JAMES_LOG_DIR* variable specifies the path where information about the activity of the TeamForge site's email component is written.

Values

Path specification

Default

```
{__LOG_DIR__}/james
```

JAMES_POSTMASTER_EMAIL

The *JAMES_POSTMASTER_EMAIL* variable specifies a valid email address for the person or machine that handles email for the domain, such as postmaster@supervillain.org.

Values

Email address specification

Default

root@{__APPLICATION_HOST__}

Comments

The mail account specified must be hosted on a separate server from the TeamForge site server.

The *SYSTEM_EMAIL*, *ADMIN_EMAIL*, and *JAMES_POSTMASTER_EMAIL* variables can specify the same address.

JAVA_HOME

The *JAVA_HOME* variable specifies the path where the JBoss JVM is running.

Values

Path specification

Default

/usr/java/jdk1.8.0_45

JBOSS_ALARM_TIMEOUT

The *JBOSS_ALARM_TIMEOUT* variable specifies the time duration within which the JBoss service is expected to respond to requests sent by jboss_watchdog.

Values

Integer

Default

20

JBOSS_JAVA_OPTS

The *JBOSS_JAVA_OPTS* variable specifies the memory settings for the JBoss Java virtual machine.

Values

Java specifications

Default

-Xms1024m -Xmx2048m

☞ **Important:** All JVM parameters but -Xms1024m and -Xmx2048m have been hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) in the `site-options.conf` file.

When you change the default value of a JVM parameter such as "-XX:HeapDumpPath", the JBoss runtime parameters include both the user defined and default values for the JVM parameter. However, JBoss runs with the user defined value and ignores the default value.

LINUX_USERNAME_MODE_ENABLED

The *LINUX_USERNAME_MODE_ENABLED* token, if set to true, overrides the default TeamForge user naming convention that bars user names with anything but an alphabet as the first character. It is recommended to use this token on sites with CVS integration.

Values

true or false

Default

false

Comment

The token is commented out (disabled) by default and is available in the `site-options.conf` file.

☞ **Important:** Do not set the *LINUX_USERNAME_MODE_ENABLED* to true on sites without CVS integration. Instead, you must set the *RELAXED_USERNAME_MODE_ENABLED* token to true in case your site has no CVS integration.

⚠ **Caution:** While TeamForge can allow user names with anything but an alphabet as the first character, the same may not be true with all or some of the integrated applications you may have on your site. As a word of caution, consider the job at hand and understand the consequences before overriding TeamForge's default user naming convention.

LISTEN_BACKLOG

The *LISTEN_BACKLOG* token is used to specify the maximum length of the queue for the pending connections in the Apache server.

Values

Integer

Default

The default value is obtained from the system Kernel configuration.

`/sbin/sysctl -n -e net.ipv4.tcp_max_syn_backlog`

LOG_DIR

The *LOG_DIR* variable specifies the path where information about the TeamForge site's activity is written.

Values

Path specification

Default

`{__SITE_DIR__}/log`

LOG_QUERY_TIME_THRESHOLD

The *LOG_QUERY_TIME_THRESHOLD* variable enables you to log database requests at INFO level if they run longer than a given period.

By default, database requests are logged at DEBUG level. Configuring a value for *LOG_QUERY_TIME_THRESHOLD* causes requests that run for a period greater than that value to be logged at the INFO level, which makes them show up in `vamessages.log`.

Set the value to zero to log all database queries at INFO.

Values

Integer (in milliseconds)

Default

1000

LOGIN_ATTEMPT_LOCK

Use the *LOGIN_ATTEMPT_LOCK* token to set the permissible number of unsuccessful login attempts after which the user account is automatically locked.

Values

1-3

☞ **Important:** You can now selectively disable this feature even if you have set the *REQUIRE_PASSWORD_SECURITY* site options token to true. Set any negative value (such as '-1' or '-10000') in case you want to disable this feature altogether.

Note that with TeamForge 8.1 and earlier versions, setting *LOGIN_ATTEMPT_LOCK=-1* means the user account would be locked at the very first unsuccessful login attempt. This behavior has been removed from TeamForge 8.2 (and later).

Default

3

LOGIN_CONFIG_XML_FILE

The *LOGIN_CONFIG_XML_FILE* variable specifies the path to the LDAP configuration file.

Values

Path specification

Default

{__DATA_DIR__}/etc/login-config.xml

MAX_WWW_CLIENT

The *MAX_WWW_CLIENT* variable specifies the maximum number of Tomcat request processing threads to be created by the HTTP connector.

Values

Integer

Default

220

MEMCACHED_SERVER_HOST

This is the host name or IP address of the server that runs Memcached. Enter the host name or IP address of the Memcached server if and only if you have set the *ENABLE_CACHING_WITH_MEMCACHED* token to true.

Values

Host name or IP address of the server that runs Memcached.

Default

localhost

See *Install Memcached* on page 571 for more information.

MEMCACHED_SERVER_PORT

This is the port on which Memcached is configured. Enter the port that TeamForge connects to in order to communicate with Memcached if and only if you have set the *ENABLE_CACHING_WITH_MEMCACHED* token to true.

Values

Memcached port number.

Default

11211

See *Install Memcached* on page 571 for more information.

MEMCACHED_SERVER_TTL

The *MEMCACHED_SERVER_TTL* value, in seconds, is the duration for which the cached information is valid (or retained) in Memcached before it is invalidated or removed from Memcached. Enter the TTL (time to live) duration, in seconds, for the *MEMCACHED_SERVER_TTL* token if and only if you have set the *ENABLE_CACHING_WITH_MEMCACHED* token to true.

Values

Time duration, in seconds.

Default

600 (seconds)

Comments

Updates to TeamForge authorization rules such as PBPs, roles and role assignments, user restrictions, enabling or disabling users and so on can take up to N seconds to take effect, where `MEMCACHED_SERVER_TTL=N` (seconds). In other words, the `mod_authnz_ctf` module uses the cached information available with Memcached irrespective of the most recent changes, if any, on the TeamForge side. This implies that having a long TTL duration, while improves the response time, can also result in cached information being out of sync with TeamForge, especially on large sites that are prone to frequent RBAC changes.

See *Install Memcached* on page 571 for more information.

MIGRATION_LOG_DIR

The *MIGRATION_LOG_DIR* variable specifies the path where information about the conversion of site data is written during an upgrade.

Values

Path specification

Default

`{__LOG_DIR__}/runtime`

MINIMUM_PASSWORD_LENGTH

The *MINIMUM_PASSWORD_LENGTH* variable sets the shortest password that the system allows when a user account is created.

Values

Integer (number of characters)

Default

6

MINIMUM_USERNAME_LENGTH

The *MINIMUM_USERNAME_LENGTH* variable sets the shortest username that the system allows when a user account is created.

Values

Integer (number of characters)

Default

3

MIRROR DATABASE HOST

The *MIRROR_DATABASE_HOST* variable is a TeamForge database token that specifies the host of the database. This variable allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values

Alphanumeric string

Default

The MIRROR_ token takes the value of DATABASE_ token.

Comment

Example: Enter *MIRROR_DATABASE_HOST*=cu349.cloud.sp.collab.net (server name)

Add this token to the `site-options.conf` only if you setup a mirror database.

MIRROR_DATABASE_NAME

The *MIRROR_DATABASE_NAME* variable is a TeamForge database token that specifies the name of the TeamForge database. This variable allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values

Alphanumeric string

Default

The MIRROR_ token takes the value of DATABASE_ token.

Comment

Example: Enter *MIRROR_DATABASE_NAME*=ctfdb

Add this token to the `site-options.conf` only if you setup a mirror database.

MIRROR_DATABASE_PASSWORD

The *MIRROR_DATABASE_PASSWORD* variable is a TeamForge database token that specifies the password of the database. This variable allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values

Alphanumeric string

Default

The MIRROR_ token takes the value of DATABASE_ token.

Comment

Example: Enter *MIRROR_DATABASE_PASSWORD*=ctfpwd

Add this token to the `site-options.conf` only if you setup a mirror database.

MIRROR_DATABASE_PORT

The *MIRROR_DATABASE_PORT* variable is a TeamForge database token that specifies the port number of the database. This variable allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values

Port specification

Default

The MIRROR_ token takes the value of the DATABASE_ token.

Comment

Example: Enter *MIRROR_DATABASE_PORT*=5432.

Add this token to the `site-options.conf` only if you setup a mirror database.

MIRROR_DATABASE_USERNAME

The *MIRROR_DATABASE_USERNAME* variable is a TeamForge database token that specifies the database user's name. This variable allows to extract the reporting data from the mirror TeamForge database through the Extract, Transform and Load (ETL) process.

Values

Alphanumeric string

Default

The MIRROR_ token takes the value of the DATABASE_ token.

Comment

Example: Enter *MIRROR_DATABASE_USERNAME*=ctfuser

Add this token to the `site-options.conf` only if you setup a mirror database.

MODPAGESPEED_ENABLED

The *MODPAGESPEED_ENABLED* variable is used to enhance the performance of the web pages.

Values

true or false

Default

false

NOTIFY_SITE_ADMINS_FOR_SITE_ACTIVITIES

The *NOTIFY_SITE_ADMINS_FOR_SITE_ACTIVITIES* variable ensures that the activities at the site level are intimated to the site administrators through email notifications.

The site administrator can receive notifications on the following operations:

- User creation
- Project creation
- Blacklisted users
- SCM operations

Values

True or False

Default

True

This token was added in TeamForge 8.0.

OBFUSCATION_ENABLED

The *OBFUSCATION_ENABLED* variable is used to run the TeamForge application in the obfuscation mode for security purpose. Password obfuscation is enabled by default. As a result, all password-related tokens are encrypted in all the TeamForge configuration files.

Values

true or false

Default

true

Comment

When the TeamForge application is running in the obfuscation mode, the database login credentials, shared secrets etc., are encrypted and stored in the TeamForge configuration files for security reasons.

OBFUSCATION_KEY

The *OBFUSCATION_KEY* variable is used by the TeamForge obfuscation component as an input to the obfuscation algorithm for encryption and decryption purposes.

Values

AlphaNumeric (length greater than or equal to 8 bytes)

Default

XSJt43wN

ONLY_SITE_ADMIN_CAN_EDIT_SINGLE_SIGN_ON

This site-options token, if set to true, ensures that only site administrators can turn on single sign on (SSO) for linked applications (including Build & Test). Set this token to false to have both site and project administrators turn SSO on and off.

Values

Either true or false.

Default

true

This token was added in TeamForge 7.2.

ORGANIZATION_EDITABLE

The *ORGANIZATION_EDITABLE* variable allows or prevents editing the organization value of a user account.

Values

true or false

Default

true

PASSWORD_CONTROL_EFFECTIVE_DATE

The *PASSWORD_CONTROL_EFFECTIVE_DATE* token is used to set the date from which the password security feature takes effect.

Values

Date (mm/dd/yyyy)

Comments

The *REQUIRE_PASSWORD_SECURITY* on page 809 is the master token that enables the password security feature.

☞ **Important:** Setting the *PASSWORD_CONTROL_EFFECTIVE_DATE* token with a date is mandatory if *REQUIRE_PASSWORD_SECURITY* on page 809 is set to true.

Example 1

Consider a site with 130 users on which the password control kit (PCK) was not active. Of the 130 users, assume that:

- 100 users did not change password in last 100 days.
- 20 users did not change password in last 85 days.
- 10 users did not change password in last 75 days.

Assume that the following tokens are set on 01/01/2014 (current date):

- REQUIRE_PASSWORD_SECURITY=true
- PASSWORD_WARNING_PERIOD=20
- PASSWORD_EXPIRY_PERIOD=90
- PASSWORD_DISABLE_PERIOD=30
- PASSWORD_DELETE_PERIOD=60

PCK runs on 01/01/2014 and if you have PASSWORD_CONTROL_EFFECTIVE_DATE=01/10/2014 (set to a future date):

- 100 users with no password change for the past 100 days would get a warning message that their passwords will expire in 10 days.
- 20 users with no password change for the past 85 days would get a warning message that their passwords will expire in 10 days.
- 10 users with no password change for the past 75 days would get a warning message that their passwords will expire in 15 days.

Example 2

Consider the following scenario in which:

- Current date = 01/01/2014
- PASSWORD_CONTROL_EFFECTIVE_DATE=01/01/2013

In this scenario, the password control effective date is set to a date in the past. As a result, password control takes immediate effect and the PCK starts disabling, deleting or expiring user accounts right away.

PASSWORD_DELETE_PERIOD

The *PASSWORD_DELETE_PERIOD* variable specifies the time frame within which a disabled user account is automatically deleted.

Values

Integer (number of days)

Default

60

> ☞ **Note:** The *PASSWORD_DELETE_PERIOD* can be disabled by setting the value to zero.

PASSWORD_DISABLE_PERIOD

The *PASSWORD_DISABLE_PERIOD* variable specifies the time frame within which a user (soft-expired) is turned into a disabled user.

Values

Integer (number of days)

Default

30

Comments

A value of zero will disable this feature.

PASSWORD_EXPIRY_PERIOD

The *PASSWORD_EXPIRY_PERIOD* variable specifies the number of days after which the users' password expires.

Values

Integer (number of days)

Default

90

> ☞ **Note:** You cannot disable the *PASSWORD_EXPIRY_PERIOD* by setting the value to zero.

PASSWORD_REQUIRES_MIXED_CASE

The *PASSWORD_REQUIRES_MIXED_CASE* variable specifies that the user password must contain mixed case letters.

Values

true or false

Default

true

PASSWORD_REQUIRES_NON_ALPHANUM

The *PASSWORD_REQUIRES_NON_ALPHANUM* variable specifies that the user password must contain a non-alphanumeric character.

Values

true or false

Default

`true`

PASSWORD_REQUIRES_NUMBER

The *PASSWORD_REQUIRES_NUMBER* variable specifies that the user password must atleast contain one number.

Values

true or false

Default

`true`

PASSWORD_WARNING_PERIOD

Set this token to alert users via emails about impending password expiration on a daily basis. Email alert starts "N" days before password expiration due date, where *PASSWORD_WARNING_PERIOD=N*, and ends only when the password is changed by the user.

Values

Positive integer (number of days).

Default

14

PGSQL_COMMIT_DELAY

The *PGSQL_COMMIT_DELAY* variable specifies the time delay between writing a commit record to the write ahead log (WAL) buffer and flushing the buffer out to disk.

Values

Integer (in microseconds)

Default

`250`

Comments

Together with the *PGSQL_COMMIT_SIBLINGS* option, this option allows a group of otherwise unrelated transactions to be flushed to disk at the same time, with possible significant performance gain.

PGSQL_COMMIT_SIBLINGS

The *PGSQL_COMMIT_SIBLINGS* variable sets the minimum number of concurrent open transactions to require before performing the delay specified by the *PGSQL_COMMIT_DELAY* option.

Values

Integer

Default

`10`

Comments

Together with the *PGSQL_COMMIT_DELAY* option, this option allows a group of otherwise unrelated transactions to be flushed to disk at the same time, with possible significant performance gain.

PGSQL_EFFECTIVE_CACHE_SIZE

The *PGSQL_EFFECTIVE_CACHE_SIZE* variable specifies the size of the OS data cache that is available to PostgreSQL. PostgreSQL can use that data to select the optimal way to execute requests.

Comments

The right value for this variable depends in part on the available RAM on the server where your site is running. Set this value at the highest amount of RAM that you expect to be always available to PostgreSQL.

See *What are the right PostgreSQL settings for my site?* on page 909 for values recommended by CollabNet.

PGSQL_LOG_DIR

The *PGSQL_LOG_DIR* variable specifies the path where information about the activity of the TeamForge site's PostgreSQL database is written.

Values

Path specification

Default

```
{__LOG_DIR__}/pgsql
```

PGSQL_MAINTENANCE_WORK_MEM

The *PGSQL_MAINTENANCE_WORK_MEM* variable specifies the maximum amount of memory to be used in maintenance operations such as `VACUUM`.

Comments

See *What are the right PostgreSQL settings for my site?* on page 909 for values recommended by CollabNet.

PGSQL_MAX_CONNECTIONS

The *PGSQL_MAX_CONNECTIONS* variable determines the number of concurrent connections available to the database server.

Values

Integer

Default

```
135
```

PGSQL_MAX_FSM_PAGES

The *PGSQL_MAX_FSM_PAGES* variable tells the `vacuum` process how many pages to look for in the shared free-space map.

Values

Integer

Default

```
500000
```

Comments

Each FSM page uses 6 bytes of RAM for administrative overhead, so increasing FSM substantially on systems low on RAM may be counter-productive.

PGSQL_MAX_FSM_RELATIONS

The *PGSQL_MAX_FSM_RELATIONS* variable specifies how many relations (tables) will be tracked in the free space map.

Values

Default

500

PGSQL_MAX_STACK_DEPTH

The *PGSQL_MAX_STACK_DEPTH* variable specifies the maximum safe depth of the server's execution stack.

Values

Integer

Default

5120

PGSQL_SHARED_BUFFERS

The *PGSQL_SHARED_BUFFERS* variable defines a block of memory that PostgreSQL will use to hold requests that are awaiting attention from the kernel buffer and CPU.

Comments

The right value for this variable depends in part on the available RAM on the server where your site is running.

See *What are the right PostgreSQL settings for my site?* on page 909 for values recommended by CollabNet.

PGSQL_STATEMENT_TIMEOUT

The *PGSQL_STATEMENT_TIMEOUT* variable is set to prevent the Postgres queries from running for a long period of time.

Values

Integer (Milliseconds)

Default

600000 (Milliseconds)

Comments

An error message is displayed for every timeout in the postgres.log file and the log message with the exid id is logged in the vamessages.log and server.log files.

PGSQL_VACUUM_COST_DELAY

The *PGSQL_VACUUM_COST_DELAY* variable controls the length of time that an I/O process will sleep when the limit set by *vacuum_cost_limit* has been exceeded.

Values

Integer (milliseconds)

Default

```
50
```

PGSQL_WAL_BUFFERS

The *PGSQL_WAL_BUFFERS* variable specifies the number of buffers available for the Write Ahead Log.

Comments

If your database has many write transactions, setting this value bit higher than default may result better usage of disk space.

See *What are the right PostgreSQL settings for my site?* on page 909 for values recommended by CollabNet.

PGSQL_WORK_MEM

The *PGSQL_WORK_MEM* variable specifies the amount of memory to be used by internal sort operations and hash tables before switching to temporary disk files. .

Comments

The right value for this variable depends in part on the available RAM on the server where your site is running.

See *What are the right PostgreSQL settings for my site?* on page 909 for values recommended by CollabNet.

PHOENIX_JAVA_OPTS

The *PHOENIX_JAVA_OPTS* variable specifies the memory settings for the Java virtual machine that supports the site's ability to send and receive email and to index data for search.

Values

Java specifications

Default

-Xms256m -Xmx256m -server -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/tmp - verbose:gc -XX:+ PrintGCTimeStamps -XX:+PrintGCDetails -Dsun.rmi.dgc.client.gcInterval=600000 - Dsun.rmi.dgc.server.gcInterval=600000 -Dsf .luceneOptimizeEvery=100000 -Djava.security.egd=file:/dev/urandom

PLANNING_BOARD_SWIM_LANE_LIMIT

By default, not more than 250 cards are shown in a planning board swimlane. However, as a site administrator, you can increase or decrease the number of cards shown in the planning board swimlanes by configuring the site options token, PLANNING_BOARD_SWIM_LANE_LIMIT.

Values

A positive number.

Default

250

Comments

When you select a planning folder in one of the swimlanes and if X is greater than N, (where X = number of artifacts in the selected planning folder and N = PLANNING_BOARD_SWIM_LANE_LIMIT), the message, `Swimlanes in the Board View is currently configured to show N artifacts only,` appears at the bottom of the swimlane.

PLANNING_FOLDER_DESC_EDITOR

The *PLANNING_FOLDER_DESC_EDITOR* variable allows you to choose the type of text that can be used in the planning folder description using the editor tool.

Values

Plain Text

Default

Plain Text

RELAXED_USERNAME_MODE_ENABLED

The *RELAXED_USERNAME_MODE_ENABLED* token, if set to true, overrides the default TeamForge user naming convention that bars user names with anything but an alphabet as the first character. It is recommended to use this token on sites without CVS integration.

Values

true or false

Default

false

Comment

The token is commented out (disabled) by default and is available in the `site-options.conf` file.

☞ **Important:** Do not set the *RELAXED_USERNAME_MODE_ENABLED* to true on sites with CVS integration. Instead, you must set the *LINUX_USERNAME_MODE_ENABLED* token to true in case your site has CVS integration.

⚠ **Caution:** While TeamForge can allow user names with anything but an alphabet as the first character, the same may not be true with all or some of the integrated applications you may have on your site. As a word of caution, consider the job at hand and understand the consequences before overriding TeamForge's default user naming convention.

REPORTS_DATABASE_NAME

The *REPORTS_DATABASE_NAME* variable specifies the name of the site's reporting database, also known as the datamart.

Values

Alphanumeric string

Default

ctfrptdb

Comments

It is OK for this variable to have the same value as *DATABASE_NAME*, because they are running in separate pgsql processes.

REPORTS_DATABASE_PASSWORD

The *REPORTS_DATABASE_PASSWORD* variable is the password for the Linux user that is authorized to read from and write to the site's reporting database.

Values

Alphanumeric string

Default

ctfrptpwd

Comments

It is OK for this variable to have the same value as *DATABASE_PASSWWORD*, because they are running in separate PostgreSQL processes.

REPORTS_DATABASE_PORT

The *REPORTS_DATABASE_PORT* variable defines a separate port for the reporting database (aka datamart). Using a separate port can improve site performance when database utilization is high.

Values

Port specification

Default

5632

Comments

As of TeamForge 6.1, only port 5632 is supported.

REPORTS_DATABASE_USERNAME

The *REPORTS_DATABASE_USERNAME* variable specifies the Linux user that is authorized to read from and write to the site's reporting database.

Values

Alphanumeric string

Default

ctfuser

Comments

For some advanced operations, you may need to log into the database as the database user. However, under normal conditions only the TeamForge site process itself needs to access the database.

It is OK for this variable to have the same value as *DATABASE_USERNAME*, because they are running in separate PostgreSQL processes.

REPORTS_ENABLE_REPORT_GENERATION

The *REPORTS_ENABLE_REPORT_GENERATION*variable is used to enable or disable the Reports tab in the UI.

Values

true or false

Default

true or false

Comments

Datamart is enabled by adding the 'datamart' service to the HOST_<hostname>token. The service is disabled if datamart is not added. The default value of the REPORTS_ENABLE_REPORT_GENERATION token is based on this service.

REQUIRE_PASSWORD_SECURITY

The *REQUIRE_PASSWORD_SECURITY* token, if set to true, enforces password security policy for the site.

Values

true or false

Default

true

Comment

This variable can be useful when an organization's security policy prohibits users from entering passwords without any restrictions. You can also set the *PASSWORD_CONTROL_EFFECTIVE_DATE* token with a date from which the password policy would be enforced. For more information, see *PASSWORD_CONTROL_EFFECTIVE_DATE* on page 802.

REQUIRE_RANDOM_ADMIN_PASSWORD

The *REQUIRE_RANDOM_ADMIN_PASSWORD* variable restricts users from setting a random admin password.

Values

true or false.

Default

True (SaaS), false (BTF)

Comment

This variable when set to true, checks for a valid mail id in the ADMIN_EMAIL token.

REQUIRE_USER_PASSWORD_CHANGE

The *REQUIRE_USER_PASSWORD_CHANGE* variable determines if the user password needs to be changed during the first login instance.

Values

true or false.

Default

true

Comment

Setting a value true makes the new system force users to change password during first login and false otherwise.

RUNTIME_LOG_DIR

The *RUNTIME_LOG_DIR* variable specifies the path where information about the activity of the TeamForge site's runtime environment is written.

Values

Path specification

Default

{__LOG_DIR__}/runtime

SCM_DEFAULT_SHARED_SECRET

The *SCM_DEFAULT_SHARED_SECRET* variable allows SCM Integrations to securely communicate with the TeamForge app server.

Values

- Alpha-numeric
- Special characters like '~!@#$%^&*'
- 16-24 byte length

Default

The default value is automatically generated during runtime.

SCM_SOAP_TIMEOUT

The *SCM_SOAP_TIMEOUT* variable is used to specify the connection timeout of the SCM soap requests between the APP and SCM servers.

Values

Integer (Milliseconds)

Default

300000 (Milliseconds)

SCM_USER_ENCRYPTED_PASSWORD

The *SCM_USER_ENCRYPTED_PASSWORD* variable is used to store the encrypted scmviewer password.

Values

- Alpha-numeric
- Special characters like '~!@#$%^&*'

Default

The default value will be in the encrypted format. See *password_util.sh* on page 762 for more information.

SEARCH_LOG_DIR

The *SEARCH_LOG_DIR* variable specifies the path where information about the activity of the TeamForge site's Lucene search component is written.

Values

Path specification

Default

```
{__LOG_DIR__}/james
```

SEARCH_MAX_FILE_SIZE

The *SEARCH_MAX_FILE_SIZE* variable sets an upper limit to the size of files that are indexed for search.

Values

Integer (bytes)

Default

10M

Comment

A value of zero or less specifies that there is no limit, which is the same as the default behavior without the variable.

SEARCH_SUPPRESS_ARCHIVE_SUB_DOCS

The *SEARCH_SUPPRESS_ARCHIVE_SUB_DOCS* variable prevents archive files from being indexed for search.

Archive files include `zip`, `gzip`, `tar`, and similar file types. They also include document files that are stored in archive format, such as `docx` files from Microsoft Word 2007.

Values

true, false

Default

`true`

SESSION_COOKIES_ONLY

the *SESSION_COOKIES_ONLY* variable restricts the persistence of all cookies to the user's current session.

If *SESSION_COOKIES_ONLY*=true, then all cookies created during the user session expire automatically when the user closes their browser. If it is false, the cookie expires according to the system logic for that particular cookie.

Values

true or false

Default

false

Comment

This variable can be useful when an organization's security policy prohibits cookies that persist across user sessions.

SOAP_ANONYMOUS_SHARED_SECRET

The *SOAP_ANONYMOUS_SHARED_SECRET* variable allows users to have an anonymous login to the TeamForge site through SOAP.

Values

String (possibly encrypted)

Default

None

Comment

The variable must be configured to a non-empty value if users need to have an anonymous login to the site through SOAP. A value must be provided if site-wide reporting is enabled.

SOAP_ARTIFACT_LIST_LIMIT

The *SOAP_ARTIFACT_LIST_LIMIT* variable is used to limit the number of artifacts returned via SOAP calls.

Values

Integer

Default

-1: this means that the artifact list retrieved via SOAP is unlimited

Comments

In TeamForge releases earlier than 6.1.1, SOAP calls returned everything that was asked for, and that is the default behavior in Teamforge 6.1.1 as well. However, sites with performance and stability issues (OutOfMemory errors) in returning a large number of artifacts can now limit the number using this token. Changing this value requires a recreate-runtime and thus a site restart.

☞ **Important:** Increasing the number of artifacts beyond the optimal 20,000 - 25,000 range might cause a heap dump.

SSL

The *SSL* variable activates Secure Socket Layer encryption for the TeamForge site.

Values

on or off

Default

on

SSL_CERT_FILE

The *SSL_CERT_FILE* specifies the path to the file where the TeamForge site's Secure Socket Layer certificate is stored.

Values

Path specification

Default

None

SSL_CHAIN_FILE

The *SSL_CHAIN_FILE* variable specifies the path to the file where the TeamForge site's SSL certficate chain file is stored.

Values

Path specification

Default

None

SSL_CIPHER_SUITE

The *SSL_CIPHER_SUITE* token disables some of the less secure methods.

Values

SSLCipherSuite method

Default

ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-DSS-AES128-GCM-SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-AES128-SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-SHA:DHE-RSA-AES256-SHA:AES128-GCM-SHA256:AES256-GCM-SHA384:AES128-SHA256:AES256-SHA256:AES128-

SHA:AES256-SHA:AES:CAMELLIA:DES-CBC3-SHA:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!PSK:!
aECDH:!EDH-DSS-DES-CBC3-SHA:!EDH-RSA-DES-CBC3-SHA:!KRB5-DES-CBC3-SHA

SSL_PROTOCOL

The *SSL_PROTOCOL* token disables some of the less secure methods.

Values

SSLProtocol method

Default

all -SSLv3 -SSLv2

SSL_KEY_FILE

The *SSL_KEY_FILE* specifies the path to the file where the TeamForge site's RSA private key is stored when Secure
Socket Layer encryption is in effect.

Values

Path specification

Default

None

SUBVERSION_BRANDING_URI

The *SUBVERSION_BRANDING_URI* variable specifies the path component of the data repository URL.

Values

BDB or FSFS

Default

BDB

SUBVERSION_REPOSITORY_BASE

The *SUBVERSION_REPOSITORY_BASE* variable specifies the path to the root directory for the site's Subversion
repositories. You can use this variable to put your source code repositories at a custom location on your site's server
or on the network.

Values

Path specification

Default

/svnroot

SVN_AUTHNZ_TIMEOUT

The *SVN_AUTHNZ_TIMEOUT* token allows you to set the timeout value (in seconds) for the mod_authnz_ctf
module.

Values

Timeout value in number of seconds.

Default

60 seconds

SYSTEM_EMAIL

The *SYSTEM_EMAIL* variable specifies a valid email address for the system administrator responsible for this site.

System administrators can use this email address to set up outage alerts and other notifications.

The mail account specified must be hosted on a separate server from the TeamForge site server.

The *SYSTEM_EMAIL*, *ADMIN_EMAIL*, and *JAMES_POSTMASTER_EMAIL* variables can specify the same address.

> ☞ **Important:** In TeamForge 6.x, the sender name and address for system-generated emails is taken from the value of the *SYSTEM_EMAIL* variable. Therefore, changing the admin user's full name or email address does not affect the sender details of system-generated emails. This is different from TeamForge 5.x, in which the sender name and address for system-generated emails is derived from the admin user's full name and email address.

Values

Email address specification

Default

root@{__APPLICATION_HOST__}

USE_BROWSER_CACHE_PASSWORD

The *USE_BROWSER_CACHE_PASSWORD* variable restricts the storage of password in the browser when you login to the site.

Values

true/false

Default

true

USE_EXTERNAL_USER_AUTHENTICATION

The *USE_EXTERNAL_USER_AUTHENTICATION* variable specifies whether users can be authenticated through a separate system, such as OpenLDAP.

Values

true or false

Default

false

USER_ACCOUNT_RESTRICTED

The *USER_ACCOUNT_RESTRICTED* variable determines whether newly created users are "restricted" or "unrestricted" users by default.

- Restricted users can access only public projects and projects of which they are members.
- Unrestricted users can access all projects except private projects of which they are not members.

Values

true or false

Default

true

USER_MONITORING_REMOVE_ENABLED

Set the *USER_MONITORING_REMOVE_ENABLED* variable to true, if you want to enable the feature that lets you remove one or more users from monitoring selected TeamForge objects.

Values

true or false

Default

false

USER_NEED_PERMISSION_TO_VIEW_FULL_USER_DETAILS

The *USER_NEED_PERMISSION_TO_VIEW_FULL_USER_DETAILS* variable restricts users from viewing other users' organization information.

Values

true or false

Default

false

USERS_WITH_NO_EXPIRY_PASSWORD

The *USERS_WITH_NO_EXPIRY_PASSWORD* token specifies the users for whom there is no expiry of password.

Values

Specify the usernames (for the user accounts) for which there is no expiry of password.

Default

USERS_WITH_NO_EXPIRY_PASSWORD=admin,nobody,system,scmviewer,scmadmin

Comment

The variable is enabled by default.

Using multi-line blocks for site options

The multi-line block configuration is generally used by old SFEE sites. To define a `site-options.conf` token with a multi-line block value, you need to follow a certain syntax.

- Declare the token name with the value "START_MULTILINE_BLOCK". Syntax:
  <TOKEN_NAME>=START_MULTILINE_BLOCK
- Specify the multi-line values beneath the token.
- Complete the multi-line block with "END_MULTILINE_BLOCK" after all the multi-line values are specified. Syntax: END_MULTILINE_BLOCK

Example

```
SOURCEFORGE_CONFIGURATION_PROPERTIES_APPEND=START_MULTILINE_BLOCK
email.suppress.project_member_added=true
email.suppress.scm_user_password_synchronized=true
END_MULTILINE_BLOCK
```

c6migrate.conf variables

These are the configuration settings you can specify in the `c6migrate.conf` file.

Name	Description	Default	Values	Comments
LOG_LEVEL	When set to "INFO", the log is not verbose.	INFO	DEBUG, INFO, WARN, ERROR	
CEE_SITE_DIR	The path to the CEE site directory, for example, `/u1/ sourcecast`.			
PROJECT_TRACKER_XML_FILE	The path to the PT XML file, for example, `/u3/ PTInstaller-1.0.45/ installer/ conf/pt.xml`.			
PROJECT_CREATION_DATE	The fall-back date is used when a project's or role's creation/ modification date is not found in the audit log.	2000-01-01\ 00:00:00		

httpd.conf

These are the changes you must make to the `/etc/httpd/conf/httpd.conf` file.

```
##
# SFEE configuration
##
# mod_deflate for improving performance
DeflateFilterNote Input instream
DeflateFilterNote Output outstream
DeflateFilterNote Ratio ratio
LogFormat '"%r" %{outstream}n/%{instream}n {%{ratio}n%%}' deflate
<Location />
 AddOutputFilterByType DEFLATE text/html
 # Netscape 4.x has some problems...
 BrowserMatch ^Mozilla/4 gzip-only-text/html
 # Netscape 4.06-4.08 have some more problems
 BrowserMatch ^Mozilla/4\.0[678] no-gzip
# NOTE: Due to a bug in mod_setenvif up to Apache 2.0.48
 # the above regex won't work. You can use the following
 # workaround to get the desired effect:
 BrowserMatch \bMSI[E] no-gzip
 # Don't compress images
 SetEnvIfNoCase Request_URI \
  \.(?:gif|jpe?g|png)$ no-gzip dont-vary
 # Make sure proxies don't deliver the wrong content
 Header append Vary User-Agent env=!dont-vary
</Location>

# mod_expires for even better performance
ExpiresActive On
ExpiresDefault "access plus 0 seconds"
ExpiresByType image/gif "access plus 1 days"
ExpiresByType image/jpeg "access plus 1 days"
ExpiresByType image/png "access plus 1 days"
ExpiresByType text/css "access plus 7 days"
```

```
ExpiresByType text/javascript "access plus 7 days"
ExpiresByType application/x-javascript "access plus 7 days"
ExpiresByType image/x-icon "access plus 7 days"

# SFEE rewrites to make the app 'live' on port 80 and not 8080
RewriteEngine on
RewriteLog logs/rewrite
RewriteLogLevel 1
# Added to supress http trace for security reasons
RewriteCond %{REQUEST_METHOD} ^TRACE
RewriteRule .* - [F]
# make '/' redirect to SFEE
RewriteRule ^/$ http://%{SERVER_NAME}/sf/ [R]
# now pass the URL to the actual SFEE application server
RewriteRule ^/sf$ http://localhost:8080/sf [P]
RewriteRule ^/sf/(.*) http://localhost:8080/sf/$1 [P]

# Pass ScmListener SOAP requests
RewriteCond %{REQUEST_URI} ^/ce-soap50/services/ScmListener
RewriteRule ^/ce-soap50/(.*) http://localhost:8080/ce-soap50/$1 [P]
#Pass all non-listeners SOAP requests. Delete next 4 lines if you don't use
 SOAP APIs.
RewriteCond %{REQUEST_URI} !^/ce-soap50/services/[^/]*Listener
RewriteRule ^/ce-soap50/(.*) http://localhost:8080/ce-soap50/$1 [P]
RewriteRule ^/ce-soap5042/(.*) http://localhost:8080/ce-soap5042/$1 [P]
RewriteRule ^/ce-soap5043/(.*) http://localhost:8080/ce-soap5043/$1 [P]

# route SCM requests to the SFEE integration server
RewriteCond %{REQUEST_URI} !^/integration/services
RewriteCond %{REQUEST_URI} !^/integration/servlet
RewriteRule ^/integration/(.*) http://localhost:7080/integration/$1 [P]
ProxyPassReverse / http://localhost:8080/
ProxyPassReverse / http://localhost:7080/
##
# end SFEE configuration
##
```

pebble-app.xml

The `pebble-app.xml` file, also known as the Pebble application configuration file, contains the text that the Pebble application displays in the TeamForge user interface.

This is an example of a default (unedited) `pebble-app.xml` file. To create your own integrated application config file, copy this one into a new file and replace the values with the values appropriate for the application you are integrating.

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE integrated-application
  PUBLIC "-//CollabNet, Inc.//DTD Integrated Application Descriptor 1.0//EN"
  "http://schema.open.collab.net/sfee50/dtd/sf-pluggable-application-
descriptor_1_0.dtd">
<integrated-application>
    <name>Pebble Blog</name>
    <description>l10n.application.description</description>
    <permissions>
        <permission dapMappedTo="View">Blog Reader</permission>
        <permission>Blog Contributer</permission>
        <permission>Blog Publisher</permission>
        <permission>Blog Owner</permission>
    </permissions>
    <prefix>PB</prefix>
    <id-pattern></id-pattern>
    <require-per-project-prefix>true</require-per-project-prefix>
```

```xml
    <require-scm-integration>true</require-scm-integration>
    <is-search-supported>false</is-search-supported>
    <!-- Page components for Integrated apps is not implemented for Alpha -->
    <page-component>
      <require-page-component>true</require-page-component>
      <page-component-details>
        <inputtype>text</inputtype>
        <resultformat>html</resultformat>
        <description>l10n.pce.description</description>
        <title>l10n.pce.title</title>
      </page-component-details>
    </page-component>
    <config-parameters>
        <!-- Pebble Configuration Parameters -->
        <param>
            <title>l10n.blogname.title</title>
            <name>blogName</name>
            <description>l10n.blogname.description</description>
            <defaultvalue>My Blog</defaultvalue>
            <displaytype valuetype="String" maxlength="25">TEXT</displaytype>
            <editable>false</editable>
        </param>
        <param>
            <title>l10n.blogdescription.title</title>
            <name>blogDescription</name>
            <description>l10n.blogdescription.description</description>
            <defaultvalue>My Awesome Blog</defaultvalue>
            <displaytype valuetype="String" maxlength="40">TEXT</displaytype>
            <editable>true</editable>
        </param>
        <param>
            <title>l10n.richtexteditor.title</title>
            <name>richTextEditorEnabled</name>
            <description>l10n.richtexteditor.description</description>
            <defaultvalue>checked</defaultvalue>
            <displaytype valuetype="String">CHECKBOX</displaytype>
            <editable>true</editable>
        </param>
        <param>
            <title>l10n.noofrecentblogentries.title</title>
            <name>recentBlogEntries</name>
            <description>l10n.noofrecentblogentries.description</description>
            <defaultvalue>3</defaultvalue>
            <displaytype valuetype="String">SELECT</displaytype>
            <option name="3">l10n.three.value</option>
            <option name="5">l10n.five.value</option>
            <option name="7">l10n.seven.value</option>
            <option name="9">l10n.nine.value</option>
            <editable>true</editable>
        </param>
    </config-parameters>
    <bundles>
        <bundle locale="en">
            <key name="l10n.application.description">Pebble Blog App</key>
            <key name="l10n.pce.description">Display Blog Title for Given
Date.</key>
            <key name="l10n.pce.title">Enter Blog Date (in yyyy-mm-dd)</key>
            <key name="l10n.blogname.title">Blog Name</key>
            <key name="l10n.blogname.description">Please provide a name for
the Blog. This appears on all blog pages</key>
            <key name="l10n.blogdescription.title">Blog Description</key>
            <key name="l10n.blogdescription.description">Please provide a
description for the Blog. This appears below blog name on all pages</key>
            <key name="l10n.richtexteditor.title">Rich Text Editor</key>
```

```
                <key name="l10n.richtexteditor.description">Enable Rich Text
 Editor for comments and Blog entries?</key>
                <key name="l10n.noofrecentblogentries.title">Recent Blog Entries</
key>
                <key name="l10n.noofrecentblogentries.description">How many recent
 blog entries do you want to see in the home page?</key>
                <key name="l10n.three.value">3</key>
                <key name="l10n.five.value">5</key>
                <key name="l10n.seven.value">7</key>
                <key name="l10n.nine.value">9</key>
        </bundle>
    </bundles>
</integrated-application>
```

pebble-dep.xml

The `pebble-dep.xml` file, also known as the Pebble deployment configuration file, contains the data that Pebble needs to interact with the TeamForge site.

A sample pebble-dep.xml file for the REST service type

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE integrated-application
  PUBLIC "-//CollabNet, Inc.//DTD Integrated Application Descriptor 1.0//EN"
  "http://schema.open.collab.net/sfee50/dtd/sf-pluggable-deploy-
descriptor_1_0.dtd">
<integrated-application>
    <name>Pebble Blog</name>
    <baseurl>https://cu064.cloud.sp.collab.net:13001/pebble/index.jsp</
baseurl>
    <gourl>https://cu064.cloud.sp.collab.net:13001/pebble/gourl/%p/%o</gourl>
    <endpoint>https://cu064.cloud.sp.collab.net:13001/pebble/services/rest/
ctfapi</endpoint>
 <servicetype>REST</servicetype>
</integrated-application>
```

A sample pebble-dep.xml file for the SOAP service type

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE integrated-application
  PUBLIC "-//CollabNet, Inc.//DTD Integrated Application Descriptor 1.0//EN"
  "http://schema.open.collab.net/sfee50/dtd/sf-pluggable-deploy-
descriptor_1_0.dtd">
<integrated-application>
    <name>Pebble Blog</name>
    <baseurl>https://cu177.cloud.maa.collab.net:13001/pebble/index.jsp</
baseurl>
    <gourl>https://cu177.cloud.maa.collab.net:13001/pebble/gourl/%p/%o</gourl>
<endpoint>https://cu177.cloud.maa.collab.net:13001/pebble/services/
PebbleIntegrationService</endpoint>
    <servicetype>SOAP</servicetype>
</integrated-application>
```

How is an integrated application described?

An integrated application is described using two XML files - a deployment configuration file and an application configuration file - that provide information to TeamForge about the configuration options exposed by the application.

In TeamForge version 6.1.1 and later, you have the ability to configure some integrated application settings using the user interface. You can also export these settings in XML format and make changes. To edit configuration settings, you would upload the XML file containing the updates.

Integrated application settings

☞ **Note:** Some of the tags are internationalized so that the application will display languages based on the browser locale. See *Internationalize your integrated application* on page 746 for more information.

<name>

This is the title of the integrated application. When the integrated application is added to a project, the button that appears on the project pages has this name. This name must be unique -- you cannot use it for any other integrated application on the same TeamForge server.

This tag is used in both deployment and application configuration files.

<adminurl>

When an application has an administration screen for configuring its parameters, this field contains that URL. It is optional.

This tag is used in the deployment configuration file.

<baseurl>

This is the URL to which a user will be directed on clicking the integrated application button in a project.

This tag is used in the deployment configuration file.

<endpoint>

This is the SOAP endpoint for the integrated application. The endpoint contains the various methods exposed by the integrated application that are called during the lifecycle of TeamForge.

This tag is used in the deployment configuration file.

<gourl>

This indicates which URL must be used when an object id for an integrated application is specified (either via Jump_to_id or on the URL as /sf/go/<objectid>). This URL can support a couple of dynamic parameters.

*   `%o` -- The object id entered by the user will be dynamically replaced here.
*   `%p` -- The project id for the object entered will be dynamically replaced here.

For example, if the Go URL is `http://go.tourl.com/tracking?id=%o` and the object ID entered is `XYZ123`, then the URL will be replaced and redirected to `http://go.tourl.com/tracking?id=XYZ123`.

This tag is used in the deployment configuration file.

<category>

This is used to display the integrated application under a separate **Category** tab. It is an optional parameter that is manually inserted based on the requirement. If the category is not defined, the integrated application appears under the **Generic** tab.

This tag is used in the deployment configuration file.

<isbaseapp>

This validates whether the integrated application is a base application configured for the specified category. It is an optional parameter that is manually inserted based on the requirement.

This tag is used in the deployment configuration file.

<is-search-supported>

This validates whether the integrated application needs to be displayed as an object category in the drop-down list for search options in TeamForge such as **Jump to ID** and **Advanced Search**. It is an optional parameter that is manually inserted based on the requirement.

This tag is used in the application configuration file.

<config-parameters>

There can be any number of configuration parameters for an integrated application and they are displayed when associating the application to a project. These parameters are filled in by the project administrator and are available in the integrated application SOAP interface as configuration parameters. The integrated application gets a chance to validate these parameters and indicate back to TeamForge whether this project association is successful by passing in a "TRUE". It can return a "FALSE" if it doesn't want this project association to succeed. Each configuration parameter is placed inside the "param" tag, which can contain multiple elements to describe the parameter.

<title>	The internationalized title that appears for a project administrator to fill in while associating the integrated application to a project.
<name>	The Java variable under which the value for this parameter will be available on the integrated application.
<description>	The internationalized description that appears when a project administrator fills in or enters a configuration parameter.
<default value>	The default value for the parameter that will appear in the user interface during the association of an integrated application to a project.
<display type>	This is the type of display control used for the configuration parameter. We support "TEXT" for text fields, "CHECKBOX" for checkbox type controls, "RADIO" for radio buttons, and "SELECT" for select dropdowns. This field can also take an attribute that says what the value type for the field should be -- whether it should be an "Integer", "String" and so on. So if the field is expecting numbers, then entering "foo" as a value will throw a validation failure.
<option>	If the display type is "RADIO" or "SELECT", then these fields contain the individual options available for the display controls. This will contain a "name" attribute that will be sent to the integrated application when that option is selected from the UI. The value of this option should be an internationalized field as it is the value visible to the user.
<editable>	This specifies whether the configuration parameter should be editable once the integrated application is associated to a project. These configuration parameters are available when you add or edit an integrated project. If a parameter should not be "edited" post association, setting this to "false" will make it non-editable.

This tag is used in the application configuration file.

<description>

This is an internationalized string for the integrated application's description. It contains information for TeamForge project and site administrators to know what the application does.

This tag is used in the application configuration file.

<id-pattern>

When trying to link to an integrated application id, this regular expression gets used for mapping. By default (if no value is provided), it looks for alphanumerical characters; in case you need specific characters to be matched (for example, JIRA, which has hyphens in ids), this value is used.

This tag is used in the application configuration file.

<page-component>

These settings are used for Project Content Editors. The integrated application content can become part of the standard Page Component data that appears in project home pages. The settings indicate the type of information that will be available from the integration application.

<input-type>	This is the input type control for an integrated app Page Component. We only support 2 types now. Either "select" so that the inputs can be shown from a "SELECT" dropdown and the users will be able to pick a value from there. Else, it can be a "text" where a simple "text" field will be entered for taking the user input.
<result-format>	This is the format in which the output of Page Component is returned. This can be a "list" which indicates that it will be a Table like output. The integrated app will send the results in an XML format and the Integrated app framework converts this into a list of records. The other option is "html", where the output from the Integrated application is just displayed on the screen.
<page-component-description>	The description that will appear when you add an Integrated application Page Component (Link to the page where " Add component" is available)
<page-component-title>	The title that will appear when adding an Integrated application Page Component (Link to the page where "PCE Add component" is available)

This tag is used in the application configuration file.

<permissions>

This is a collection of permissions that are exposed by the integrated application. There could be any number of such permissions. These permissions will appear as a part of the project's roles (existing ones, as well as ones newly added) and can be assigned along with other tool permissions. You can map one of these permissions with a "dapMappedTo" attribute -- this indicates the permission to be used when a user logs in without authentication (for example, for public projects). Typically, this is the permission to read data so that it doesn't need a login name; it varies from one application to another.

This tag is used in the application configuration file.

<prefix>

If the "require-per-project-prefix" attribute is false, the value of this tag is used for identifying the integrated application in Go URLs, associations, and linkifications. If the "require-per-project-prefix" attribute is true, the value is used only for the "Host" project. Each project must fill in its value as part of adding the integrated application. Click here for *steps to add integrated applications to a project*.

This tag is used in the application configuration file. The prefix can contain alpha-numeric characters and cannot be more than six characters in length. For more information about prefix, see *How does an integrated application interact with other TeamForge tools?*

<require-per-project-prefix>

An integrated application can indicate to TeamForge whether the object ids that it generates are uniquely identifiable across the entire application (if yes, the value for the attribute is "false") or whether they need to be project-specific (in this case, the value for the attribute is "true"). If an integrated application needs per-project prefix, you must enter the prefix value when the integrated application is added to a project.

This tag is used in the application configuration file.

<require-scm-integration>

This indicates whether SCM commits need to be vaildated. Some applications might have business rules which indicate that a commit can be made only if certain conditions are met. If the integrated application has any such rules, the value for the attribute should be "true". There are also a couple of methods to be implemented in the SOAP endpoint.

This tag is used in the application configuration file.

<require-page-component>

Some integrated applications choose not to expose details as Page Components. For those that don't, set this tag to "false" and for those that do, set it to "true". If the value is "true", you must provide the "page-component-details" tags as well.

This tag is used in the application configuration file.

<servicetype>

TeamForge 6.1.0 and earlier releases supported only SOAP as the mechanism to talk from TeamForge to the integrated application. TeamForge 6.1.1 and later support REST calls. The servicetype tag indicates whether the protocol used for communication is REST or SOAP.

This tag is used in the deployment configuration file.

For examples of how these tags are used in the integration of the Pebble bogging application, see *pebble-dep.xml* and *pebble-app.xml*.

install.conf

The `install.conf` file contains the data needed to manage the Pebble installer.

This is an example of a default (unedited) `install.conf` file. To create your own Pebble installer config file, copy this one into a new file and replace the values with the values appropriate for the application you are integrating.

```
#Location of the Pebble Blog
pebble.base.dir=/u1/pebble
#CTF Information
ctf.baseurl=http://cu073.cubit.maa.collab.net/
#Tomcat Information
tomcat.port=13000
domain=cu073.cubit.maa.collab.net
timezone=Asia/Calcutta
java_home=/usr/java/jdk1.6.0_26/
protocol=http
java_opts=-Xms512m -Xmx512m
# This where the Pebble Blogs are stored
data.dir=/u1/pebble-data
secretkey=mistywasacat
```

install.conf

The `install.conf` file contains the data needed to manage the Review Board installer.

This is an example of a default (unedited) `install.conf` file. To create your own Review Board installer config file, copy this one into a new file and replace the values with the values appropriate for the application you are integrating.

```
#Location of the Review Board installation directory
rb_dir=/u1/reviewboard

#Location of the Review Board data directory
rb_data_dir=/opt/collabnet/reviewboard/data

#Review Board site information
domain= myapp.collab.net
rb_database_type=postgresql
rb_database_name=ctfrbdb
rb_database_user=ctfrbuser
rb_database_password=ctfrbpwd
rb_database_host= <database_hostname>
rb_database_port=5432

#CTF Information
ctf_base_url=https://myapp.collab.net
ctf_site_var_dir=/opt/collabnet/teamforge/var
```

iptables

This is the `/etc/sysconfig/iptables` output that will enforce the recommended security configuration.

```
 # Firewall configuration written by redhat-config-securitylevel
 # Manual customization of this file is not recommended.
 *filter
 :INPUT ACCEPT [0:0]
 :FORWARD ACCEPT [0:0]
 :OUTPUT ACCEPT [0:0]
 :RH-Firewall-1-INPUT - [0:0]
 -A INPUT -j RH-Firewall-1-INPUT
 -A FORWARD -j RH-Firewall-1-INPUT
 -A RH-Firewall-1-INPUT -i lo -j ACCEPT
 -A RH-Firewall-1-INPUT -p icmp --icmp-type any -j ACCEPT
 -A RH-Firewall-1-INPUT -p 50 -j ACCEPT
 -A RH-Firewall-1-INPUT -p 51 -j ACCEPT
 -A RH-Firewall-1-INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
 -A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j
  ACCEPT
 -A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 25 -j
  ACCEPT
 -A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j
  ACCEPT
 -A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j
  ACCEPT
 -A RH-Firewall-1-INPUT -j REJECT --reject-with icmp-host-prohibited
 COMMIT
```

login-config.xml

This is the sample `application-policy` block that you can copy into your `login-config.xml` file to support LDAP authentication.

Notes

Replace the default `application-policy` block of the `login-config.xml` file with this code, then make the modifications specified in *Set up LDAP integration for the TeamForge site* on page 576. Option values that must be modified are highlighted in bold.

- When the username is passed to the login module from TeamForge, it is translated into a DN for lookup on the LDAP server. The DN that is sent to the LDAP server is `<principalDNPrefix><username><principalDNSuffix>`.
- In this example `application-policy` block, the username is stored in the `People` organizational unit in the `dev.sf.net` domain. This is represented as `,ou=People,dc=dev,dc=sf,dc=net`
- This example contains a single `login-module` section. If you are authenticating against multiple LDAP servers, include one `login-module` section per LDAP server, with the required option values modified appropriately for each one. If the same username exists in more than one LDAP server, the instance on the first LDAP server will be used.

Sample code

```
        <application-policy name="SourceForge">
          <authentication>
            <login-module code="org.jboss.security.auth.spi.LdapLoginModule"
 flag="sufficient" >
              <module-option name="allowEmptyPasswords">false</module-option>
              <module-option name="principalDNPrefix">uid=</module-option>
              <module-option
 name="principalDNSuffix">,ou=People,dc=dev,dc=sf,dc=net</module-option>
              <module-option
 name="java.naming.factory.initial">com.sun.jndi.ldap.LdapCtxFactory</module-
option>
              <module-option name="java.naming.provider.url">ldap://
util.dev.sf.net:389/</module-option>
              <module-option
 name="java.naming.security.authentication">simple</module-option>
            </login-module>
          </authentication>
        </application-policy>
```

Sample code for Active Directory integration

Active Directory is not supported. However, these sample lines in the `login-config.xml` file may help you make it work for a simple AD setup, without complex directory structures requiring additional search parameters.

Set the values of `java.naming.provider.url`, `principalDNSuffix` and `rolesCtxDN` as appropriate to your site.

For more detailed instructions, see *http://www.jboss.org/community/wiki/LdapLoginModule*.

```
        <login-module code="org.jboss.security.auth.spi.LdapLoginModule"
 flag="required" >
              <module-option name="java.naming.provider.url">ldaps://
<server_name>:636/</module-option>
              <module-option name="allowEmptyPasswords">false</module-option>
              <module-option name="principalDNSuffix">@foo.bar.com</module-
option>
              <module-option name="rolesCtxDN">dc=Foo,dc=Bar,dc=Com</module-
option>
              <module-option name="matchOnUserDN">true</module-option>
              <module-option name="uidAttributeID">sAMAccountName</module-
option>
              <module-option name="roleAttributeID">memberOf</module-option>
```

```
            <module-option name="roleAttributeIsDN">true</module-option>
            <module-option name="roleNameAttributeID">name</module-option>
        </login-module>
```

The patch manifest file
The patch manifest file contains all the information about the patch.

Overview

The manifest file for each patch is named `manifest-[patch#]`. The manifest file is a text file containing a set of configuration tokens.

☞ **Note:** The first patch is named `manifest-1`.

Contents

The manifest file contains these tokens:

PATCH_LEVEL	The patch level which this patch provides.
	☞ **Note:** The *PATCH_LEVEL* value is used (along with information in `[DISTRIBUTION_DIR]/version/core-version.txt`) to fill in `[DISTRIBUTION_DIR]/conf/patches` with the current release and patch level. If `[DISTRIBUTION_DIR]/conf/patches` does not exist, it is created.
PATCH_DESCRIPTION	A description of the patch.
UNINSTALL_LIST	A list of RPMs to uninstall (using relative paths, comma separated).
INSTALL_LIST	A list of RPMs to install (using relative paths, comma separated).

☞ **Note:** Comments in the manifest file are identified by a leading hash (#).

postresql.conf
The `/var/lib/pgsql/9.3/data/postresql.conf` file controls the behavior of the PostreSQL database.

Shared port

- If the database and the datamart are using the same port (port 5432), this configuration file supports both.
- If the database and the datamart are on separate boxes, identical copies of this configuration file must exist in `/var/lib/pgsql/9.3/data` on both boxes.

```
# -----------------------------
# PostgreSQL configuration file
# -----------------------------
#
<snip>


#------------------------------------------------------------------------
# CONNECTIONS AND AUTHENTICATION
```

```
#-------------------------------------------------------------------------------

# - Connection Settings -

#listen_addresses = 'localhost'  # what IP address(es) to listen on;
     # comma-separated list of addresses;
     # defaults to 'localhost', '*' = all
     # (change requires restart)

listen_addresses = '127.0.0.1,<database_host_ip>'  # what IP address(es) to
 listen on;
#port = 5432 # (change requires restart)


<snip>
```

Separate ports

If the database and the datamart are on the same box but using separate ports, copies of this configuration file must
exist in both /var/lib/pgsql/9.3/data and /var/lib/pgsql/9.3/reports. Each copy must identify
a different port.

This file is in /var/lib/pgsql/9.3/data.

This copy of the file is in /var/lib/pgsql/9.3/reports. Note the different port number.

```
# ----------------------------
# PostgreSQL configuration file
# ----------------------------
#
<snip>

#-------------------------------------------------------------------------------
# CONNECTIONS AND AUTHENTICATION
#-------------------------------------------------------------------------------

# - Connection Settings -

#listen_addresses = 'localhost'  # what IP address(es) to listen on;
     # comma-separated list of addresses;
     # defaults to 'localhost', '*' = all
     # (change requires restart)

listen_addresses = '127.0.0.1,<database_host_ip>'  # what IP address(es) to
 listen on;
#port = 5632 # (change requires restart)


<snip>
```

Reference information about integrated applications

Here is some stuff you may need to know to work with integrated applications.

IntegratedAppSupport

CollabNet provides servlet helper libraries that take care of authenticating integrated application requests with
TeamForge. They also provide context information for each subsequent form processing within the application or
links clicked within the application.

The heart of this is the `IntegratedAppSupport` class, which must be called during every request of an integrated application. This class takes the HttpServletRequest and HttpServletResponse for each request and determines whether the user is already authenticated. It also provides project- and user-related information that can be used throughout the request.

☞ **Tip:** It is a good idea to store this as a ThreadLocal so that it can be used from anywhere in the application.

IntegratedAppSupport, on successful validation, provides several objects that can be used through the integrated application. .

Parameters

The method is specified within parentheses at the end of each parameter.

SoapSessionId	This identifies a user. It can be used for making any TeamForge webservice calls over SOAP. TeamForge expects that every call be associated with a valid SoapSessionId. This method lets you fetch the SoapSessionId for the current session of the user. (getSoapSessionId())
WebSessionId	This is the jsessionid for the current user. This is the servlet-container-specific id that might be used for some integrated applications. (getWebSessionId())
CtfBaseUrl	Retrieves the base URL for Collabnet TeamForge (getCtfBaseUrl()).
ProjectPath	Retrieves the TeamForgeProject path for this project. This would typically be "projects.<projectname>" (getProjectPath())
CollabNetSoap handle	Retrieves the handle for CollabNet SOAP (Refer to ctf_xxx_sdk.zip for SOAP calls that can be made using TeamForge) (getCollabNetSoap()). Documentation for the SOAP methods exposed are available in ctf_xxx_sdk.zip.
PluggableAppSoap handle	Retrieves the handle for pluggable application SOAP calls (getPluggableAppSoap()).Documentation for the SOAP methods exposed are available in ctf_xxx_sdk.zip.
RbacAppSoap	Retrieves the handle for RBAC SOAP Calls (getRbacAppSoap()).Documentation for the SOAP methods exposed are available in ctf_xxx_sdk.zip.
IntegratedAppId	Retrieves the linkid for this request. (getIntegratedAppId())
IntegratedAppPrefix	Retrieves the Prefix to use for go-urls, associations and linkifications (getIntegratedAppPrefix()).
IntegratedAppName	Retrieves the name of this integrated app (getIntegratedAppName()).
ProjectId	Retrieves the TeamForge project id (getProjectId()).
userSoapDO	Retrieves the SOAP data object information for the current user (getUserSoapDO()).

SOAP calls for integrated applications

When you integrate an application with TeamForge, you will need to make SOAP calls from the integrated application into TeamForge to respond to events on either side.

To integrate an application with TeamForge using the Integrated Application Framework, you have to implement the following methods for TeamForge to communicate with the integrated application.

The interface should be exposed via SOAP, and the endpoint for this interface should be defined by the `<endpoint>` tag in the XML Application configuration file.

☞ **Tip:** There is no direct API for managing users/groups from TeamForge to an integrated application. However, if your integrated application has the notion of users and user groups, you can listen to "user events" or "group events" through an event handler in TeamForge and call this interface using that data. You can also make SOAP calls from the integrated application directly into TeamForge to get user information.

Javadoc

For a complete description of all methods, see the *Javadoc for the Pebble integration service*.

SOAP calls for adding, editing or deleting an integrated application

You can use these SOAP calls to enable project administrators to add an integrated application to their projects, edit an integrated application, or delete it from their projects.

createProjectConfig

This method is called when an integrated application is added to a project from **Project Admin** > **Tools** > **Add Tool**.

Example	In `com/collabnet/pebble/adapter/ PebbleIntegrationService.java`, see the `createProjectConfig` method, line 48.

editProjectConfig

This method is called when editing an integrated application in a project from **Project Admin** > **Tools**.

Example	In `com/collabnet/pebble/adapter/ PebbleIntegrationService.java`, see the `editProjectConfig` method, line 202.

deleteProjectConfig

This method is called when deleting an integrated application in a project from **Project Admin** > **Tools**.

Example	In `com/collabnet/pebble/adapter/ PebbleIntegrationService.java`, see the `deleteProjectConfig` method, line 213.

SOAP calls for SCM in integrated applications

You can use these SOAP calls to enable users to change data in an integrated application based on actions in a TeamForge source code repository.

scmPreCommit

This serves as the pre-Commit hook for an integrated application. TeamForge will call this method if require-scm-integration flag is set to true in the integrated app xml descriptor. This would mean that a commit made into TeamForge will only succeed if this method returns a "true" (String).

☞ **Note:** All TeamForge ids and ids from other integrated applications are removed from this list even though it could be part of the original commit message. Also these ids are without the prefix. The integrated application can use these ids to find if they are valid for the particular commit and respond with a "true" or a "false" followed by an optional error message that can be displayed to the user who is making the commit.

scmPostCommit

This method is called after a commit is made and post processing needs to happen for that particular commit in the integrated application. A typical use case will be to store the commit information as part of some object within the

integrated application. Please note that any errors thrown as part of this will not stop the commit from happening as the commit has been already made.

SOAP calls for search in integrated applications

The getSearchResults method is called by TeamForge as part of doing a regular search. Use this method to include data from your integrated application in TeamForge search results.

Example

In `com/collabnet/pebble/adapter/PebbleIntegrationService.java`, see the `getSearchResults` method, line 409.

SOAP calls for project templates in integrated applications

You can use these SOAP calls to enable an integrated application to be included in project templates created by TeamForge.

createTemplate

This method is called when a TeamForge template is created on a project that contains this integrated application.

Example	In `com/collabnet/pebble/adapter/PebbleIntegrationService.java`, see the `createTemplate` method, line 334.

getTemplateMetadata

This method is called when displaying the template content in "Project Tools/Included in Template" section at the time of template creation.

Example	In `com/collabnet/pebble/adapter/PebbleIntegrationService.java`, see the `getTemplateMetadata` method, line 351.

getTemplateContent

This method is called when viewing the content of an existing template.

Example	In `com/collabnet/pebble/adapter/PebbleIntegrationService.java`, see the `getTemplateContent` method, line 367.

validateParametersForTemplatizedProject

This method is called to validate the configuration parameters provided when creating a project from a template.

Example	In `com/collabnet/pebble/adapter/PebbleIntegrationService.java`, see the `validateParametersForTemplatizedProject` method, line 381.

createTemplatizedProjectConfig

This method is called when a project is created from a template and this integrated application is part of the template. This is the equivalent of the "createProjectConfig" call except that it can be called when a project gets created from a template.

Example	In `com/collabnet/pebble/adapter/PebbleIntegrationService.java`, see the `createTemplatizedProjectConfig` method, line 395.

SOAP calls for objects in integrated applications

If your integrated application has the notion of objects, you can use these SOAP calls to enable users to act on those objects from within TeamForge.

getObjectTitle

Return the title for the provided object id. This will be used when associating an integrated app object to a TeamForge object. If the user enters an integrated application id, say, PB_20100602, the first part of it is the per-project prefix (PB) and the second part is the actual object id in the integrated application. This is the value passed into "getObjectTitle" so that it can return the Title for the object identifier. The title is then displayed as part of the association information in TeamForge.

Example	In `com/collabnet/pebble/adapter/` `PebbleIntegrationService.java`, see the `getObjectTitle` method, line 306.

findProjectIdForObject

Return the project to which the object belongs. This method is called only when the require-per-project-prefix parameter in the XML is false. When require-per-project-prefix is true, TeamForge knows the project to which the object belongs.

Example	In `com/collabnet/pebble/adapter/` `PebbleIntegrationService.java`, see the `findProjectIdForObject` method, line 294.

Pebble
Pebble integration service
Here's the WSDL for the Pebble blogging application's integration service.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="https://cu123.cubit.maa.collab.net:13000/
pebble/services/PebbleIntegrationService" xmlns:apachesoap="http://
xml.apache.org/xml-soap" xmlns:impl="https://
cu123.cubit.maa.collab.net:13000/pebble/services/PebbleIntegrationService"
 xmlns:intf="https://cu123.cubit.maa.collab.net:13000/pebble/services/
PebbleIntegrationService" xmlns:soapenc="http://schemas.xmlsoap.org/soap/
encoding/" xmlns:tns1="http://util.integratedapps.soap50.ce.collabnet.com"
 xmlns:tns2="qConfigParam" xmlns:tns3="http://com.collab.net/pebble/
XMLSchema" xmlns:tns4="qConfigParamDO" xmlns:tns5="qSearchInfo"
 xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:wsdlsoap="http://
schemas.xmlsoap.org/wsdl/soap/" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
<!--WSDL created by Apache Axis version: 1.4
Built on Apr 22, 2006 (06:55:48 PDT)-->
 <wsdl:types>
  <schema targetNamespace="qConfigParamDO" xmlns="http://www.w3.org/2001/
XMLSchema">
   <import namespace="http://com.collab.net/pebble/XMLSchema"/>
   <import namespace="http://util.integratedapps.soap50.ce.collabnet.com"/>
   <import namespace="qSearchInfo"/>
   <import namespace="http://xml.apache.org/xml-soap"/>
   <import namespace="https://cu123.cubit.maa.collab.net:13000/pebble/
services/PebbleIntegrationService"/>
   <import namespace="qConfigParam"/>
   <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
   <complexType name="qConfigParamDO">
    <sequence>
     <element name="defaultValue" nillable="true" type="soapenc:string"/>
```

```
        <element name="description" nillable="true" type="soapenc:string"/>
        <element name="displayType" nillable="true" type="soapenc:string"/>
        <element name="maxLength" type="xsd:int"/>
        <element name="name" nillable="true" type="soapenc:string"/>
        <element name="title" nillable="true" type="soapenc:string"/>
        <element name="valueType" nillable="true" type="soapenc:string"/>
       </sequence>
      </complexType>
     </schema>
     <schema targetNamespace="http://com.collab.net/pebble/XMLSchema"
    xmlns="http://www.w3.org/2001/XMLSchema">
        <import namespace="qConfigParamDO"/>
        <import namespace="http://util.integratedapps.soap50.ce.collabnet.com"/>
        <import namespace="qSearchInfo"/>
        <import namespace="http://xml.apache.org/xml-soap"/>
        <import namespace="https://cu123.cubit.maa.collab.net:13000/pebble/
    services/PebbleIntegrationService"/>
        <import namespace="qConfigParam"/>
        <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
        <complexType name="qArrayOfConfigParamDO">
         <complexContent>
          <restriction base="soapenc:Array">
           <attribute ref="soapenc:arrayType"
    wsdl:arrayType="tns4:qConfigParamDO[]"/>
          </restriction>
         </complexContent>
        </complexType>
     </schema>
     <schema targetNamespace="qConfigParam" xmlns="http://www.w3.org/2001/
    XMLSchema">
        <import namespace="http://com.collab.net/pebble/XMLSchema"/>
        <import namespace="qConfigParamDO"/>
        <import namespace="http://util.integratedapps.soap50.ce.collabnet.com"/>
        <import namespace="qSearchInfo"/>
        <import namespace="http://xml.apache.org/xml-soap"/>
        <import namespace="https://cu123.cubit.maa.collab.net:13000/pebble/
    services/PebbleIntegrationService"/>
        <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
        <complexType name="qConfigParam">
         <sequence>
          <element name="configParam" nillable="true"
    type="tns3:qArrayOfConfigParamDO"/>
          <element name="deleted" type="xsd:boolean"/>
          <element name="locked" type="xsd:boolean"/>
          <element name="parentProjectId" nillable="true" type="soapenc:string"/>
          <element name="projectId" nillable="true" type="soapenc:string"/>
          <element name="projectName" nillable="true" type="soapenc:string"/>
          <element name="projectUrl" nillable="true" type="soapenc:string"/>
          <element name="template" type="xsd:boolean"/>
          <element name="userId" nillable="true" type="soapenc:string"/>
         </sequence>
        </complexType>
     </schema>
     <schema targetNamespace="http://xml.apache.org/xml-soap" xmlns="http://
    www.w3.org/2001/XMLSchema">
        <import namespace="http://com.collab.net/pebble/XMLSchema"/>
        <import namespace="qConfigParamDO"/>
        <import namespace="http://util.integratedapps.soap50.ce.collabnet.com"/>
        <import namespace="qSearchInfo"/>
        <import namespace="https://cu123.cubit.maa.collab.net:13000/pebble/
    services/PebbleIntegrationService"/>
        <import namespace="qConfigParam"/>
        <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
        <complexType name="mapItem">
```

```
     <sequence>
      <element name="key" nillable="true" type="xsd:anyType"/>
      <element name="value" nillable="true" type="xsd:anyType"/>
     </sequence>
    </complexType>
    <complexType name="Map">
     <sequence>
      <element maxOccurs="unbounded" minOccurs="0" name="item"
 type="apachesoap:mapItem"/>
     </sequence>
    </complexType>
    <complexType name="Vector">
     <sequence>
      <element maxOccurs="unbounded" minOccurs="0" name="item"
 type="xsd:anyType"/>
     </sequence>
    </complexType>
   </schema>
   <schema targetNamespace="https://cu123.cubit.maa.collab.net:13000/pebble/
services/PebbleIntegrationService" xmlns="http://www.w3.org/2001/XMLSchema">
    <import namespace="http://com.collab.net/pebble/XMLSchema"/>
    <import namespace="qConfigParamDO"/>
    <import namespace="http://util.integratedapps.soap50.ce.collabnet.com"/>
    <import namespace="qSearchInfo"/>
    <import namespace="http://xml.apache.org/xml-soap"/>
    <import namespace="qConfigParam"/>
    <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
    <complexType name="ArrayOf_xsd_anyType">
     <complexContent>
      <restriction base="soapenc:Array">
       <attribute ref="soapenc:arrayType" wsdl:arrayType="xsd:anyType[]"/>
      </restriction>
     </complexContent>
    </complexType>
    <complexType name="ArrayOf_soapenc_string">
     <complexContent>
      <restriction base="soapenc:Array">
       <attribute ref="soapenc:arrayType" wsdl:arrayType="soapenc:string[]"/>
      </restriction>
     </complexContent>
    </complexType>
   </schema>
   <schema targetNamespace="qSearchInfo" xmlns="http://www.w3.org/2001/
XMLSchema">
    <import namespace="http://com.collab.net/pebble/XMLSchema"/>
    <import namespace="qConfigParamDO"/>
    <import namespace="http://util.integratedapps.soap50.ce.collabnet.com"/>
    <import namespace="http://xml.apache.org/xml-soap"/>
    <import namespace="https://cu123.cubit.maa.collab.net:13000/pebble/
services/PebbleIntegrationService"/>
    <import namespace="qConfigParam"/>
    <import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
    <complexType name="qSearchInfo">
     <sequence>
      <element name="attachment" nillable="true" type="soapenc:boolean"/>
      <element name="comment" nillable="true" type="soapenc:boolean"/>
      <element name="document" nillable="true" type="soapenc:boolean"/>
      <element name="keyWord" nillable="true" type="soapenc:string"/>
      <element name="plugAppName" nillable="true" type="soapenc:string"/>
      <element name="selectProject" nillable="true"
 type="impl:ArrayOf_xsd_anyType"/>
     </sequence>
    </complexType>
   </schema>
```

```
  </wsdl:types>

  <wsdl:message name="validateParametersForTemplatizedProjectResponse">

  </wsdl:message>

  <wsdl:message name="getPageComponentContentRequest">

     <wsdl:part name="oneTimeToken" type="soapenc:string"/>

     <wsdl:part name="projectId" type="soapenc:string"/>

     <wsdl:part name="projectIntegratedApplicationId"
type="soapenc:string"/>

     <wsdl:part name="pageComponentParam" type="soapenc:string"/>

     <wsdl:part name="rowsPerPage" type="soapenc:int"/>

     <wsdl:part name="locale" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="getSearchResultsRequest">

     <wsdl:part name="oneTimeToken" type="soapenc:string"/>

     <wsdl:part name="searchInfo" type="tns5:qSearchInfo"/>

     <wsdl:part name="locale" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="getPageComponentContentResponse">

     <wsdl:part name="getPageComponentContentReturn" type="soapenc:string"/
>

  </wsdl:message>

  <wsdl:message name="findProjectIdForObjectRequest">

     <wsdl:part name="oneTimeToken" type="soapenc:string"/>

     <wsdl:part name="objectId" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="createProjectConfigRequest">

     <wsdl:part name="oneTimeToken" type="soapenc:string"/>

     <wsdl:part name="iaConfig" type="tns2:qConfigParam"/>

  </wsdl:message>

  <wsdl:message name="createTemplateResponse">

  </wsdl:message>

  <wsdl:message name="getPageComponentParametersRequest">

     <wsdl:part name="oneTimeToken" type="soapenc:string"/>
```

```
    <wsdl:part name="projectId" type="soapenc:string"/>

    <wsdl:part name="locale" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="getSoapSessionIdRequest">

    <wsdl:part name="oneTimeToken" type="soapenc:string"/>

    <wsdl:part name="iAppSupport" type="xsd:anyType"/>

  </wsdl:message>

  <wsdl:message name="getObjectTitleResponse">

    <wsdl:part name="getObjectTitleReturn" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="editProjectConfigRequest">

    <wsdl:part name="oneTimeToken" type="soapenc:string"/>

    <wsdl:part name="iaConfig" type="tns2:qConfigParam"/>

  </wsdl:message>

  <wsdl:message name="findProjectIdForObjectResponse">

    <wsdl:part name="findProjectIdForObjectReturn" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="getConfigurationParametersRequest">

    <wsdl:part name="oneTimeToken" type="soapenc:string"/>

    <wsdl:part name="projectId" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="createProjectConfigResponse">

  </wsdl:message>

  <wsdl:message name="deleteProjectConfigRequest">

    <wsdl:part name="oneTimeToken" type="soapenc:string"/>

    <wsdl:part name="iaConfig" type="tns2:qConfigParam"/>

  </wsdl:message>

  <wsdl:message name="getConfigurationParametersResponse">

    <wsdl:part name="getConfigurationParametersReturn"
type="tns3:qArrayOfConfigParamDO"/>

  </wsdl:message>

  <wsdl:message name="getTemplateContentRequest">

    <wsdl:part name="oneTimeToken" type="soapenc:string"/>
```

```
    <wsdl:part name="templateName" type="soapenc:string"/>

    <wsdl:part name="locale" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="deleteProjectConfigResponse">

  </wsdl:message>

  <wsdl:message name="validateParametersForTemplatizedProjectRequest">

    <wsdl:part name="oneTimeToken" type="soapenc:string"/>

    <wsdl:part name="iaConfig" type="tns2:qConfigParam"/>

    <wsdl:part name="projectConfigurationParameters"
type="apachesoap:Map"/>

  </wsdl:message>

  <wsdl:message name="getObjectTitleRequest">

    <wsdl:part name="oneTimeToken" type="soapenc:string"/>

    <wsdl:part name="objectId" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="createTemplatizedProjectConfigResponse">

  </wsdl:message>

  <wsdl:message name="editProjectConfigResponse">

  </wsdl:message>

  <wsdl:message name="getTemplateMetadataResponse">

    <wsdl:part name="getTemplateMetadataReturn" type="apachesoap:Map"/>

  </wsdl:message>

  <wsdl:message name="getTemplateMetadataRequest">

    <wsdl:part name="oneTimeToken" type="soapenc:string"/>

    <wsdl:part name="locale" type="soapenc:string"/>

  </wsdl:message>

  <wsdl:message name="createTemplatizedProjectConfigRequest">

    <wsdl:part name="oneTimeToken" type="soapenc:string"/>

    <wsdl:part name="iaConfig" type="tns2:qConfigParam"/>

    <wsdl:part name="templateId" type="soapenc:string"/>

    <wsdl:part name="title" type="soapenc:string"/>

  </wsdl:message>
```

```
   <wsdl:message name="getSoapSessionIdResponse">

      <wsdl:part name="getSoapSessionIdReturn" type="soapenc:string"/>

   </wsdl:message>

   <wsdl:message name="getTemplateContentResponse">

      <wsdl:part name="getTemplateContentReturn" type="apachesoap:Map"/>

   </wsdl:message>

   <wsdl:message name="createTemplateRequest">

      <wsdl:part name="oneTimeToken" type="soapenc:string"/>

      <wsdl:part name="sourceProjectId" type="soapenc:string"/>

      <wsdl:part name="templateId" type="soapenc:string"/>

      <wsdl:part name="templateName" type="soapenc:string"/>

      <wsdl:part name="creatorId" type="soapenc:string"/>

   </wsdl:message>

   <wsdl:message name="getPageComponentParametersResponse">

      <wsdl:part name="getPageComponentParametersReturn"
type="impl:ArrayOf_soapenc_string"/>

   </wsdl:message>

   <wsdl:message name="getSearchResultsResponse">

      <wsdl:part name="getSearchResultsReturn" type="soapenc:string"/>

   </wsdl:message>

   <wsdl:portType name="PebbleIntegrationService">

      <wsdl:operation name="getSoapSessionId" parameterOrder="oneTimeToken
iAppSupport">

         <wsdl:input message="impl:getSoapSessionIdRequest"
name="getSoapSessionIdRequest"/>

         <wsdl:output message="impl:getSoapSessionIdResponse"
name="getSoapSessionIdResponse"/>

      </wsdl:operation>

      <wsdl:operation name="createProjectConfig"
parameterOrder="oneTimeToken iaConfig">

         <wsdl:input message="impl:createProjectConfigRequest"
name="createProjectConfigRequest"/>

         <wsdl:output message="impl:createProjectConfigResponse"
name="createProjectConfigResponse"/>

      </wsdl:operation>
```

```
      <wsdl:operation name="getConfigurationParameters"
parameterOrder="oneTimeToken projectId">

          <wsdl:input message="impl:getConfigurationParametersRequest"
name="getConfigurationParametersRequest"/>

          <wsdl:output message="impl:getConfigurationParametersResponse"
name="getConfigurationParametersResponse"/>

      </wsdl:operation>

      <wsdl:operation name="editProjectConfig" parameterOrder="oneTimeToken
iaConfig">

          <wsdl:input message="impl:editProjectConfigRequest"
name="editProjectConfigRequest"/>

          <wsdl:output message="impl:editProjectConfigResponse"
name="editProjectConfigResponse"/>

      </wsdl:operation>

      <wsdl:operation name="deleteProjectConfig"
parameterOrder="oneTimeToken iaConfig">

          <wsdl:input message="impl:deleteProjectConfigRequest"
name="deleteProjectConfigRequest"/>

          <wsdl:output message="impl:deleteProjectConfigResponse"
name="deleteProjectConfigResponse"/>

      </wsdl:operation>

      <wsdl:operation name="findProjectIdForObject"
parameterOrder="oneTimeToken objectId">

          <wsdl:input message="impl:findProjectIdForObjectRequest"
name="findProjectIdForObjectRequest"/>

          <wsdl:output message="impl:findProjectIdForObjectResponse"
name="findProjectIdForObjectResponse"/>

      </wsdl:operation>

      <wsdl:operation name="getObjectTitle" parameterOrder="oneTimeToken
objectId">

          <wsdl:input message="impl:getObjectTitleRequest"
name="getObjectTitleRequest"/>

          <wsdl:output message="impl:getObjectTitleResponse"
name="getObjectTitleResponse"/>

      </wsdl:operation>

      <wsdl:operation name="createTemplate" parameterOrder="oneTimeToken
sourceProjectId templateId templateName creatorId">

          <wsdl:input message="impl:createTemplateRequest"
name="createTemplateRequest"/>

          <wsdl:output message="impl:createTemplateResponse"
name="createTemplateResponse"/>
```

```
      </wsdl:operation>

      <wsdl:operation name="getTemplateMetadata"
parameterOrder="oneTimeToken locale">

         <wsdl:input message="impl:getTemplateMetadataRequest"
name="getTemplateMetadataRequest"/>

         <wsdl:output message="impl:getTemplateMetadataResponse"
name="getTemplateMetadataResponse"/>

      </wsdl:operation>

      <wsdl:operation name="getTemplateContent" parameterOrder="oneTimeToken
templateName locale">

         <wsdl:input message="impl:getTemplateContentRequest"
name="getTemplateContentRequest"/>

         <wsdl:output message="impl:getTemplateContentResponse"
name="getTemplateContentResponse"/>

      </wsdl:operation>

      <wsdl:operation name="validateParametersForTemplatizedProject"
parameterOrder="oneTimeToken iaConfig projectConfigurationParameters">

         <wsdl:input
message="impl:validateParametersForTemplatizedProjectRequest"
name="validateParametersForTemplatizedProjectRequest"/>

         <wsdl:output
message="impl:validateParametersForTemplatizedProjectResponse"
name="validateParametersForTemplatizedProjectResponse"/>

      </wsdl:operation>

      <wsdl:operation name="createTemplatizedProjectConfig"
parameterOrder="oneTimeToken iaConfig templateId title">

         <wsdl:input message="impl:createTemplatizedProjectConfigRequest"
name="createTemplatizedProjectConfigRequest"/>

         <wsdl:output message="impl:createTemplatizedProjectConfigResponse"
name="createTemplatizedProjectConfigResponse"/>

      </wsdl:operation>

      <wsdl:operation name="getSearchResults" parameterOrder="oneTimeToken
searchInfo locale">

         <wsdl:input message="impl:getSearchResultsRequest"
name="getSearchResultsRequest"/>

         <wsdl:output message="impl:getSearchResultsResponse"
name="getSearchResultsResponse"/>

      </wsdl:operation>

      <wsdl:operation name="getPageComponentParameters"
parameterOrder="oneTimeToken projectId locale">

         <wsdl:input message="impl:getPageComponentParametersRequest"
name="getPageComponentParametersRequest"/>
```

```
        <wsdl:output message="impl:getPageComponentParametersResponse"
  name="getPageComponentParametersResponse"/>

      </wsdl:operation>

      <wsdl:operation name="getPageComponentContent"
  parameterOrder="oneTimeToken projectId projectIntegratedApplicationId
  pageComponentParam rowsPerPage locale">

        <wsdl:input message="impl:getPageComponentContentRequest"
  name="getPageComponentContentRequest"/>
```

pebble-dep.xml

The `pebble-dep.xml` file, also known as the Pebble deployment configuration file, contains the data that Pebble needs to interact with the TeamForge site.

A sample pebble-dep.xml file for the REST service type

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE integrated-application
  PUBLIC "-//CollabNet, Inc.//DTD Integrated Application Descriptor 1.0//EN"
  "http://schema.open.collab.net/sfee50/dtd/sf-pluggable-deploy-
descriptor_1_0.dtd">
<integrated-application>
    <name>Pebble Blog</name>
    <baseurl>https://cu064.cloud.sp.collab.net:13001/pebble/index.jsp</
baseurl>
    <gourl>https://cu064.cloud.sp.collab.net:13001/pebble/gourl/%p/%o</gourl>
    <endpoint>https://cu064.cloud.sp.collab.net:13001/pebble/services/rest/
ctfapi</endpoint>
 <servicetype>REST</servicetype>
</integrated-application>
```

A sample pebble-dep.xml file for the SOAP service type

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE integrated-application
  PUBLIC "-//CollabNet, Inc.//DTD Integrated Application Descriptor 1.0//EN"
  "http://schema.open.collab.net/sfee50/dtd/sf-pluggable-deploy-
descriptor_1_0.dtd">
<integrated-application>
    <name>Pebble Blog</name>
    <baseurl>https://cu177.cloud.maa.collab.net:13001/pebble/index.jsp</
baseurl>
    <gourl>https://cu177.cloud.maa.collab.net:13001/pebble/gourl/%p/%o</gourl>
<endpoint>https://cu177.cloud.maa.collab.net:13001/pebble/services/
PebbleIntegrationService</endpoint>
    <servicetype>SOAP</servicetype>
</integrated-application>
```

pebble-app.xml

The `pebble-app.xml` file, also known as the Pebble application configuration file, contains the text that the Pebble application displays in the TeamForge user interface.

This is an example of a default (unedited) `pebble-app.xml` file. To create your own integrated application config file, copy this one into a new file and replace the values with the values appropriate for the application you are integrating.

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE integrated-application
  PUBLIC "-//CollabNet, Inc.//DTD Integrated Application Descriptor 1.0//EN"
```

```
   "http://schema.open.collab.net/sfee50/dtd/sf-pluggable-application-
descriptor_1_0.dtd">
<integrated-application>
    <name>Pebble Blog</name>
    <description>l10n.application.description</description>
    <permissions>
        <permission dapMappedTo="View">Blog Reader</permission>
        <permission>Blog Contributer</permission>
        <permission>Blog Publisher</permission>
        <permission>Blog Owner</permission>
    </permissions>
    <prefix>PB</prefix>
    <id-pattern></id-pattern>
    <require-per-project-prefix>true</require-per-project-prefix>
    <require-scm-integration>true</require-scm-integration>
    <is-search-supported>false</is-search-supported>
    <!-- Page components for Integrated apps is not implemented for Alpha -->
    <page-component>
      <require-page-component>true</require-page-component>
      <page-component-details>
        <inputtype>text</inputtype>
        <resultformat>html</resultformat>
        <description>l10n.pce.description</description>
        <title>l10n.pce.title</title>
      </page-component-details>
    </page-component>
    <config-parameters>
        <!-- Pebble Configuration Parameters -->
        <param>
            <title>l10n.blogname.title</title>
            <name>blogName</name>
            <description>l10n.blogname.description</description>
            <defaultvalue>My Blog</defaultvalue>
            <displaytype valuetype="String" maxlength="25">TEXT</displaytype>
            <editable>false</editable>
        </param>
        <param>
            <title>l10n.blogdescription.title</title>
            <name>blogDescription</name>
            <description>l10n.blogdescription.description</description>
            <defaultvalue>My Awesome Blog</defaultvalue>
            <displaytype valuetype="String" maxlength="40">TEXT</displaytype>
            <editable>true</editable>
        </param>
        <param>
            <title>l10n.richtexteditor.title</title>
            <name>richTextEditorEnabled</name>
            <description>l10n.richtexteditor.description</description>
            <defaultvalue>checked</defaultvalue>
            <displaytype valuetype="String">CHECKBOX</displaytype>
            <editable>true</editable>
        </param>
        <param>
            <title>l10n.noofrecentblogentries.title</title>
            <name>recentBlogEntries</name>
            <description>l10n.noofrecentblogentries.description</description>
            <defaultvalue>3</defaultvalue>
            <displaytype valuetype="String">SELECT</displaytype>
            <option name="3">l10n.three.value</option>
            <option name="5">l10n.five.value</option>
            <option name="7">l10n.seven.value</option>
            <option name="9">l10n.nine.value</option>
            <editable>true</editable>
        </param>
```

```xml
        </config-parameters>
    <bundles>
        <bundle locale="en">
            <key name="l10n.application.description">Pebble Blog App</key>
            <key name="l10n.pce.description">Display Blog Title for Given
 Date.</key>
            <key name="l10n.pce.title">Enter Blog Date (in yyyy-mm-dd)</key>
            <key name="l10n.blogname.title">Blog Name</key>
            <key name="l10n.blogname.description">Please provide a name for
 the Blog. This appears on all blog pages</key>
            <key name="l10n.blogdescription.title">Blog Description</key>
            <key name="l10n.blogdescription.description">Please provide a
 description for the Blog. This appears below blog name on all pages</key>
            <key name="l10n.richtexteditor.title">Rich Text Editor</key>
            <key name="l10n.richtexteditor.description">Enable Rich Text
 Editor for comments and Blog entries?</key>
            <key name="l10n.noofrecentblogentries.title">Recent Blog Entries</
key>
            <key name="l10n.noofrecentblogentries.description">How many recent
 blog entries do you want to see in the home page?</key>
            <key name="l10n.three.value">3</key>
            <key name="l10n.five.value">5</key>
            <key name="l10n.seven.value">7</key>
            <key name="l10n.nine.value">9</key>
        </bundle>
    </bundles>
</integrated-application>
```

Extend TeamForge

TeamForge provides you with the interrelated extension features to suit your organization's specific needs.

Building on the TeamForge platform

TeamForge is an open platform that you can build on to provide functionality beyond what's in the basic application.

Use some or all of the interrelated extension features to tailor TeamForge to your organization's unique needs.

Event handlers: Create custom work flows in TeamForge

When you add an event handler to your TeamForge site, you can automatically react to system events in ways that help your site's project members or administrators.

How TeamForge event handlers work

The TeamForge custom event handling framework allows third-party event handlers to register for TeamForge-specific application events and notifies them whenever such an event occurs.

The event handling framework implements an extended flavor of the observer pattern. It TeamForge application events are triggered whenever a property of a TeamForge object (e.g. tracker item, discussion item, wiki page) has been changed or is going to be changed, if no event handler objects (i.e. blocks the event).

For example, you can block deletions of projects for all users, add a comment to a tracker item whenever an association has been modified, or design your own tracker workflow engine.

Writing custom event handlers requires at least some basic knowledge in the Java programming language or (if you use the examples shipped with this post) a script language that is installed on the TeamForge server, such as shell, Python, or Perl. For these instructions, we'll assume that you are familiar with basic programming techniques.

The event handler framework differentiates between two types of events:

• **Asynchronous**: If a handler registers for asynchronous events, it is informed that a change has just happened. The handler can decide to trigger further changes by calling TeamForge web services, but it cannot block the change because it has already happened.

Asynchronous event handlers are good for triggering system events, such as changing an artifact status or sending an email. See *Using an asynchronous event handler: trigger follow-up events* on page 850.

• **Synchronous**: If a handler has registered for synchronous events, it gets informed whenever a change has been anticipated by a user. It can examine the properties that should be changed and decide whether to accept the change or block it. A synchronous event handler cannot trigger further changes on the currently processed object, since other handlers in the event handler chain must also have the chance to block the anticipated change.

A synchronous event handler is the appropriate way to show an alert directly in the TeamForge UI, for example. See *Using a synchronous event handler: Send event handler output to the TeamForge UI* on page 849.

Technically, all event handlers have to be part of a Java archive (JAR) file with a TeamForge specific deployment descriptor that describes which events should be intercepted. This JAR file then has to be uploaded to the TeamForge application server. No restart is necessary, but the event handling cache has to be refreshed.

In practice, you can customize a TeamForge site's behavior without any knowledge of Java if you can write scripts in a language that can deal with environment variables, write to standard out/error (to control what will be displayed in the TeamForge UI as result of the handler's execution) and control the return code (to decide whether to block the event or not).

We will show you how to come up with your own custom event handlers based on two examples.

Event handler example: Comment on associations

This event handler adds a comment to a tracker item whenever an association is added to or deleted from this tracker item.

This example illustrates how to intercept a specific event, trigger a follow-up action by calling TeamForge web services, and add a comment based on the formatting template which is specified as part of a property file.

The code for example one can be found *here*.

👉 **Tip:** You can treat a JAR file like a zip file: just extract it with your favorite zip program and have a look at the files that are part of it.

1. When you extract the JAR file you'll find a structure like this:

    `com/vasoftware/sf/plugin`

    This directory contains the class file of your event handler. There may be additional directories containing Java class files. If you like to include Java libraries, you have to unpack their JAR files and add their class files (including directory structure) in the event handler JAR.

    `META-INF/event.xml`

    This file contains the events and operations your handler class will intercept. It is also common to have additional files in the META-INF directory, such as property files to control the behavior of the event handler.

    This example event handler is provided as-is (i.e. not supported as part of any TeamForge release). As with all event handlers, use it at your own risk. CollabNet cannot guarantee any SLAs on third-party code.

    👉 **Tip:** If you want to change the behavior of custom event handlers at runtime (without redeploying the JAR file), you may want to look at the *TeamForge integration data API*. For the purpose of this example, we will stick with property files.

2. In the `event.xml` file, you will find some lines like this:

    ```
    <event api="6.1" mode="asynchronous">
        <type>relationship</type>
        <operation>*</operation>

     <handler>com.vasoftware.sf.plugin.AsynchronousRelationshipEventListener</
    handler>
    </event>
    ```

    This tells the event handling framework that the class `AsynchronousRelationshipEventListener` is responsible for intercepting events of type "relationship" (aka associations) for every possible operation. The handler will be called after the event has happened (asynchronous mode) and the passed data structures will be compatible with the events format defined in TeamForge 6.0 (in other words, the event handler extends the `EventHandler60` class). If you only want to intercept certain operations, you can specify those instead of the wildcard character (`*`). Supported operations are usually create, update, move and delete, but every event can define its own operations.

3. The `config.properties` file is used to control the formatting of the comment that gets added when an association has been modified.

    The initialize-method of the handler class (`AsynchronousRelationshipEventListener`) shows how property files can be parsed within a custom event handler.

4. By default, the user triggering the event is also the user executing the event handler. If you want to run your event handler with a different user account, specify it in the user element, like this:

    ```
    <event api="6.1" mode="synchronous">
        <type>*</type>
        <operation>*</operation>

     <handler>com.vasoftware.sf.plugin.AsynchronousRelationshipEventListener</
    handler>
    ```

```
        <user>foo</user>
</event>
```

> ☞ **Note:** Be careful with this option, because running code on behalf of a different user opens the door for all kind of exploits if you do not check the user's input properly. Also, this option will not allow you to access the original user's session id any more, but you can always create a new session with a super user account (credentials saved in a property file) if you have to.

Event handler example: Execute a hook script

When a TeamForge event arrives, this event handler looks to see whether there is a script in the TeamForge file system with the name/operation of the event, and then calls that script with all information from the event contained within environment variables.

This example illustrates how to intercept arbitrary TeamForge events, examine the event's properties, map them to system environment variables and call a script in the file system with a name corresponding to the intercepted event.

You can use this event handler to customize your TeamForge site's behavior without any knowledge of the Java programming language as long as you can write scripts in a language that can deal with environment variables, write to standard out/error (to influence what will be displayed in TeamForge's UI as result of the handler's execution) and influence the return code (to decide whether to block the event or not).

The code for this example can be found *here*.

> ☞ **Tip:** You can treat a JAR file like a zip file: just extract it with your favorite zip program and have a look at the files that are part of it.

1. When you extract the JAR file you'll find a structure like this:

   ```
   com/collabnet/ctf/events
   ```
   This directory contains the class file of your event handler. There may be additional directories containing Java class files. If you like to include Java libraries, you have to unpack their JAR files and add their class files (including directory structure) in the event handler JAR.

   ```
   META-INF/event.xml
   ```
   This file contains the events and operations your handler class will intercept. It is also common to have additional files in the META-INF directory, such as property files to control the behavior of the event handler.

   This example event handler is provided as-is (i.e. not supported as part of any TeamForge release). As with all event handlers, use it at your own risk. CollabNet cannot guarantee any SLAs on third-party code.

   > ☞ **Tip:** If you want to change the behavior of custom event handlers at runtime (without redeploying the JAR file), you may want to look at the *TeamForge integration data API*. For the purpose of this example, we will stick with property files.

2. In the `event.xml` file, you will find some lines like this:

   ```
   <event api="6.1" mode="asynchronous">
    <type>*</type>
    <operation>*</operation>
    <handler>com.collabnet.ctf.events.AsynchronousHookScriptEventListener</
   handler>
   </event>

   <event api="6.1" mode="synchronous">
    <type>*</type>
    <operation>*</operation>
    <handler>com.collabnet.ctf.events.SynchronousHookScriptEventListener</
   handler>
   </event>
   ```

   These lines tell TeamForge to register two event handlers, one asynchronous (`AsynchronousHookScriptEventListener`) and one synchronous

(`SynchronousHookScriptEventListener`) for arbitrary events (wildcard *). It also tells TeamForge that the handler classes will extend the `EventHandler60` interface.

3. By default, the user triggering the event is also the user executing the event handler. If you want to run your event handler with a different user account, specify it in the user element, like this:

```
<event api="6.1" mode="synchronous">
     <type>*</type>
     <operation>*</operation>
     <handler>com.collabnet.ctf.events.SynchronousHookScriptEventListener</
handler>
     <user>foo</user>
</event>
```

> ☞ **Note:** Be careful with this option, because running code on behalf of a different user opens the door for all kind of exploits if you do not check the user's input properly. Also, this option will not allow you to access the original user's session id any more, but you can always create a new session with a super user account (credentials saved in a property file) if you have to.

It is possible to register multiple handlers for different events, but you can also use one handler to intercept both synchronous and asynchronous events.

Event handler example: Hook scripts
These sample hook scripts should give you an idea how custom event handlers can be written. Feel free to adjust them to your own needs.

> ☞ **Note:** These sample event handlers are not officially supported by CollabNet and must be used at your own risk.

- Hooks must be owned by `sf-admin` for security, and must have the executable bit set.
- To configure a site to prevent projects being deleted, we could create this file: `/opt/collabnet/teamforge/hooks/synchronous/project_delete`

```
#!/bin/sh
echo Sorry, projects cannot be deleted on this site 1>&2
exit 1
```

- To automatically create an initial directory structure in an SVN repository when the repository is created, you might create this file: `/opt/collabnet/teamforge/hooks/asynchronous/repository_create`

```
#!/bin/sh
/usr/bin/svn
mkdirhttp://localhost/svn/repos/${tf_original_RepositoryDirectory:9:999}/
trunk
http://localhost/svn/repos/${tf_original_RepositoryDirectory:9:999}/tags
http://localhost/svn/repos/${tf_original_RepositoryDirectory:9:999}/
branches
-m "Inital Structure"
--username admin --password mypassword --non-interactive --no-auth-cache
exit 0
```

Extend `EventHandler60` for your TeamForge event handler classes
All event handlers (synchronous and asynchronous) have to extend
`com.vasoftware.sf.events.EventHandler60`.

This is how our example "Comment on associations" event handler does it:

```
package com.vasoftware.sf.plugin;
...
public class AsynchronousRelationshipEventListener extends EventHandler60 {
...
```

```
}
```

The folloiwng table lists the JAR file location for different versions of TeamForge that contains EventHandler60.

TeamForge Version	JAR File
6.1	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/sfevents.jar`
6.1.1	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar`
6.2	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar`
6.2.0.1	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar`
7.0	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar`
7.1	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/sfevents.jar`
8.0	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/events-api.jar`
8.0.0.1	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/events-api.jar`
8.1	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/events-api.jar`
8.1.0.1	`/opt/collabnet/teamforge/dist/deployments/saturn.ear/lib/events-api.jar`
16.3	`/opt/collabnet/teamforge/dist/deployments/lib/events-sdk.jar`
16.7 and later	`/opt/collabnet/teamforge/dist/deployments/lib/event.jar`

- EventHandler60 has one abstract method you have to override:

```
@Override
public void processEvent() throws Exception {
...
}
```

- This method does not take any parameters, so how do we access the relevant information?

  There are some interesting methods of the base class (`EventHandler60`) you can call:

getSessionKey: Returns a session id of the user who is going to (synchronous handler) / has triggered (asynchronous handler) the event we just intercepted. If you used the user element in event.xml, it will contain a session id for the user you specified there.

getEventContext: Returns a data structure containing the event type, operation, project, comment and user name

getOriginalData: In case of a synchronous event handler, this will return a representation of the object the event is going to change. In case of an asynchronous event handler, this will return the representation of the object before it was changed by the event. The data structure used to represent the object is the same that would have been used in CollabNet's SOAP API.

getUpdatedData: In case of a synchronous event handler, this will return a representation of the object how it will look after the event has happened (you can still block it). In case of an asynchronous event handler, this will return the representation of the object after it was changed by the event.

Let's assume a user wants to change the priority of a tracker item from 3 to 4. If you have registered a synchronous event handler, this one is triggered before the change can actually be performed. getOriginalData returns an ArtifactSoapDO object of the tracker item with the priority field set to 3. getUpdatedData contains an ArtifactSoapDO object of the tracker item with the priority field set to 4.

- If you block the event (by throwing an exception), the change does not happen and the user is presented with an error message. (See next section for how to influence this error message.)
- If you do not block the event (by just returning from the processEvent method), all registered asynchronous handlers are called. getOriginalData and getUpdatedData contain exactly the same objects as in the synchronous case. However, the semantic is different: They are no longer representing the current and anticipated state, but the previous and current state of the object in question.
- The following code snippet (taken from our "Hook script" event handler example) shows how to retrieve all information available to an event handler.

```
String type = getEventContext().getType();
String operation = getEventContext().getOperation();
String projectId = getEventContext().getProjectId();
String comment = getEventContext().getComment();
String userName = getEventContext().getUsername();
String originalDataClassName = getOriginalData().getClass().toString();
String updatedDataClassName = getUpdatedData().getClass().toString();
Object originalData = getOriginalData();
Object updatedData = getUpdatedData();
```

Using a synchronous event handler: Send event handler output to the TeamForge UI
When we have extracted all data available to the event handler, how do we interact with the user interface?

☞ **Note:** Only synchronous event handlers can directly communicate with the UI, because if the event has already happened (as it has, in the case of asynchronous handlers), the user who triggered the event may already have been logged out.

- You use three independent actions to interact with the TeamForge UI:
  a) Add a success message to the UI that gets displayed as the result of the action just triggered by the user.

     This can be done by calling the addSuccessMessage method of the EventHandler base class (see SynchronousHookScriptEventListener.java of example two for details).
  b) Add an error message to the UI that gets displayed as the result of the action just triggered by the user.

     This can be done by calling the addErrorMessage method of the EventHandler base class.
  c) Block the event you intercepted.

     This can be done by throwing an exception in the processEvent method. The payload of your exception will be displayed in the UI.

All three forms of UI feedback can be used in combination. For example, it is possible to display an error message even if you did not block the event, and it is possible to show many error and success messages together.

- What happens if the event in question was not triggered by a user logged into the TeamForge Web UI but by a client using the TeamForge web services?

  In this case, error and success messages do not reach the SOAP client. However, the payload of the exception object thrown when the event was blocked is delivered as part of the SOAP fault element.

- While synchronous event handlers enable you to block events and/or to provide additional feedback to the currently logged in user, they should not be used to trigger follow-up actions (like changing TeamForge artifacts or interacting with external systems).

  Remember that these handlers are running in the main TeamForge event loop and nothing else will happen until you return from the `processEvent` method, so return as fast as you can.

Using an asynchronous event handler: trigger follow-up events

Use an asynchronous event handler to communicate with TeamForge, external systems, processes or system resources.

> ☞ **Important:** To avoid accidentally locking the main TeamForge event queue down (and essentially rendering the system unusable), use only asynchronous event handlers (not synchronous event handlers) to trigger events.

Interacting with TeamForge is done as you would do it if you had to write a Java program to interact with TeamForge using its web services API. The only difference is that you will connect to `localhost` (since your handler is running locally) and that you already have a valid session ID.

- You do not have to include the SOAP SDK classes in your event JAR file, because they are already in the TeamForge class path.

  This code snippet extracted from our association converter example (*Event handler example: Comment on associations* on page 845) shows how to do it:

  ```
  ITrackerAppSoap trackerClient = (ITrackerAppSoap)
   ClientSoapStubFactory.getSoapStub(
  ITrackerAppSoap.class, "http://localhost:8080");
  ...
  ArtifactSoapDO artifact = trackerClient.getArtifactData(getSessionKey(),
   originId);
  trackerClient.setArtifactData(getSessionKey(), artifact, finalComment,
   null, null, null);
  ```

  > ☞ **Tip:** You may use external libraries in your event handler by placing their `.class` files into your event JAR file. The only tricky part is if TeamForge is using a different version of this library (which will take precedence). In this case, you would have to recompile your library with a different package namespace.

- Using the session key provided by the event handler is actually only going to work if the SOAP call you are using is not throwing an exception.

  The session ID passed into your handler is associated with an already running transaction that will be aborted if an exception is thrown as part of this session. Part of rolling back the transaction is rolling back the JVM's call stack which contains your event handling code, so you will not be able to catch the web services exception. If you like to to handle web service exceptions, you have to create your own session id by logging into TeamForge again by calling `ICollabNetSoap.login` with some credentials stored as part of your handler. (You can store them in a property file in the `META-INF` directory.)

Best practices for working with custom TeamForge event handlers

In general, watch out for deadlocks and favor asynchronous over synchronous event handlers.

Beware of deadlocks.

Having custom event handlers that modify other objects can be dangerous if there it is possible for that handler or another handler to chain in the opposite direction. An example of this is an event handler that updates a task when an associated artifact is updated and updates the artifact when the associated task is updated. It is possible for two users to modify each object at the same time causing the two event handlers to wait on each other. The task handler would

have a lock on the task bean in the application server while the artifact handler would have a lock on the artifact bean. When the custom event handlers fired, they would wait for the locks to be released but since the two threads have the locks each other needs and are waiting on the opposite objects, a deadlock would occur.

Asynchronous is safer

Custom event handlers will be the least worrisome when they are responsible for data validation or secondary object creation (or association creation). Object modification is possible but adds greater complexity due to the risks involved with locking multiple objects across many threads. If you are unsure, use asynchronous handlers to modify objects instead since the lock on the original object will be gone by the time the asynchronous handler is executed.

Calling and waiting for synchronous hooks currently doesnt have a timeout. As long as your synchronous hook is running, the whole TeamForge site will be blocked for all users accessing the site. Some events trigger other events. For example creating a project actually calls the create project hook, wiki page hooks, and so on. Badly written or slow hooks can cripple a site.

Write to a file

Write your diagnostics messages in a file and not on stdout/stderr, since TeamForge does not read from stdout/stderr before the script completes. In the case of synchronous hooks, this could lead to a situation where the script blocks because the pipes buffer between the script process and the TF process is completely filled.

No cascading

Due to the nature of custom event handling, custom events cannot cascade. This means that if a custom event handler catches an event and creates an object that it or another custom event handler would normally process, the event bypasses the custom event handlers. This is to prevent looping and infinite object creation. While there are ways for event handlers to avoid this, it would be a fairly difficult task since all of your event handlers would have to use a circular event detection algorithm. Rather than adding that complexity, we just eliminated the possibility.

Event handler life cycle

For every single call to the processEvent method, a new object of your class will be instantiated. A best practice to avoid costly reinitialization every time (remember that the TeamForge event loop thread is blocked while you are doing this) is to delegate all synchronization work to a method you always call in your constructor which checks a static variable whether the initialization has already been done and if not, just returns without any further action (code snippet from example one):

```
private static boolean initialized = false;
public  AsynchronousRelationshipEventListener() {
initialize();
}

private synchronized void initialize() {
if (initialized ) {
return;
}
initialized = true;
// proceed with initialization
...
}
```

Logging in into TeamForge and initializing network connections file resources are costly operations that should be handled in such a method instead of doing it all over again.

Event spooling

While it is true that asynchronous handlers may consume considerably more time than synchronous ones, there is only one thread for those handlers, so events may queue up if you do expensive operations. A best practice is to capture the event in your asynchronous event handler, write all necessary information to the local file system (comparable to a

mail spooling directory) and return. At the same time, you can have a separate application reading from the spooling directory. This way, you never get into a situation where you miss TeamForge events, or things queue up just because you run into a blocking operation.

Incremental changes

The event handler parser is really picky on the exact format of your JAR file. A best practice is to base your work on an already existing event handler and then adapt it to your own needs by doing incremental changes while checking whether it still works.

Watch out for loops

Your follow-up actions may trigger your handler to be called again. You have to protect your handler from an infinite update loop if that happens. A best practice is to add a check to your event handler to see whether the user initiating the event is the same user you are using to perform follow-up actions.

Roll back sparingly

Throwing an exception in a synchronous event handler blocks the intercepted event and rolls back the transaction associated with the change. Rolling back transactions also means that the data the user entered is not saved. If this happens accidentally due to a wrongly programmed event handler, it can be frustrating to your users, so make sure that you only throw exceptions in your handler code when you really want to enforce the rollback.

Catch errors generically

It is quite easy to miss an exception you did not expect (like a null pointer exception, parsing exception, time out exception, any other malfunction in your own code). A best practice is to introduce a generic catch block in your handler and only rethrow the exception if it was an intended exception (see SynchronousHookScriptEventListener):

```
} catch (Exception e) {
 if (!intendedException) {
 log.error("Exception occured: " + e.getMessage(), e);
 } else {
...
 throw e;
 }
 }
```

Reference information about event handlers

Here is some stuff you may need to know to work with event handlers.

event.xml

Here is an example of an `event.xml` file. Use it to create a new `event.xml` file for your custom event handler in the `META-INF` directory of your `jar` file.

In this example, the CreateTestCaseWorkflow class will be invoked only on artifact creation and will run after the artifact creation is successfully committed. The EventTestListener will run for every single event and will run synchronously thereby having the opportunity to cancel the event.

```
<?xml version="1.0"?>
<!DOCTYPE event-handler
 PUBLIC "-//VA Software, Inc.//DTD Data Object 1.1//EN"
 "http://schema.vasoftware.com/sf/dtd/sf-event-handler_1_0.dtd">

<event-handler>
    <event api="6.1" mode="asynchronous">
        <type>artifact</type>
        <operation>create</operation>
        <user>admin</user>
```

```
 <handler>com.collabnet.ctf.events.CreateTestCaseWorkflow</
handler>
    </event>

  <event api="6.1" mode="synchronous">
       <type>*</type>
       <operation>*</operation>
 <handler>com.collabnet.ctf.events.EventTestListener</handler>
    </event>
</event-handler>
```

DTD for custom event handler

An event handler works by subscribing to system events and then responding when event occurs. You subscribe to events by placing an XML file in the custom event handler JAR that describes what events are being monitored and who is doing the monitoring.

Here is the current DTD.

```
<!ELEMENT event-handler (description?, event+)>

<!-- Optional description for the Event Handler -->
<!ELEMENT description (#PCDATA)>

<!ELEMENT event (type, operation, user?, handler)>

<!-- This is the API level that the event handler expects the
 ObjectSoapDO objects to be marshaled for -->
<!ATTLIST event api (6.1|6.2) #REQUIRED>

<!--
    Whether the event handler should run synchronously (allows it
 to cancel the event) or asynchronously after the event
    has completed and is committed to the database.
-->
<!ATTLIST event mode (synchronous|asynchronous) #REQUIRED>

<!-- The object type being handled (e.g., artifact, task,
 document). '*' is all object types. -->
<!ELEMENT type (#PCDATA)>

<!-- This is the operation that is being listened for (e.g.,
 create. delete. move, update). '*' is all operations. -->
<!ELEMENT operation (#PCDATA)>

<!--
    If the event handler needs to run as a different user (ie., a
 site admin account) the username should be
    passed in this attribute.  If the username is not valid the
 event handler will throw an abort exception.  That
    means that if the handler is processing synchronously the
 event will be canceled and the user will get an
    exception message.
-->
<!ELEMENT user (#PCDATA)>

<!--
   Name of a fully qualified class to handle the event.
-->
<!ELEMENT handler (#PCDATA)>
```

ENABLE_UI_FOR_CUSTOM_EVENT_HANDLERS

To support branding and customization changes, set the *ENABLE_UI_FOR_CUSTOM_EVENT_HANDLERS* variable to true.

Values

true or false

Default

true

API guide: Use TeamForge features via SOAP

When you set up another application to communicate with the CollabNet TeamForge API, the remote app's users can do almost anything they can do in the CollabNet product's Web interface.

The CollabNet TeamForge web services offer external access to core application functionality and data, such as work tracking, task management, and hardware provisioning.

See *CollabXchange* for examples of successful integrations using CollabNet APIs.

Get started with the CollabNet TeamForge SOAP API

A CollabNet TeamForge plugin can provide any of the features a user can access through the Web interface.

1. Browse the *CollabNet user help* to get a full picture of the functionality you can deliver with the SOAP API.
2. Get an idea of the sorts of applications that have been developed.

   - Look at the applications available for download on *CollabXchange*.
   - Review the reference applications on the *CollabNet community site*. These samples illustrate some of the more common methods for using TeamForge via the SOAP API. They interact with most of the TeamForge data objects, including users, projects, trackers and artifacts, code commits and tasks.

     ☞ **Note:** If you find a defect in a sample program or have a suggestion for improvement, please post a message in the *developer discussion forum*.
3. Set up the tools you will need:

   - A Java JDK, 1.8.0_45 or later, from *http://www.oracle.com/technetwork/java/index.html*.
   - Apache Ant, version 1.7.0 or later, from *http://ant.apache.org*.
4. Choose a place to host your plugin project online.

   We recommend *http://www.collab.net/community*, because you get access to all the collaboration support tools that CollabNet TeamForge provides.
5. Get the SDK from *http://www.collab.net/community/teamforge*.

   The SDK includes everything needed to develop and deploy your application:

   - Source files and compiled output.
   - Full JavaDoc reference material. Inside the package, look for `doc/com/collabnet/ce/soap60/ webservices`.
   - Annotated code examples.
6. Get help from other CollabNet TeamForge users and CollabNet staff in the TeamForge *API discussion forum*.

Example: Send commit data to TeamForge via SOAP

Use the Commit Tool provided with TeamForge to transmit your commit data to TeamForge.

The SCM Adapter allows you to develop integrations with almost any SCM tool, then exchange commit data with TeamForge. After you have created an SCM integration using the TeamForge API, you will use the Commit Tool provided with TeamForge to transmit your commit data to TeamForge.

For any SCM tool that supports triggers, you can use the tool's triggering mechanism to script the following actions. Otherwise, you can perform them manually for each commit.

☞ **Note:** The SCM Adapter option is always available, but will work only if you have developed an integration with an SCM tool using the TeamForge API.

1. Get the Commit Tool from `$SOURCEFORGE_HOME/integration/CommitTool.py`
2. Add the SCM server to your TeamForge installation.

   For the end user instructions, see *Integrate a source code server* on page 281 .
3. Create a repository on the SCM server.

   For the end user instructions, see *Create a source code repository* on page 79.
4. Use the Commit Tool to start a commit.

   `CommitTool.py create <systemId> <path> <username> <host> <port>`

   where:

   - `systemId` is the external system identifier that was given to the integration server. You can find this on the **Repository Details** page.
   - `path` is the path on the external system that was given to the repositorey. See the **Repository Details** page.
   - `username` is the TeamForge user name that will be performing the commit.
   - `host` is the TeamForge application server machine hostname.
   - `port` is the TeamForge application server machine SOAP port.
5. After the commit has been created, use the Commit Tool to add files with versions and actions.

   `CommitTool.py add <filename <version [<status [<fromfile <fromversion]]`

   where:

   - `filename` is the name of the file that is being placed into the commit.
   - `version` is the version of the file that is being placed into the commit.
   - `status` is the status of the file in the repository. Valid status values are ′added′, ′deleted′, ′modified′, ′moved′, and ′copied′. Status is optional and will default to ′added′ if no value is provided.
   - `fromfile` is the name of the original filename if this file was copied or moved. It is required on a copy or move, but is not allowed on an add, delete, or modify.
   - `fromversion` is the version of the original filename is this file was copied or moved. It is required on a copy or move, but is not allowed on an add, delete, or modify.
6. When all actions and files have been added to the commit, use the Commit Tool to transmit the commit to CollabNet TeamForge.

   `CommitTool.py commit <commitmessage>`

   where `<commitmessage>` is a description about the commit that will be displayed along with the files, versions, and operations.

   ☞ **Note:** To create an association, use the standard associate command [`<item id>`]. No special syntax is required when using the Commit Tool.

   For all of the above commands, when run correctly, there will be no output, and the return value of the program will be "0", indicating success. Failure will show a message and return with a non "0" return value.

   ☞ **Tip:** To see the status of the current commit, you can use the Commit Tool to print the output of the current commit:

   ```
   CommitTool.py print
   ```

   It will show output similar to this:

   ```
   Repository: <username>@<systemid>:<path> Modified filename 1.2
   Added document 1.1 Copied newfile 1.1 (From: origfile 1.3) <status>
   <filename> <version> (From: <fromfile> <fromversion>)
   ```

Update an application to the TeamForge 8.2 API

To use any of the new API calls introduced in TeamForge 8.2, you must update your existing applications to use the 8.2 API.

What's new in TeamForge 8.2

Click the links in the following table and refer to the Change History section to know the new and changed methods in TeamForge 8.2.

TeamForge service	Changed in 8.2?	Link to change history
CollabNet service	**Yes**	Javadoc for the *CollabNet* service
DiscussionApp service	No	Javadoc for the *DiscussionApp* service
DocumentApp service	**Yes**	Javadoc for the *DocumentApp* service
FileStorageApp and SimpleFileStorageApp services	No	Javadoc for the *FileStorageApp* and *SimpleFileStorageApp* services
FrsApp service	No	Javadoc for the *FrsApp* service
IntegrationDataApp service	No	Javadoc for the *IntegrationDataApp* service
NewsApp service	No	Javadoc for the *NewsApp* service
PageApp service	No	Javadoc for the *PageApp* service
PlanningApp service	**Yes**	Javadoc for the *PlanningApp* service
PluggableApp service	No	Javadoc for the *PluggableApp* service
RbacApp service	No	Javadoc for the *RbacApp* service
ScmApp service	**Yes**	Javadoc for the *ScmApp* service
TaskApp service	No	Javadoc for the *TaskApp* service
TrackerApp service	No	Javadoc for the *TrackerApp* service
WikiApp service	No	Javadoc for the *WikiApp* service
CategorizationApp service	No	Javadoc for the *CategorizationApp* service

☞ **Important:** The APIs for SourceForge Enterprise Edition 4.x, CollabNet SourceForge Enterprise 5.x, CollabNet TeamForge 5.x, and CollabNetTeamForge 6.x are fully compatible with CollabNet TeamForge 8.2. However, the calls from the different API versions are not interchangeable.

1. For calls that use the TeamForge-provided Java SDK classes, update all references to the package containing the classes.

For this version...	the package is located at...
5.x	`com.collabnet.ce.soap50`
6.x/7.x/8.x	`com.collabnet.ce.soap60`

2. For calls that access the WSDL directly, point your application to the new WSDL location.

For this version...	the WSDL is located at...
5.x	`http://<teamforge#server>/ce#soap50/services/<service#name>?wsdl`
6.x/7.x/8.x	`http://<teamforge#server>/ce#soap60/services/<service#name>?wsdl`

3. Update any calls that have been changed in the 8.2 API.

TeamForge SOAP API reference
TeamForge provides a SOAP service for each application.
TeamForge services available via SOAP
TeamForge provides a SOAP service for each tool in the application.

☞ **Note:** The WSDL for each TeamForge service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/<myService>?wsdl`. For an example, to review the WSDL for the Tracker service, see *http://ctf.open.collab.net/ce-soap60/services/TrackerApp?wsdl*. For information on the SOAP API, including the pluggable component package and classes, see the *SOAP package summary.*

The CollabNet service
The CollabNet service is the primary service that handles logging into and out of CollabNet TeamForge , finding and creating users, and retrieving lists of projects, project members, and project data.

The CollabNet service must always be accessed first, to authenticate the user and return a sessionId. The sessionId is a necessary parameter that must be passed to the methods in the other services.

The CollabNet service also handles retrieving and managing lists of users and projects, which are often the first steps that an application must perform.

Some examples of activities that are managed by the CollabNet service are:

* Returning a list of all CollabNet TeamForge users or projects.
* Creating a new user.
* Providing a user with a list of all projects of which he or she is a member.
* Creating an association between two items.
* Managing user groups and their membership.

User groups, added in CollabNet TeamForge Enterprise Edition 4.4 Service Pack 1, simplify permission management for projects. User groups are site#wide sets of users, managed by site administrators. Project administrators can add user groups to roles, just like they can add individual project members.

After authenticating the user through the CollabNet service, you can then access all of the other services that manage the items and activities associated with the other CollabNet TeamForge applications, such as trackers, tasks, documents, or file releases. A new "Anonymous Login" capability introduced in CollabNet TeamForge Enterprise Edition 4.4 adds the ability to log in with the privileges and access rights of a non#authenticated user.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the cemain service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/CollabNet?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/CollabNet?wsdl*.

New and changed methods

Refer to the Change History section on the *CollabNet service Javadoc* for a list of new and changed methods.

The discussion service

The DiscussionApp service handles the activities and items associated with forums.

Some examples of activities that are managed by the DiscussionApp service are:

• Creating a forum, forum topic, or forum post.
• Returning a list of all posts in a forum.
• Returning a list of forum posts created by a specified user.
• Returning a list of forum posts matching a search string.
• Deleting a forum, forum topic, or forum post.

After authenticating the user through the CollabNet service, you can then access the DiscussionApp service to work with forums, forum topics, and forum posts.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the DiscussionApp service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/DiscussionApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/DiscussionApp?wsdl*.

New and changed methods

Refer to the Change History section on the *DiscussionApp service Javadoc* for a list of new and changed methods.

The document service

The DocumentApp service handles the activities and items associated with documents and document folders.

Some examples of activities that are managed by the DocumentApp service are:

• Creating a document or document folder.
• Editing a document's details.
• Returning the name of the user who last edited or locked a document.
• Returning a list of documents matching a search string.
• Returning a list of all document folders in a project.
• Returning a list of all open document reviews in a project.
• Moving a document or document folder.

After authenticating the user through the CollabNet service, you can access the DocumentApp service to work with documents and document folders.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the DocumentApp service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/DocumentApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/DocumentApp?wsdl*.

New and changed methods

Refer to the Change History section on the *DocumentApp service Javadoc* for a list of new and changed methods.

The file release service

The FrsApp service handles the activities and items associated with file releases.

Some examples of activities that are managed by the FrsApp service are:

- Creating a package, release, or file.
- Editing a package, release, or file.
- Returning a list of all packages in a project.
- Returning a list of all releases in a package.
- Returning the status, maturity value, or other attribute of a release.

After authenticating the user through the CollabNet service, you can access the FrsApp service to work with file releases.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the file release service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/FrsApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/FrsApp?wsdl*.

New and changed methods

Refer to the Change History section on the *FrsApp service Javadoc* for a list of new and changed methods.

The file storage service

The FileStorageApp and SimpleFileStorageApp services handle uploading of files and simple data bytes.

Some examples of activities that are managed by the FileStorageApp and SimpleFileStorageApp services are:

- Uploading a file to CollabNet TeamForge .
- Downloading a file from CollabNet TeamForge .
- Checking the size of a file.

After authenticating the user through the CollabNet service, you can then access the FileStorageApp and SimpleFileStorageApp services to work with file uploads.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the file storage service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/FileStorageApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/FileStorageApp?wsdl*.

New and changed methods

Refer to the Change History section on the *FileStorageApp service Javadoc* and *SimpleFileStorageApp service Javadoc* for a list of new and changed methods.

The integration data service

The IntegrationDataApp service enables applications to associate their own data or metadata with CollabNet TeamForge objects.

☞ **Tip:** For information about installing integrated apps on a TeamForge site, see the *Integrated applications*.

An application can register its own namespace, to prevent collisions with other applications, and then store key#value pairs of data that are associated with any CollabNet TeamForge object.

- Developers who are creating integrations between CollabNet TeamForge and external applications can use this service to maintain associations between CollabNet TeamForge objects and counterparts in other systems.
- Developers creating customizations or enhancements can use this service to store additional data not directly supported by other CollabNet TeamForge APIs. For example, you can create extra, hidden fields on any object (such as an artifact, a user or a project) and use a Velocity customization to display that data where you need it.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *Javadoc for the integration data service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/ services/IntegrationDataApp?wsdl`. For example, here's the copy on the CollabNet web site: *http:// ctf.open.collab.net/ce-soap60/services/IntegrationDataApp?wsdl*.

New and changed methods

Refer to the Change History section on the *IntegrationDataApp service Javadoc* for a list of new and changed methods.

The news service

The NewsApp service handles the activities and items associated with news posts.

News posts for a project are visible on the project′s home page, and the application′s main page lists news posts for all projects visible to the current user.

Starting with CollabNet TeamForge , project news is presented on the project′s home page via a "Project News" page component, which can optionally be deleted or added to other pages. Each instance of the Project News component is a view of the same set of project news posts.

Some examples of activities that are managed by the NewsApp service are:

- Returning a list of all news posts in a project.
- Returning a list of all news posts matching a search string.
- Returning a list of all news posts in all projects of which a specified user is a member.
- Creating a news post.
- Deleting a news post.

After authenticating the user through the CollabNet service, you can access the NewsApp service to work with news posts.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the news app*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/ services/NewsApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/NewsApp?wsdl*.

New and changed methods

Refer to the Change History section on the *NewsApp service Javadoc* for a list of new and changed methods.

The planning folder service
The Planning Folder service enables you to create and maintain planning folders from external applications.

Planning folders are visible in the tracker tool of each project, as an organizing level sitting on top of trackers. Planning folders are deisgned to support agile management practices.

Some examples of activities managed by the PlanningApp service are:

- Creating and deleting planning folders.
- Reordering planning folders.
- Setting the statuses of planning folders.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the planning folder app*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/PlanningApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/PlanningApp?wsdl*.

New and changed methods

Refer to the Change History section on the *PlanningApp service Javadoc* for a list of new and changed methods.

The project pages service
The PageApp service handles the activities and items associated with project pages.

The project pages feature enables project owners to customize their project home by creating their own custom pages. Multiple pages can be added, and these pages can be assembled into a hierarchy, browseable via a "tree" navigation on the left side of the project′s home. The content of each page can be customized by adding components of various types, such as Text, News, and Document Folder.

Some examples of activities that are managed by the PageApp service are:

- Creating pages and components.
- Managing the page hierarchy.
- Managing the order of components on a page.
- Updating text components with new HTML content.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the project pages service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/PageApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/PageApp?wsdl*.

New and changed methods

Refer to the Change History section on the *PageApp service Javadoc* for a list of new and changed methods.

The project categorization service

The CategorizationApp service handles the activities and items associated with creation and management of Project Categorization.

Some examples of activities that are managed by the CategorizationApp service are:

- Creating categories
- Adding categories to projects
- Retrieving category and project data.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the CategorizationApp service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/CategorizationApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/CategorizationApp?wsdl*.

New and changed methods

Refer to the Change History section on the *CategorizationApp service Javadoc* for a list of new and changed methods.

The role-based access control service

The RbacApp service handles the activities and items associated with role creation and management.

Some examples of activities that are managed by the RbacApp service are:

- Creating a role.
- Adding a user to a role.
- Adding permissions to a role.
- Adding a user group to a role.

Permissions are managed in terms of "clusters." Permission clusters correspond to the sets of permissions that are managed in the role administration section of the application's user interface, such as "view" or "create/edit/view." Clusters can be associated with top#level folders in the application (e.g. with trackers in the Tracker application, or packages in the File Releases application). The RbacApp service allows this same permission management to be performed via SOAP.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the role-based access control service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/RbacApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/RbacApp?wsdl*.

New and changed methods

Refer to the Change History section on the *RbacApp service Javadoc* for a list of new and changed methods.

The software configuration management (SCM) service

The ScmApp service handles the activities and items associated with integrated software configuration management (SCM) applications.

Some examples of activities that are managed by the ScmApp service are:

- Returning a list of files associated with a code commit.
- Returning a list of all repositories in a project.
- Returning a list of all code commits in a repository.
- Editing commit information.

After authenticating the user through the CollabNet service, you can access the ScmApp service to work with integrated SCM applications.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the software configuration management service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/ScmApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/ScmApp?wsdl*.

SCM Adapter

The SCM Adapter allows you to develop integrations with almost any SCM tool, then exchange commit data with CollabNet TeamForge . An SCM Adapter option is added to the Type menu on the CollabNet TeamForge Create Integration page. Use this option to add your SCM integration server to CollabNet TeamForge .

☞ **Note:** The SCM Adapter option is always available, but will work only if you have developed an integration with an SCM tool using the CollabNet TeamForge API.

After you have created an SCM integration using the CollabNet TeamForge API, you will use the Commit Tool provided with CollabNet TeamForge to transmit your commit data to CollabNet TeamForge .

New and changed methods

Refer to the Change History section on the *ScmApp service Javadoc* for a list of new and changed methods.

The tasks service
The TaskApp service handles the activities and items associated with tasks and task folders.

Some examples of activities that are managed by the Task App service are:

- Returning a list of all tasks assigned to a specified user.
- Returning a list of all task folders in a project.
- Returning a list of all tasks matching a search string.
- Listing task dependencies.
- Creating a task or task folder.
- Editing a task.
- Moving tasks and task folders.

After authenticating the user through the CollabNet service, you can access the TaskApp service to work with tasks and task folders.

☞ **Note:** Task folders are referred to as task groups in the method descriptions.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the tasks service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/TaskApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/TaskApp?wsdl*.

New and changed methods

Refer to the Change History section on the *TaskApp service Javadoc* for a list of new and changed methods.

The tracker service

The TrackerApp service handles the activities and items associated with trackers and tracker artifacts.

Some examples of activities that are managed by the TrackerApp service are:

- Returning a list of all trackers in a project.
- Returning a list of all tracker artifacts matching a search string.
- Returning a list of all tracker artifacts assigned to a specified user.
- Creating a tracker artifact.
- Editing a tracker artifact.
- Moving a tracker artifact.

After authenticating the user through the CollabNet service, you can access the TrackerApp service to work with trackers and tracker artifacts.

Detailed information

For a complete description of each method in the TrackerApp service, including the SOAP data objects (DO), data lists, and data rows, see the *JavaDoc for the tracker service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/TrackerApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/TrackerApp?wsdl*.

New and changed methods

Refer to the Change History section on the *TrackerApp service Javadoc* for a list of new and changed methods.

The wiki service

The WikiApp service handles activities associated with project wiki pages.

Some examples of activities that are managed by the WikiApp service are:

- Creating a Wiki page.
- Adding Wiki content.
- Retrieving Wiki content in HTML format.

Detailed information

For a complete description of each method, including its parameter definitions and SOAP faults, see the *JavaDoc for the wiki service*.

WSDL

The WSDL for this service is available on your TeamForge site at `https://<mysite.com>/ce-soap60/services/WikiApp?wsdl`. For example, here's the copy on the CollabNet web site: *http://ctf.open.collab.net/ce-soap60/services/WikiApp?wsdl*.

New and changed methods

Refer to the Change History section on the *WikiApp service Javadoc* for a list of new and changed methods.

Context-specific objects for manipulating Velocity pages

The rendering context is a set of objects made available to the Velocity template. You can access the members of these objects in a template using the standard Velocity syntax.

ArrayTool

This is necessary because Velocity does not provide accessors into arrays and some of the API calls will return SoapNamedValues objects that have two matching arrays of name and value.

length(Object[] array)

Get the length of the array.

array

> The array being examined.

Returns: The length of the array (null array will return 0).

get(Object[] array, int index)

Get the object at the specified index of the array.

array

> The array that contains the object of interest.

index

> The index we are pulling the object from.

Returns: The object stored at that array of null if the array is empty or position is invalid.

FORM [FormTool]

Manages opening and closing of forms.

startForm(String action, String formId)

Get the opening tag for a form.

action

> The path to the action that is handling the form (e.g. /cemain/do/ login).

formId

> The id/name of the form (they will be the same).

Returns: The open tag for the form and any standard hidden elements.

startForm(String action, String formId, String formName)

Get the opening tag for a form.

action

> The path to the action that is handling the form (e.g. /cemain/do/ login).

formId

> The id of the form.

formName

> The name of the form.

Returns: The open tag for the form and any standard hidden elements.

endForm()

Close out a form.

Returns: The form closing tag.

LINK [LinkTool]
Creates links; see default templates for examples.

getPageUrl(String urlBase, String action)

Get the url for a page being linked to.

urlBase

> The base url.

action

> The action being linked to.

Returns: The value to place in the href attribute of a link.

getUserUrl(String username, String fullName)

Generates URL for user details.

username

> User name.

fullName

> User's full name.

Returns: URL to user details page.

MESSAGE [MessageTool]
Enables internationalization of templates, providing functions for localizing strings.

get(String bundle, String key[, String arg0[, String arg1[, String arg2]]])

Returns the message.

bundle

> Resource bundle.

key

> Message key.

arg0

> Message argument.

arg1

> Message argument.

arg2

> Message argument.

Returns: Message.

getFieldLabel(String bundle, String key, boolean required)

Returns the message with the appropriate label modifier (':' and possibly an asterisk if field is required).

bundle

> The bundle where the message lives.

key

> The key where the message is stored.

required

> If the field is required.

Returns: The message with appropriate label modifiers.

STRING [StringTool]
Returns a formatted file size string, such as 45KB, 100MB, or 1GB

formatFileSize(long fileSize)
fileSize

> File size in bytes.

Returns: File size string.

wordWrap(String text, int width)

Wraps text string at word boundaries.

s

> String to word wrap.

width

> Number of characters at which to wrap.

Returns: Word wrapped string.

escapeXml(String s)

Escape out the xml from the text

s

> The text that we need to escape xml from

Returns: The text with xml escaped

TEXTPARSER [TextParserTool]
Converts text into "linkified" html, with IDs and wiki words converted into appropriate links.

parseText(String text)

Return linkified text (add href tags to object ids and urls).

text

> Text to process.

Returns: Replacement text with link replacement.

Example integration script
This integration script demonstrates how to use the SCM Adapter to create an integration with Subversion.

Overview

This is a post#commit script that gets executed either from or as the hooks or post#commit trigger.

> ☞ **Note:** This example will work for Subversion only. It will not work in its present form for any other SCM tool. It is intended as an example that you can reference when creating scripts for other SCM tools. The

example also contains error checking as an example only; post#commit triggers fail to abort a commit on Subversion.

```sh
#!/bin/sh
# Location of SOURCEFORGE_HOME
SOURCEFORGE_HOME="/opt/collabnet/teamforge/sourceforge_home"
# The Hostname of the application server
SOURCEFORGE_SOAP_HOST="ctf.open.collab.net"
# The port used for soap communication
SOURCEFORGE_SOAP_PORT="8080"
# The "Repository Path" of the repository to commit to.
SOURCEFORGE_REPOSITORY="myrepository"
# The external system id of the integration server.
SOURCEFORGE_SYSTEMID="exsy1001"
# Path to the svnlook command
SVNLOOK=/usr/bin/svnlook
# Path to the CommitTool.py command
COMMITTOOL="$SOURCEFORGE_HOME/integration/CommitTool.py"
# If the author metadata of the commit is empty, use this user. This
will normally
# happen if a commit occurs that bypasses authentication
DEFAULTUSER="admin"
# Repository storage path. This is passed in by Subversion
REPOS="$1"
# Transaction id of this commit.  This is passed in by Subversion
TXN="$2"
# Execute an svnlook log to get the Commit Message of this commit
LOGMSG=`"$SVNLOOK" log -r "$TXN" "$REPOS"`
# Execute the svnlook command to get the author of the commit if
any.
AUTHOR=`"$SVNLOOK" author -r "$TXN" "$REPOS"`
if [ "X$AUTHOR" == "X" ]; then
AUTHOR="$DEFAULTUSER"
fi
$SVNLOOK changed -r "$TXN" "$REPOS" | (
# Create a CommitTool transaction.
"$COMMITTOOL" create "$SOURCEFORGE_SYSTEMID"
"$SOURCEFORGE_REPOSITORY" "$AUTHOR" \
"$SOURCEFORGE_SOAP_HOST" "$SOURCEFORGE_SOAP_PORT"
if [ $? -ne 0 ]; then
exit 1;
fi
read TYPE FILE
# Iterate through each line of the svnlook changes command and add a
file with
# the appropriate status to the CommitTool.py transaction.
while [ "X$TYPE" != "X" ]; do

# default to Added
COMMAND="Added"
case "$TYPE" in
U)
COMMAND="Modified"
;;
D)
COMMAND="Deleted"
;;
A)
COMMAND="Added"
;;

esac
```

```
        "$COMMITTOOL" add "$FILE" "$TXN" "$COMMAND" >> /dev/null
        if [ $? -ne 0 ]; then
        exit 1;
        fi
        read TYPE FILE
        done
        # Submit the current transaction to sourceforge with the commit
  message.
        "$COMMITTOOL" commit "$LOGMSG" >> /dev/null

        if [ $? -ne 0 ]; then
        exit 1;
        fi
        )
        if [ $? -ne 0 ]; then
        exit 1;
        fi
        exit 0
```

Context-specific object members

You can access the members of these objects in a template using the standard Velocity syntax.

For example:

```
${PAGE_INFO.actionName()}
$(PAGE_INFO.isSuperUser()}
${GLOBAL.datePattern()}
${API44.discussionApp()}
```

etc.

PAGE_INFO [PageInformationTool]

Allows access to information specific to the page that is passed in.

Information	Description
actionName	The name of the action that was requested (e.g. viewArtifact, listTrackers)
currentUser	User object that contains information about the current user: • $PAGE_INFO.currentUser.username (Login ID) • $PAGE_INFO.currentUser.fullName (Full name) • $PAGE_INFO.currentUser.email (E#mail address) • $PAGE_INFO.currentUser.locale (Locale, such as en_US) • $PAGE_INFO.currentUser.lastLogin (Date and time of last login)
path	The full path string of the request's target object (e.g. projects.test/trackers.foo/bar). If no path is present on the request, this returns an empty string.
projectPath	The project path (e.g. projects.test) on the current folderPath or an empty

Information	Description	
		string if none exists (no path context or just a project context).
itemName	The name of the item (e.g. artf1234, Home) on the current request or an empty string if there is no path context or just a project or folder context.	
objectId	The id of the requested object (e.g. artf1234, proj1234).	
objectType	The type string of the current object (i.e. An artifact would have Tracker.Artifact	
projectId	The ID of the project on the request or the project that contains the current folder or item.	
requestUrl	The URL that was requested (e.g. /ce/tracker/do/viewArtifact/ projects.test/tracker.foo/artf1234).	
isSuperUser	True if the current user is a site admin.	
isLoggedIn	True if the current user is logged into the system.	

GLOBAL [GlobalTool]
Allows access to various kinds of static information about the site.

Static Information	Description
datePattern	The standard date pattern used for date conversions.
imageRoot	The root URL path of CollabNet TeamForge images.
isApprovalRequiredForNewUsers	A flag that indicates whether new users must be approved before they can use the site.
isRequireAssociationOnDocumentCreate	A flag that indicates whether creation of a document requires an association with an object.
isSelfCreationEnabled	A flag that indicates whether visitors are allowed to create new accounts in the application by themselves.
isUsingExternalAuthentication	A flag that indicates whether logins are authenticated against an external server. (Currently this means LDAP.)

ApiTool
API60 [ApiTool] exposes the CollabNet TeamForge SOAP service interfaces for use by Velocity templates.

While the interfaces corresponding to the SOAP API are provided, the implementation is efficient; the SOAP network protocol is not actually used.

☞ **Note:** API50 is present for backward compatibility and may be removed in future. API44 and API43 are removed completely.

Interfaces

Interface	Description
sessionKey	Gets the soap session id that should be passed in as the first parameter to soap service methods.
discussionApp	Gives access to the methods in the Discussion application.
documentApp	Gives access to the methods in the Documents application.
frsApp	Gives access to the methods in the File Releases application.
integrationDataApp	Gives access to the methods for Data Integration.
newsApp	Gives access to the methods in the Project News application.
rbacApp	Gives access to the methods for role#based access control.
scmApp	Gives access to the methods for version control (SCM) integration.
SourceForge	Gives access to the methods in the main CollabNet TeamForge application.
taskApp	Gives access to the methods in the Tasks application.
trackerApp	Gives access to the methods in the Tracker application.
wikiApp	Gives access to the methods in the Wiki application.
categorizationApp	Gives access to the methods in project categorization.
emptyFilter	Gets an empty filter that can be passed into methods that require soap filters.
PageApp	Handles the activities and items associated with project pages.
PlanningApp	Gives access to methods in the Planning application.

Example

This is an example of how you might use some popular API calls in a Velocity template:

```
## Display Server version, current project title and description
#set( $sessionKey = $API60.sessionKey )
#set( $projectId = $PAGE_INFO.projectId )
#set( $projectData = $API60.sourceForge.getProjectData($sessionKey,
$projectId) )

<p>Server Version: $API60.sourceForge.getVersion($sessionKey)</p>
<p>Project Title: ${projectData.title}</p>
<p>Project Desc: ${projectData.description}</p>
```

Lab Management API methods

Almost all Lab Management functionality is available via an API method.

Lab Management API methods come in two forms: signed and unsigned.

- Signed methods require authentication, in accordance with *the User Authentication documentation*. Most of the API methods in Lab Management, and all of the most useful ones, are *signed* methods which require authentication to use. The `cubit_api_client.py` client makes using these signed methods easy by handling the authentication negotiation for you.
- Unsigned methods do not require authentication.

☞ **Note:** This API is under active development. We have started with what we feel are the most common actions which would need to be performed on an automated basis, and will be expanding this list over time as we receive feedback about what our users want and need. Please let us know if there are missing features or bugs!

CheckHostConsistency
Check the setup of all hosts in the domain to make sure their representation is internally complete within Lab Management.

Overview

This method is only available to users to Lab Management Domain Admins and can be considered a maintenance method that should rarely, if ever, need to be run. Three areas of setup consistency are checked for each host:

1. Verification and re-application of the host's internal database representation within Lab Management. This may be needed after an upgrade of the Lab Management application is installed.
2. Verification that the IP address assigned to the host matches the hostname. If you suspect DNS setup issues in your domain, this test should catch them (unless the errors are transient). Lab Management will not let you create hosts with invalid DNS, but if DNS for the host gets changed improperly after the host is created, your host will experience many problems in the Lab Management environment until it is fixed.
3. Verification of the IP mask/gateway configured for the host with the actual one it resolves to.
4. Re-generation of statistics system performance monitoring configuration files for the host. If monitoring is not properly working for a host, regenerating the configuration files will usually fix the problem.

Depending on the number of hosts in your domain, this method can take some time to complete. If you have questions on the output of this method, please see your local Lab Management Administrator or contact CollabNet directly for assistance.

URL

`/cubit_api/1/check_host_consistency`

Authentication

This method requires *authentication* using an API key.

Parameters

`sig` **(Required, once)**	API authentication hash signature.
	Type: String
`userid` **(Required, once)**	The login name of the user initiating the request.
	Type: String

Example Response

Successful completion of host consistency check:

```xml
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
    <output>All Hosts OK</output>
```

```
    </cubit>

    If the user is not a Lab Management Domain Admin:
    <?xml version='1.0'?>
    <cubit version='1'>
        <error>User 'grue' is not authorized to run this method.</error>
    </cubit>

    If there is an error re-generating the host's internal database
    representation for at least one host:
    <?xml version='1.0'?>
    <cubit version='1'>
        <status>OK</status>
            <output>Hosts which were not successfully updated:
            cu011.dev.cubitdemo.net</output>
    </cubit>

    If the system performance statistics configuration files are not
    properly generated for at least one host:
    <?xml version='1.0'?>
    <cubit version='1'>
        <status>OK</status>
            <output>Hosts which are not properly setup to collect
            performance data: cu011.dev.cubitdemo.net</output>
    </cubit>

    If there is a DNS mismatch for at least one host:
    <?xml version='1.0'?>
    <cubit version='1'>
        <status>OK</status>
            <output>Hosts which require re-ip'ing:
            cu011.dev.cubitdemo.net</output>
    </cubit>

    If there is a IP mask mis-match:
    <?xml version='1.0'?>
    <cubit version='1'>
        <status>OK</status>
            <output>Hosts whose IP mask/gateway does not match with the
            IP mask/gateway it resolves to:
            cu011.dev.cubitdemo.net</output>
    </cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

CloudCreateHosts
Create one or more hosts from a cloud given the host-type and size.

The cloud will match the host-type and size to one (or more) source(s) from the cloud in order to fill the entire order. Host types and sizes cannot be mixed in a single API call. All the hosts must be the same type/size. The method will fail if the cloud's sources cannot accomodate that many hosts of the given type/size.

Any user who is a member of a project that has been given permission to allocate from this cloud can create instances.

URL

`/cubit_api/1/allocate_hosts_from_cloud`

Authentication

This method requires *authentication* using an API key.

Parameters

alloc_hours (zero or once)

Amount of time for which to allocate the host. The allocation limit is subject to the limit set by the project or cloud administrator.

Type: Float

alloc_minutes (zero or once)

Similar to alloc_hours parameter, but in minutes. It is mutually exclusive with the alloc_hours option, neither of these options should be specified, or only one of these options should be specified. If alloc_hours or alloc_minutes is set to 0, or if alloc_hours and alloc_minutes is unset, the allocation time defaults to the longest possible time available in the project.

Type: Integer

cloud (Required, once)

Name of Lab Management cloud to allocate instances from.

Type: String

count (zero or once)

The number of identical instances to bring up. The instances will all be running the same profile and version. Must be an integer greater than 0. If unset, defaults to 1.

Type: Integer

descr (zero or once)

Set the specified string as a description for the allocated hosts. This is useful if you wish to uniquely identify a group of hosts which have all been allocated to the same user at the same time for the same purpose.

Type: String

dont_delete (zero or once)

By default, EC2 instances will automatically be deleted when they are deallocated. If dont_delete is set to True, the instances created will not be deleted when they are deallocated, and will instead revert back to the project pool.

Type: String

host_type (Required, once)

The name of the host_type of machine to create. Must be one of the host-types supported by the cloud.

Type: String

profile (Required, once)

Name of the profile to assign to the host. The profile must, of course, be a profile that is eligible to be built by the host-type/size you selected.

Type: String

project (Required, once)

Name of Lab Management project to place hosts into. You must have permission to add hosts to this project: that is, you must either be a Project Admin or Delegated Host Management must be turned on in your project.

Type: String

revision (zero or once)	Revision number of the profile you wish to assign to the host. Mutually exclusive with the `version` option, but at least one of the `version` or `revision` options must be specified.
	Type: Integer
sig (Required, once)	API authentication hash signature.
	Type: String
size (Required, once)	The name of the machine size to create. Must be one of the sizes supported by the given host_type.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String
version (zero or once)	Version number or tag name of the profile you wish to assign to the host. The special version tag `HEAD` always denotes the latest version of the profile at the moment of execution. Note that profile tags can move between versions: this is a useful feature, but you should be aware of it. Mutually exclusive with the `revision` option, but at least one of the `version` or `revision` options must be specified.
	Type: String

Example Response

```
Successful allocation of 10 instances:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
    <output>ec2-pending-1204832590-85</output>
    <output>ec2-pending-1204832595-09</output>
    <output>ec2-pending-1204832599-97</output>
    <output>ec2-pending-1204832611-49</output>
    <output>ec2-pending-1204832615-60</output>
    <output>ec2-pending-1204832620-29</output>
    <output>ec2-pending-1204832624-25</output>
    <output>ec2-pending-1204832628-46</output>
    <output>ec2-pending-1204832640-90</output>
    <output>ec2-pending-1204832645-60</output>
</cubit>

<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user lacks permission to create hosts in the project:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to create hosts in this project.</error>
</cubit>

If the project does not have the permissions to allocate from the
requested cloud:
<?xml version='1.0'?>
```

```
    <cubit version='1'>
        <error>This project is not authorized to create hosts in this cloud.</
error>
    </cubit>

    If the cloud does not have enough space on the sources to allocate that
 many hosts:
    <?xml version='1.0'?>
    <cubit version='1'>
        <error>This cloud does not have enough resources to create [n]
 hosts.</error>
    </cubit>
```

Response Codes

- `400` - Login failed / Insufficient permissions

GetAuthToken
GetAuthToken(userid) -> token

URL

`/cubit_api/1/get_auth_token`

Authentication

This method does not require authentication.

Parameters

This method does not take any parameters.

Response Codes

This method does not have any documented response codes.

HostAllocate
Allocate a host to a user.

Hosts must be in the `Free` state to be allocated, although users with Project Admin access can re-allocate hosts which are in the `Allocated` state to other users in their project. Domain Admin users can re-allocate machines between projects and users, although if a machine is reallocated to a user in another project, that user must also be a member of the destination project. The user must be authorized to allocate the host. If the host is in any of the following states: `Immutable`, `Powercycle`, `Rebuild`, or `Rebuilding`, this method will fail.

URL

`/cubit_api/1/alloc_host`

Authentication

This method requires *authentication* using an API key.

Parameters

| **alloc_hours (zero or once)** | Amount of time for which to allocate the host. The allocation limit is subject to the limit set by the project or cloud administrator. |
| | Type: Float |

alloc_minutes (zero or once)	Similar to alloc_hours parameter, but in minutes. It is mutually exclusive with the `alloc_hours` option, neither of these options should be specified, or only one of these options should be specified. If `alloc_hours` or `alloc_minutes` is set to 0, or if `alloc_hours` and `alloc_minutes` is unset, the allocation time defaults to the longest possible time available in the project. Type: Integer
alloc_proj (zero or once)	The project to allocate the machine to. The `alloc_user` must be a valid Lab Management user in the project, or else this method will fail. Only Domain Admins can change the project that a host is assigned to. If not specified, the project is not changed. At least one of `alloc_user` or `alloc_proj` must be specified. Type: String
alloc_user (zero or once)	Login name of user to allocate host to. Only Project Admins or better can specify a user other than themselves. Other users can only allocate machines to themselves. If not specified, defaults to `userid`. At least one of `alloc_user` or `alloc_proj` must be specified. Type: String
force (zero or once)	If the machine is currently in the `Allocated` state, the `force` option must be given to reassign the host to another user. The `force` option is only available to Project Admins and above. If the host is a virtual host, and any virtual guests of this host are in the `Allocated` state, this option must also be used, or the entire allocation will fail. The only valid value for this parameter is `True`. Type: String
guests (zero or once)	If the machine is a virtual host and has active virtual guests, setting this parameter to `True` will move the virtual guests along with the virtual host. Because moving a virtual host will move several hosts at once, we have a separate parameter to confirm this action. The only valid value for this parameter is `True`. If the host has no virtual guests, this option has no effect. Type: String
host (Required, once)	Fully qualified hostname to allocate. Type: String
sig (Required, once)	API authentication hash signature. Type: String
userid (Required, once)	The login name of the user initiating the request. Type: String

Example Response

```
Successful host allocation:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to allocate this host.</error>
</cubit>

If the host is allocated to a deleted project:
<?xml version='1.0'?>
<cubit version='1'>
    <error>Cannot allocate host 'cu001.dev.cubitdemo.net' to a
    deleted project 'look'</error>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

HostAssignProfileAndRebuild

Assign a profile to the host and rebuild it with that profile/version.

The user must be authorized to rebuild the host. If the host is in any of the states which disallow rebuilding, such as `Immutable`, `Rebuild`, or `Rebuilding`, this method will fail. The host must also not be of a type which prohibits rebuilding (e.g., an EC2 host). If you do not want to change the profile a host is running, and just want to rebuild a host, you can save a few keystrokes and use the Rebuild method instead.

URL

`/cubit_api/1/assign_rebuild`

Authentication

This method requires *authentication* using an API key.

Parameters

`host` (Required, once)	Fully qualified hostname to assign profile to and rebuild.
	Type: String
`profile` (Required, once)	Name of the profile to assign to the host.
	Type: String
`revision` (zero or once)	Revision number of the profile you wish to assign to the host. Mutually exclusive with the `version` option, but at least one of the `version` or `revision` options must be specified.
	Type: String
`sig` (Required, once)	API authentication hash signature.
	Type: String
`userid` (Required, once)	The login name of the user initiating the request.
	Type: String

version (zero or once)

Version number or tag name of the profile you wish to assign to the host. The special version tag HEAD always denotes the latest version of the profile at the moment of execution. Note that profile tags can move between versions: this is a useful feature, but you should be aware of it. Mutually exclusive with the revision option, but at least one of the version or revision options must be specified.

Type: String

Example Response

```
Successful initiation of rebuild:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to allocate a profile to this host and
rebuild it.</error>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostDelete

Delete one or more hosts.

The host must not be in the Immutable state for this method to work. Hosts can be deleted by users under the following conditions:

- Lab Management Domain Admins can delete any host, regardless if it is physical, virtual, or from a remote cloud
- Users who are *not* Lab Management Domain Admins can delete *virtual guests* if the parent host is allocated to them and Delegated Host Management is turned on in their project.
- Users who are *not* Lab Management Domain Admins can also delete *remote cloud* hosts allocated to them if Delegated Host Management is turned on in their project.

URL

/cubit_api/1/delete_host

Authentication

This method requires *authentication* using an API key.

Parameters

force (zero or once)

If the machine being deleted does not belong to the user initiating the request, the force option is required. The only valid value for this parameter is True.

Type: String

guests (zero or once)	If the machine is a virtual host and has active virtual guests, setting this parameter to `True` will delete the virtual guests along with the virtual host. Because deleting a virtual host will delete several hosts at once, we have a separate parameter to confirm this action. The only valid value for this parameter is `True`. If the host has no virtual guests, this option has no effect. Use of this option can be really convenient, or it can really ruin someone's day!
	Type: String
hosts (Required, once)	List of comma-separated, fully qualified hostnames to delete. Hosts must not be in the `Immutable` state.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
Successful host delete of the host cu011.cubit.example.com:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
    <output>cu011.cubit.example.com</output>
</cubit>

If the user is unauthorized:
<?xml version='1.0'?>
<cubit version='1'>
    <error>0 of 1 hosts deleted successfully. Failed hosts below.</error>
    <output>cu013.cubit.example.com: You are not authorized to delete this
host.</output>
</cubit>

If a deletion of hosts partially succeeds:
<?xml version='1.0'?>
<cubit version='1'>
    <error>6 of 8 hosts deleted successfully. Failed hosts below.</error>
    <output>cu013.cubit.example.com: You are not authorized to delete this
host</output>
    <output>cu016.cubit.example.com: Host deletion failed</output>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

HostFree
Free a host currently assigned to a user.

Hosts must be in the `Allocated` state to be freed. A user can only free hosts already assigned to them, although Project Admins and Domain Admins can free hosts in their project which are currently allocated. If it is a virtual guest and "Delete on Deallocation" flag is ON, the virtual guest would be deleted after it is set Free.

URL

```
/cubit_api/1/free_host
```

Authentication

This method requires *authentication* using an API key.

Parameters

force (zero or once)	If the machine being freed does not belong to the user initiating the request, the `force` option is required. Only Project Admins and Domain Admins can free hosts which do not belong to them. The only valid value for this parameter is `True`.
	Type: String
host (Required, once)	Fully qualified hostname to free. Hosts must be in the `Allocated` state to be freed.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
Successful freeing of host:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to free this host.</error>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

HostGetAssignedProfile
Get the assigned profile for the host, and returns the profile name along with the revision and version number.

Output is available in both text and XML formats. You can also get this information from the *QueryAlloc* on page 895 method, but the output of this method is easier to parse if you have scripts checking a profile. The user requesting information about a host must have at least login privileges to that host.

URL

```
/cubit_api/1/assigned_profile
```

Authentication

This method requires *authentication* using an API key.

Parameters

host (Required, once)	Fully qualified hostname to query.
	Type: String
output (Required, once)	Output type, either 'txt' (for text output) or 'xml' (for XML output).
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
Sample XML output:
<?xml version='1.0'?>
<cubit version='1'>
  <profile revision="1838" version="5">solaris10_x86</profile>
</cubit>

Sample text output:
solaris10_x86  1838  5

If user is unauthorized:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You do not have permissions to view this host</error>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostGetMyAssignedProfile
Get the profile and version assigned to the host invoking this method.

URL

/cubit_api/1/my_profile

Authentication

This method does not require authentication.

Parameters

output (Required, once)	Output type, either 'txt' (for text output) or 'xml' (for XML output).
	Type: String

Example Response

```
Sample XML output:
<?xml version='1.0'?>
<cubit version='1'>
  <profile revision="26" version="2">rhel3_base</profile>
</cubit>

Sample text output:
rhel3_base  26  2
```

Response Codes

- `200` - ok

HostPerfColl
Get performance collection data from remote hosts managed by Lab Management.

This method takes one parameter, `data`. The hostname is auto-derived using reverse DNS. There is no other way to verify the sending host.

The currently supported statgroups are:

- `net`: Returns a dictionary of network interface transfer metrics. The key is the name of the interface. Value is a tuple containing: (*bytes_in, bytes_out, errors_in, errors_out, pkts_in, pkts_out*)
- `loadavg`: Returns a tuple containing the (1, 5, 15) minute load average on the system.
- `storage`: Returns a dictionary of disk storage metrics. The key is the name of the mounted partition and the value is a tuple of: (`block_size, blocks_avail, blocks_used, block_errors`)
- `systatd`: Returns a string containing the systatd output file for the system.

URL

`/cubit_api/1/host_perfcoll`

Authentication

This method does not require authentication.

Parameters

This method does not take any parameters.

Response Codes

This method does not have any documented response codes.

HostPowercycle
Powercycle a host.

This method immediately executes a powercycle of the host. No graceful shutdown is run on the host at the operating system level. It is recommended that this method only be run when the normal operating shutdown/reboot sequence has failed or is unavailable. You can only powercycle machines in the `Allocated`, `Free`, and `Immutable` states.

URL

`/cubit_api/1/powercycle`

Authentication

This method requires *authentication* using an API key.

Parameters

force (zero or once)
If the machine being powercycled is a virtual host, the force option is required. The only valid value for this parameter is True.
Type: String

host (Required, once)
Fully qualified hostname to powercycle.
Type: String

sig (Required, once)
API authentication hash signature.
Type: String

userid (Required, once)
The login name of the user initiating the request.
Type: String

Example Response

```
Successful completion of powercycle:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to powercycle this host.</error>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostRebuildCancel
Cancel a rebuild for a host.

The host must be in Rebuild state for this method to work. In order to cancel a rebuild for a host, the user must have permission to rebuild the host, that is, they must either:

- Have the host allocated to them
- Be a Lab Management Project Admin in the project, or
- Be a Lab Management Domain Admin

URL

/cubit_api/1/rebuild_cancel

Authentication

This method requires *authentication* using an API key.

Parameters

host (Required, once)	Fully qualified hostname to cancel rebuild of. Hosts must be in the Rebuild state.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
Successful host rebuild cancel:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to cancel a rebuild for this host.</
error>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

HostRebuild

Rebuild a host with the currently selected profile and profile version.

The user must be authorized to rebuild the host. The host must also not be in any of the states which prohibit rebuilding, such as Immutable, Powercycle, Rebuild, or Rebuilding, or this method will fail.

The host must also indicate that it is rebuildable (e.g., not an EC2 host), or this method will fail.

URL

/cubit_api/1/rebuild

Authentication

This method requires *authentication* using an API key.

Parameters

host (Required, once)	Fully qualified hostname to rebuild.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
Successful initiation of rebuild:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>

If the user is unauthorized, or the host does not exist:
<?xml version='1.0'?>
<cubit version='1'>
    <error>You are not authorized to rebuild this host.</error>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

PblChangeDesc
Project Build Library (PBL) file and directory description changing interface. This function is used to change the description for files and directories in the Project Build Library. This function may be useful if you are building your own PBL client.

URL

`/cubit_api/1/pbl_changedesc`

Authentication

This method requires *authentication* using an API key.

Parameters

comment (zero or once)	An optional comment to leave about the operation being performed. The comment will not appear in the PBL, but it will be in the audit log entry for this event.
	Type: String
desc (Required, once)	The new text description of the file. This description will completely replace any description currently set for the file.
	Type: String
path (Required, once)	The path to the file being operated on. For example, if the complete file URL is `/pbl/zork/pub/foo/bar/test.txt`, the `path` would be `/foo/bar/test.txt`
	Type: String
proj (Required, once)	The name of the project which contains the file we are operating on.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String

type (Required, once)	The type of file that we are operating on. Valid values are 'pub' and 'priv'.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

PblDelete

Project Build Library (PBL) file and directory delete interface. This function is used to permanently remove files and directories in the Project Build Library. This function may be useful if you are building your own PBL client.

URL

`/cubit_api/1/pbl_delete`

Authentication

This method requires *authentication* using an API key.

Parameters

comment (zero or once)	An optional comment to leave about the operation being performed. This will not appear in the PBL, but it will be in the audit log entry for this event.
	Type: String
dryrun (Required, once)	If this parameter is set to True, the specified path will not actually be deleted. The only valid value for this parameter is True.
	Type: String
force (zero or once)	If the force option is present and set to True, and the specified path argument is a directory, a recursive delete of the directory will be performed. The only valid value for this parameter is True.
	Type: String
path (Required, once)	The path to the file being operated on. For example, if the complete file URL is /pbl/zork/pub/foo/bar/test.txt, the path would be /foo/bar/test.txt
	Type: String

proj (Required, once)	The name of the project which contains the file we are operating on.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
type (Required, once)	The type of file that we are operating on. Valid values are 'pub' and 'priv'.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>
```

Response Codes

- 200 - ok
- 400 - Login failed / Insufficient permissions

PblMove
Project Build Library (PBL) file and directory move interface. This function may be useful if you are building your own PBL client.

URL

/cubit_api/1/pbl_move

Authentication

This method requires *authentication* using an API key.

Parameters

comment (zero or once)	An optional comment to leave about the operation being performed. This will not appear in the PBL, but it will be in the audit log entry for this event.
	Type: String
destpath (Required, once)	The path to move the destination file to. For example, if the complete file URL is /pbl/zork/pub/foo/bar/test.txt, the srcpath would be /foo/bar/test.txt. Two important things to note about this parameter:
	1. If you specify a path which does not exist, that path will be automatically created for you as part of the move.
	2. If the destpath parameter ends with a slash ("/"), the destination will be assumed to be a directory. If

it does not end with a slash, the destination will be assumed to be a file.

Type: String

destprj (zero or once)

The destination project for the file. You must have permissions to upload files into both the `srcprj` and the `destprj`, or the move will fail. If not specified, defaults to `srcprj`.

Type: String

desttype (zero or once)

The destination type of the file. Valid values are '`pub`' and '`priv`'. If not specified, defaults to `srctype`.

Type: String

force (zero or once)

If the `force` option is present, the file will be moved even if a file with that name currently exists, and the previous file will be deleted and replaced with this file. The only valid value for this parameter is `True`.

Type: String

sig (Required, once)

API authentication hash signature.

Type: String

srcpath (Required, once)

The path to the source file being operated on. For example, if the complete file URL is `/pbl/zork/pub/foo/bar/test.txt`, the `srcpath` would be `/foo/bar/test.txt`

Type: String

srcprj (Required, once)

The project in which the source file is located.

Type: String

srctype (Required, once)

The source type of the file. Valid values are '`pub`' and '`priv`'.

Type: String

userid (Required, once)

The login name of the user initiating the request.

Type: String

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

PblUpload
Project Build Library (PBL) file and directory upload interface. This function is used to create files and directories in the Project Build Library. Unless you are building your own PBL client, this function will not be very useful to you.

URL

```
/cubit_api/1/pbl_upload
```

Authentication

This method requires *authentication* using an API key.

Parameters

comment (zero or once)	An optional comment to leave about the operation being performed. This will not appear in the PBL, but it will be in the audit log entry for this event. Type: String
desc (zero or once)	The text description of the file. Type: String
file (Required, once)	URL-encoded name and contents of the file, encoded as per *RFC 1867*. Type: String
force (zero or once)	If the `force` option is present, the file will be uploaded even if a file with that name currently exists. A file cannot be uploaded over a directory of the same name, even if the `force` parameter is present. The only valid value for this parameter is `True`. Type: String
md5sum (Required, once)	The md5 checksum of the file being uploaded. Type: String
path (Required, once)	The path to the file being operated on. For example, if the complete file URL is `/pbl/zork/pub/foo/bar/test.txt`, the `path` would be `/foo/bar`. Note: the `path` parameter is used slightly differently in this method than in other methods, since the filename is not appended to the end of the parameter. Type: String
proj (Required, once)	The name of the project which contains the file we are operating on. Type: String
sig (Required, once)	API authentication hash signature. Type: String
type (Required, once)	The type of file that we are operating on. Valid values are 'pub' and 'priv'. Type: String
userid (Required, once)	The login name of the user initiating the request. Type: String

Example Response

```
<?xml version='1.0'?>
```

```
<cubit version='1'>
    <status>OK</status>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

ProfileAdd

Add a profile to the system.

The following conditions must hold for a profile to be added by name:

- The `profile_name` must be specified.
- The `profile_file` must not specified.
- The profile must already exist in SVN.
- The user calling this method must be a Domain Admin.

The following conditions must hold for a profile to be uploaded:

- The `profile_file` must be specified.
- The `profile_name` must not be specified
- The user calling this method must have permission to add profiles to the specified project.

URL

`/cubit_api/1/add_profile`

Authentication

This method requires *authentication* using an API key.

Parameters

`can_users_modify` **(Required, once)**	Set this to `True` if user's with Root Access or better should be able to make modifications to the project. If set to `False`, only the owning user and users with Project Admin access (or better) will be able to modify the profile.
	Type: String
`is_prebuilt` **(Required, once)**	Set this to `True` if this profile a VMWare or other prebuilt image, `False` if this profile is installed via a network install method.
	Type: String
`is_public` **(Required, once)**	If the profile is public (i.e can be built by any project). Valid values are `True` or `False`.
	Type: String
`owner` **(zero or once)**	Lab Management user that this profile will be associated with. The user must be a member of the owning project in order to be able to modify the profile. If unset, this profile will not belong to any particular user.
	Type: String
`profile_file` **(zero or once)**	The filename of a profile to be uploaded. The specified file must:

1. Exist on your local system.
2. Contain a valid profile name. Profile names can only contain letters, numbers, underscores ("_"), and dashes ("-"). Profile names are case sensitive.
3. Not currently exist as a profile name on the system.
4. Be a valid XML file, and contain all the needed attributes for a Lab Management profile.

profile_name (zero or once)	The name of a profile that already exists in Lab Management's Subversion Repository.
	Type: String
project (zero or once)	Project that will own this profile. If unset, this profile will not belong to any particular project.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
summary (Required, once)	Summary of the purpose of this profile.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
Successful addition of the profile rhel3_base:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
    <output>rhel3_base</output>
</cubit>

If the user is unauthorized:
<?xml version='1.0'?>
<cubit version='1'>
    <error>Failed to add profile rhel3_base.</error>
    <output>User 'userid' does not have Domain Admin rights.</output>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

ProfileDelete
Delete a profile from the system.

Profiles can be deleted under the following conditions:

- No host has ever been built or is currently building with this profile.
- The user calling this method has permission to modify this profile.

URL

`/cubit_api/1/delete_profile`

Authentication

This method requires *authentication* using an API key.

Parameters

profiles (Required, once)	List of comma-separated, profile names to delete. Profiles must never have been used to build any hosts. The user invoking this API must have modify rights on the profiles.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
Successful delete of the profile rhel3_base:
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
    <output>rhel3_base</output>
</cubit>

If the user is unauthorized:
<?xml version='1.0'?>
<cubit version='1'>
    <error>0 of 1 profiles deleted successfully. Failed profiles below.</
error>
    <output>rhel3_base: You do not have modify rights on this profile.</
output>
</cubit>

If a deletion of profiles partially succeeds:
<?xml version='1.0'?>
<cubit version='1'>
    <error>7 of 8 profiles deleted successfully. Failed profiles below.</
error>
    <output>rhel3_base: This profile has been used to build hosts.</
output>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

ProfileGetMyPkglist
List the packages associated with the profile assigned to the calling host.

URL

`/cubit_api/1/getmypkglist`

Authentication

This method does not require authentication.

Parameters

`output` **(Required, once)**	Output type, either 'txt' (for text output) or 'xml' (for XML output).
	Type: String

Example Response

```
Sample XML output:
<?xml version='1.0'?>
<cubit version='1'>
  <package version="1.4.3-6">libtool-libs</package>
  <package version="2.1.15-10">cyrus-sasl-md5</package>
  <package version="20020927-11.30.4">iputils</package>
  <package version="3.03-28">perl-HTML-Tagset</package>
  <package version="1.4.1-2">python-optik</package>
  ...
  <package version="1.7-23">time</package>
  <package version="3.2.3-54">cpp</package>
  <package version="8.3.5-92.4">tcl</package>
  <package version="0.2.3-7.1">audiofile</package>
</cubit>

Sample text output:
libtool-libs  1.4.3-6
cyrus-sasl-md5  2.1.15-10
iputils  20020927-11.30.4
perl-HTML-Tagset  3.03-28
python-optik  1.4.1-2
...
time  1.7-23
cpp  3.2.3-54
tcl  8.3.5-92.4
audiofile  0.2.3-7.1
```

Response Codes

• `200` - ok

ProfileGetPkglist
List packages associated with a particular version or revision of a profile.

If neither version nor revision is set, the latest (HEAD) revision of the profile is selected. You must have permissions to view the profile in order to retrieve a package listing. The profile must be public, or you must be a valid user in the project that owns the profile.

URL

`/cubit_api/1/getpkglist`

Authentication

This method requires *authentication* using an API key.

Parameters

`output` (zero or once)	Ouput mode. 'txt' or 'xml' Type: String
`profile` (Required, once)	Profile to get package list for. Type: String
`revision` (zero or once)	Revision of profile to get package list for. Type: Integer
`sig` (Required, once)	API authentication hash signature. Type: String
`userid` (Required, once)	The login name of the user initiating the request. Type: String
`version` (zero or once)	Version of profile to get package list for. Type: Integer

Example Response

```
(output='xml'):

<?xml version='1.0'?>
<cubit version='1'>
  <package version="1.2.3">pkgname1</package>
  <package version="4.5.6">pkgname2</package>
  <package version="7.8.9">pkgname3</package>
</cubit>

(output='txt'):
pkg1  1.2.3
pkg2  4.5.6
pkg3  7.8.9
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

QueryAlloc
Determine, at either a *domain*, *project*, *user*, or *host* level, how hosts are allocated and who they are allocated to.

Users are only allowed to get information about hosts which they have access to see. For example, if you are not a member of a project, you cannot query that project's host allocations. If you query the domain, you will only see allocation information for projects to which you belong. This method is used by Lab Management internally, and it is made available for users as well, since they may find this data useful.

URL

`/cubit_api/1/query_alloc`

Authentication

This method requires *authentication* using an API key.

Parameters

full (zero or once)	If full=true then the complete information about the host is printed.
	Type: String
maxhosts (zero or once)	The maximum number of hosts to return for the query. If type=domain, this will be the maximum number of hosts to return per project (i.e. if this is set to 5 and there are two projects in the domain, up to 10 hosts may be returned). If type=host, this parameter will not be used. If maxhosts is set, the returned hosts will be sorted by the host's name.
	Type: Integer
name (zero or once)	The name of the host, project, or user to be queried. If type=domain, this argument is not required. If type={host,project} then this argument is required. If type=user, this argument is optional, defaulting to the name of the user who you authenticate as. Only Lab Management Domain Admins are allowed to query user allocations for users other than themselves.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
skiphosts (zero or once)	The number of initial hosts that will be skipped over when returning the results. This only has an effect when maxhosts is also used. If type=domain, the hosts skipped are per per project. If type=hosts, this paramenter will not be used.
	Type: Integer
type (Required, once)	The type of service. Valid values are host, project, user and domain.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
Here is a sample output where we query type=host and
name=cu011.dev.cubitdemo.net:

<?xml version='1.0'?>
<cubit version='1'>
  <project name="zork">
    <host version="2">
      <name>cu011.dev.cubitdemo.net</name>
      <alloc_user>yz</alloc_user>
      <alloc_project>zork</alloc_project>
      <labels>
        <label name="cubit_user">yz</label>
        <label name="cubit_project">zork</label>
      </labels>
    </host>
```

```
        </project>
</cubit>

Here is a sample output where we query type=host,
name=cu011.dev.cubitdemo.net and full=True:

<?xml version='1.0'?>
<cubit version='1'>
      <project name=zork>
        <host version="2">
<name>cu011.dev.cubitdemo.net</name>
<alloc_user>yz</alloc_user>
<alloc_project>zork</alloc_project>
<hardware>
        <vendor>Collabnet</vendor>
        <modelname>HP DL380</modelname>
        <ncpu>2</ncpu>
        <cpuarch>i386</cpuarch>
        <cpuname>Xeon</cpuname>
        <cpumhz>1400</cpumhz>
        <memsize unit="MB">2048</memsize>
        <disksize unit="GB">144</disksize>
</hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_base</label>
    <label name='install_mode'>kickstart-net</label>
  <label name='cubit_state'>Allocated</label>
  <label name='cubit_alloc_time'>1170799967</label>
  <label name='cubit_state_mtime'>1170799967</label>
  <label name='cubit_build_time'>1169682748</label>
  <label name='cubit_project'>zork</label>
  <label name='cubit_user'>yz</label>
  <label name='vmware-server'/>
  <label name='cubit_lom_addr'>cu3-l.sjc.collab.net</label>
</labels>
<groups>
  <aigroup>DEFAULT</aigroup>
</groups>
<network>
  <routing>
    <default>
     <ipv4_addr>192.168.202.1</ipv4_addr>
    </default>
  </routing>
     <interface>
         <name>eth0</name>
         <sol_name>e1000g0</sol_name>
         <mac_addr>00:16:35:c3:9b:ed</mac_addr>
         <bootproto>static</bootproto>
         <addr>
             <ipv4_addr>192.168.202.21</ipv4_addr>
             <ipv4_mask>255.255.255.0</ipv4_mask>
             <dnsname>cu011.dev.cubitdemo.net</dnsname>
         </addr>
     </interface>
</network>
</host>

        </project>
</cubit>

Here is a sample output where we query type=domain:

<?xml version='1.0'?>
```

```
<cubit version='1'>
  <domain>
  <project name="zork">
      <host>
        <name>cu011.dev.cubitdemo.net</name>
        <alloc_user>yz</alloc_user>
        <alloc_project>zork</alloc_project>
        <labels>
          <label name="cubit_user">yz</label>
          <label name="cubit_project">zork</label>
        </labels>
      </host>
      <host>
        <name>cu012.dev.cubitdemo.net</name>
        <alloc_user>yz</alloc_user>
        <alloc_project>zork</alloc_project>
        <labels>
          <label name="cubit_user">yz</label>
          <label name="cubit_project">zork</label>
        </labels>
      </host>
  </project>
  </domain>
</cubit>
```

Here is a sample output where we query type=domain and full=True:

```
<?xml version='1.0'?>
<cubit version='1'>
  <domain>
  <project name="zork">
      <host version="2">
<name>cu011.dev.cubitdemo.net</name>
<alloc_user>yz</alloc_user>
<alloc_project>zork</alloc_project>
<hardware>
        <vendor>Collabnet</vendor>
        <modelname>HP DL380</modelname>
        <ncpu>2</ncpu>
        <cpuarch>i386</cpuarch>
        <cpuname>Xeon</cpuname>
        <cpumhz>1400</cpumhz>
        <memsize unit="MB">2048</memsize>
        <disksize unit="GB">144</disksize>
</hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_base</label>
    <label name='install_mode'>kickstart-net</label>
  <label name='cubit_state'>Allocated</label>
  <label name='cubit_alloc_time'>1170799967</label>
  <label name='cubit_state_mtime'>1170799967</label>
  <label name='cubit_build_time'>1169682748</label>
  <label name='cubit_project'>zork</label>
  <label name='cubit_user'>yz</label>
  <label name='vmware-server'/>
  <label name='cubit_lom_addr'>cu3-l.sjc.collab.net</label>
</labels>
<groups>
  <aigroup>DEFAULT</aigroup>
</groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1</ipv4_addr>
```

```
            </default>
        </routing>
        <interface>
            <name>eth0</name>
            <sol_name>e1000g0</sol_name>
            <mac_addr>00:16:35:c3:9b:ed</mac_addr>
            <bootproto>static</bootproto>
            <addr>
                <ipv4_addr>192.168.202.21</ipv4_addr>
                <ipv4_mask>255.255.255.0</ipv4_mask>
                <dnsname>cu011.dev.cubitdemo.net</dnsname>
            </addr>
        </interface>
</network>
</host>


        <host version="2">
<name>cu012.dev.cubitdemo.net</name>
<alloc_user>yz</alloc_user>
<alloc_project>zork</alloc_project>
<hardware>
        <vendor>Collabnet</vendor>
        <modelname>HP DL380</modelname>
        <ncpu>1</ncpu>
        <cpuarch>i386</cpuarch>
        <cpuname>Xeon</cpuname>
        <cpumhz>1400</cpumhz>
        <memsize unit="MB">2048</memsize>
        <disksize unit="GB">144</disksize>
</hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_all</label>
    <label name='install_mode'>kickstart-net</label>
  <label name='cubit_state'>Allocated</label>
  <label name='cubit_alloc_time'>1170800012</label>
  <label name='cubit_state_mtime'>1170800012</label>
  <label name='cubit_build_time'>1169687903</label>
  <label name='cubit_project'>zork</label>
  <label name='cubit_user'>yz</label>
  <label name='vmware-server'/>
  <label name='cubit_lom_addr'>cu4-l.sjc.collab.net</label>
</labels>
<groups>
  <aigroup>DEFAULT</aigroup>
</groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1</ipv4_addr>
    </default>
  </routing>
        <interface>
            <name>eth0</name>
            <sol_name>e1000g0</sol_name>
            <mac_addr>00:16:35:5a:fe:39</mac_addr>
            <bootproto>static</bootproto>
            <addr>
                <ipv4_addr>192.168.202.22</ipv4_addr>
                <ipv4_mask>255.255.255.0</ipv4_mask>
                <dnsname>cu012.dev.cubitdemo.net</dnsname>
            </addr>
        </interface>
</network>
</host>
```

```
    </project>
  </domain>
</cubit>

Here is a sample output where we query type=project and name=zork:

<?xml version='1.0'?>
<cubit version='1'>
  <project name="zork">
      <host version="2">
        <name>cu011.dev.cubitdemo.net</name>
        <alloc_user>yz</alloc_user>
        <alloc_project>zork</alloc_project>
        <labels>
          <label name="cubit_user">yz</label>
          <label name="cubit_project">zork</label>
        </labels>
      </host>
      <host version="2">
        <name>cu012.dev.cubitdemo.net</name>
        <alloc_user>yz</alloc_user>
        <alloc_project>zork</alloc_project>
        <labels>
          <label name="cubit_user">yz</label>
          <label name="cubit_project">zork</label>
        </labels>
      </host>
      <host version="2">
        <name>cu013.dev.cubitdemo.net</name>
        <alloc_user>grue</alloc_user>
        <alloc_project>zork</alloc_project>
        <labels>
          <label name="cubit_user">grue</label>
          <label name="cubit_project">zork</label>
        </labels>
      </host>
      <host version="2">
        <name>cu014.dev.cubitdemo.net</name>
        <alloc_user>root</alloc_user>
        <alloc_project>zork</alloc_project>
        <labels>
          <label name="cubit_user">root</label>
          <label name="cubit_project">zork</label>
        </labels>
      </host>
      <host version="2">
        <name>cu019.dev.cubitdemo.net</name>
        <alloc_user>root</alloc_user>
        <alloc_project>zork</alloc_project>
        <labels>
          <label name="cubit_user">root</label>
          <label name="cubit_project">zork</label>
        </labels>
      </host>
      <host version="2">
        <name>cu022.dev.cubitdemo.net</name>
        <alloc_user>grue</alloc_user>
        <alloc_project>zork</alloc_project>
        <labels>
          <label name="cubit_user">grue</label>
          <label name="cubit_project">zork</label>
        </labels>
      </host>
      <host version="2">
```

```
               <name>cu091.dev.cubitdemo.net</name>
               <alloc_user>root</alloc_user>
               <alloc_project>zork</alloc_project>
               <labels>
                 <label name="cubit_user">root</label>
                 <label name="cubit_project">zork</label>
               </labels>
            </host>
      </project>
</cubit>
```

Here is a sample output where we query type=project, name=zork and full=True:

```
<?xml version='1.0'?>
<cubit version='1'>
  <project name="zork">
      <host version="2">
<name>cu011.dev.cubitdemo.net</name>
<alloc_user>yz</alloc_user>
<alloc_project>zork</alloc_project>
<hardware>
        <vendor>Collabnet</vendor>
        <modelname>HP DL380</modelname>
        <ncpu>2</ncpu>
        <cpuarch>i386</cpuarch>
        <cpuname>Xeon</cpuname>
        <cpumhz>1400</cpumhz>
        <memsize unit="MB">2048</memsize>
        <disksize unit="GB">144</disksize>
</hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_base</label>
    <label name='install_mode'>kickstart-net</label>
  <label name='cubit_state'>Allocated</label>
  <label name='cubit_alloc_time'>1170799967</label>
  <label name='cubit_state_mtime'>1170799967</label>
  <label name='cubit_build_time'>1169682748</label>
  <label name='cubit_project'>zork</label>
  <label name='cubit_user'>yz</label>
  <label name='vmware-server'/>
  <label name='cubit_lom_addr'>cu3-l.sjc.collab.net</label>
</labels>
<groups>
  <aigroup>DEFAULT</aigroup>
</groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1</ipv4_addr>
    </default>
  </routing>
      <interface>
          <name>eth0</name>
          <sol_name>e1000g0</sol_name>
          <mac_addr>00:16:35:c3:9b:ed</mac_addr>
          <bootproto>static</bootproto>
          <addr>
              <ipv4_addr>192.168.202.21</ipv4_addr>
              <ipv4_mask>255.255.255.0</ipv4_mask>
              <dnsname>cu011.dev.cubitdemo.net</dnsname>
          </addr>
      </interface>
</network>
```

```xml
        </host>

        <host version="2">
<name>cu012.dev.cubitdemo.net</name>
<alloc_user>yz</alloc_user>
<alloc_project>zork</alloc_project>
<hardware>
        <vendor>Collabnet</vendor>
        <modelname>HP DL380</modelname>
        <ncpu>1</ncpu>
        <cpuarch>i386</cpuarch>
        <cpuname>Xeon</cpuname>
        <cpumhz>1400</cpumhz>
        <memsize unit="MB">2048</memsize>
        <disksize unit="GB">144</disksize>
</hardware>
<labels>
  <label name='cfprofile' rev='322' version='4'>rhel3_all</label>
    <label name='install_mode'>kickstart-net</label>
  <label name='cubit_state'>Allocated</label>
  <label name='cubit_alloc_time'>1170800012</label>
  <label name='cubit_state_mtime'>1170800012</label>
  <label name='cubit_build_time'>1169687903</label>
  <label name='cubit_project'>zork</label>
  <label name='cubit_user'>yz</label>
  <label name='vmware-server'/>
  <label name='cubit_lom_addr'>cu4-l.sjc.collab.net</label>
</labels>
<groups>
  <aigroup>DEFAULT</aigroup>
</groups>
<network>
  <routing>
    <default>
      <ipv4_addr>192.168.202.1</ipv4_addr>
      </default>
  </routing>
    <interface>
        <name>eth0</name>
        <sol_name>e1000g0</sol_name>
        <mac_addr>00:16:35:5a:fe:39</mac_addr>
        <bootproto>static</bootproto>
        <addr>
            <ipv4_addr>192.168.202.22</ipv4_addr>
            <ipv4_mask>255.255.255.0</ipv4_mask>
            <dnsname>cu012.dev.cubitdemo.net</dnsname>
        </addr>
    </interface>
</network>
</host>

        <host version="2">
<name>cu013.dev.cubitdemo.net</name>
<alloc_user>grue</alloc_user>
<alloc_project>zork</alloc_project>
<hardware>
        <vendor>Collabnet</vendor>
        <modelname>HP DL380</modelname>
        <ncpu>1</ncpu>
        <cpuarch>i386</cpuarch>
        <cpuname>Xeon</cpuname>
        <cpumhz>1400</cpumhz>
        <memsize unit="MB">1024</memsize>
        <disksize unit="GB">25</disksize>
```

```
      </hardware>
      <uuid>51 51 bb a0 00 15 15 51-81 9d fb 47 77 00 51 61</uuid>
      <labels>
        <label name='cfprofile' rev='322' version='6'>rhel4_all</label>
          <label name='install_mode'>kickstart-net</label>
        <label name='cubit_state'>Allocated</label>
        <label name='cubit_alloc_time'>1170960674</label>
        <label name='cubit_state_mtime'>1170963906</label>
        <label name='cubit_build_time'>1170963906</label>
        <label name='cubit_project'>zork</label>
        <label name='cubit_user'>grue</label>
        <label name='vmware-guest'>cu011.dev.cubitdemo.net</label>
        <label name='cubit_lom_addr'>cu011.dev.cubitdemo.net</label>
      </labels>
      <groups>
        <aigroup>DEFAULT</aigroup>
      </groups>
      <network>
        <routing>
          <default>
            <ipv4_addr>192.168.202.1</ipv4_addr>
          </default>
        </routing>
          <interface>
              <name>eth0</name>
              <sol_name>e1000g0</sol_name>
              <mac_addr>00:50:56:00:00:01</mac_addr>
              <bootproto>static</bootproto>
              <addr>
                  <ipv4_addr>192.168.202.23</ipv4_addr>
                  <ipv4_mask>255.255.255.0</ipv4_mask>
                  <dnsname>cu013.dev.cubitdemo.net</dnsname>
              </addr>
          </interface>
      </network>
      </host>
        </project>
      </cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

QueryMembership
Determine the projects and the hosts that the user has permission to access.

If name parameter is not passed, the permission details of the initiating user will be displayed. Only Lab Management Domain Admins are permitted to query users other than themselves.

URL

`/cubit_api/1/query_membership`

Authentication

This method requires *authentication* using an API key.

Parameters

name (zero or once)	The name of the user to get the permission details. This argument is optional, defaulting to the name of the user who you authenticate as. Only Lab Management Domain Admins are permitted to query users other than themselves.
	Type: String
sig (Required, once)	API authentication hash signature.
	Type: String
userid (Required, once)	The login name of the user initiating the request.
	Type: String

Example Response

```
Here is a sample output where we query name=grue:

<?xml version='1.0'?>
<cubit version='1'>
  <user name="grue">
    <project name="testing">
      <summary>Cubit Testing</summary>
      <roles>CUBIT - Domain Admin</roles>
    </project>
    <project name="zim">
      <summary>invador</summary>
      <roles>CUBIT - Domain Admin</roles>
    </project>
    <project name="zork">
      <summary>For dorks</summary>
      <roles>CUBIT - Domain Admin</roles>
    </project>
  </user>
</cubit>
```

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

Status

Return the status of the Lab Management web service. This provides some indication of whether or not the Lab Management web service is working properly. The only status returned is `OK`.

URL

`/cubit_api/1/status`

Authentication

This method does not require authentication.

Parameters

This method does not take any parameters.

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
</cubit>
```

Response Codes

- `200` - ok

StatusSigned
Return the status of the Lab Management web service, and the name of the authenticated user when authentication is used via a signed method.

This provides proof that the API, and authentication to the API are working properly. The only status returned is `OK`. If the authentication is not successful, you will get a `Permission Denied` error. If the API key is expired, you will get a `Permission Denied: API key expired` error.

URL

`/cubit_api/1/status_signed`

Authentication

This method requires *authentication* using an API key.

Parameters

`sig` **(Required, once)**	API authentication hash signature.
	Type: String
`userid` **(Required, once)**	The login name of the user initiating the request.
	Type: String

Example Response

```
<?xml version='1.0'?>
<cubit version='1'>
    <status>OK</status>
    <userid>grue</userid>
</cubit>
```

In addition to returning the OK message, the username of the authenticated user is provided.

Response Codes

- `200` - ok
- `400` - Login failed / Insufficient permissions

UserCanLogin

Given a userid and a hostname, return whether or not the user is authorized to login provided that the appropriate credentials are presented.

The HTTP response code will always be 200 for a successful query whether or not access is permitted. The document body will be text only, and contain one of the following:

- `0`: no access
- `1`: user level access
- `2`: root level access
- `3`: host owner access
- `4`: project admin access

URL

`/cubit_api/1/can_login`

Authentication

This method does not require authentication.

Parameters

host (Required, once)	The fully qualified hostname of the host to test for access.
	Type: String
userid (Required, once)	The Lab Management userid of the user to test for access.
	Type: String

Example Response

```
3
```

Response Codes

- `200` - ok

Frequently asked questions about CollabNet TeamForge 8.2

Here is some background knowledge to help you make the most of TeamForge.

TeamForge installation/upgrade/configuration/administration FAQs

To install CollabNet TeamForge, you download the software, make decisions about how you want the site to work, and set up data for the site to work with.

Before you begin the installation you will want to consider hardware requirements and other factors like supported software. For more information about installation see *Platform specification for TeamForge 8.2* on page 747.

How many servers do I need to run a CollabNet TeamForge site?

You can run TeamForge on one server or split up its services among multiple servers.

TeamForge functionality is delivered by several discrete services. Each service can run on its own machine or share a machine with one or more other services. You assign specific services to specific boxes when you customize your TeamForge installation by editing the `site-options.conf` file.

TeamForge core functionality	This is known internally as the `app` server. It implements JBoss and Apache services. One and only one instance of this application must be present.
Database	The `database` application handles site users' data. You set the type of database by setting the value of the *DATABASE_TYPE* token to oracle or pgsql. One and only one instance of this application must be present.
Subversion	Subversion can be used to provide source control functionality. It uses the Tomcat and Apache services. A site can have zero, one, or more than one instances of the `svn` application, running on an arbitrary number of machines.
CVS	CVS can be used to provide source control functionality. It uses the Tomcat service. A site can have zero, one, or more than one instances of the `cvs` application, running on an arbitrary number of machines.
etl	The reporting service.
datamart	A mirror of your site database for the reporting engine to work with.
Orchestrate	TeamForge Orchestrate is a TeamForge capability that provides traceability for product life cycle activities such as work items, SCM commits, continuous integration (CI) builds, and code reviews. See Orchestrate's *Installation requirements* on page 618 for more information.
Git	TeamForge 8.2 supports an integration with the Git distributed version control tool powered by `Gerrit`. TeamForge 8.2 supports Git integration based on Gerrit

2.11.x (version 8.5.x of the integration). For more information, see *Set up Git* on page 719.

Black Duck Code Sight (BDCS)

TeamForge 8.2 supports the Black Duck Code Sight source code search engine. For more information, see *Set up Black Duck Code Sight* on page 727.

In principle, services can be combined in any configuration, with some constraints, such as:

- Only one site database, datamart and reporting service.
- Multiple source control integration services, but only one per box.

In practice, CollabNet has identified few configurations as the most useful for a wide variety of site. You can follow the instructions here to set up your site in one of these configurations, or you can adapt one of them to your own conditions.

Should I upgrade to TeamForge 8.2 on a new server?

You can upgrade TeamForge on new hardware, or you can stick with the same server your current site is running on. To choose, consider how your members use your site and how you maintain it.

Upgrade on new hardware

Upgrading on new hardware is a little more complex, but provides more flexibility.

Do it this way if you want to:

- Change to a different operating system.
- Stage your upgraded site before moving it into production.
- Serve the site from a different data center.
- Support an expanded user base.
- Minimize downtime associated with the upgrade.

Upgrade on the same hardware

If you are OK with your current hardware setup and you don't want to rock the boat, you can run your new TeamForge site on the same box where your current site is running.

Do this if you want to:

- Keep your upgrade as simple as possible.
- Stick with a tested hardware setup.
- Use virtual hardware.
- Support a stable user base.

Should I move my TeamForge database to its own server?

If you expect your site to have heavy user traffic, you may want to run the site's database on its own server.

☞ **Note:** Before moving your database to its own server, make sure you have access to someone with advanced skills in the database service you are using.

The advantage of hosting a service on a separate server is that it does not share CPU, RAM or I/O bandwidth with the server that is hosting the main TeamForge application.

☞ **Note:** Each TeamForge site can have only one database server.

To help decide whether you need a separate database server, consider these approximate values:

	Shared TeamForge-database server	Standalone database server
Daily users	Fewer than 1000	More than 1000
Daily discussion forum entries	Fewer than 1000	More than 1000

Should I move my source control application to its own server?

If you anticipate heavy source code check-in and check-out traffic, consider setting up the source code application on its own server.

To host your source control services on their own server, you must set up a source code repository server and integrate it with TeamForge. You can integrate any number of source code servers with your TeamForge site.

The advantage of hosting a service on a separate server is that it does not share CPU, RAM or I/O bandwidth with the server that is hosting the main TeamForge application.

☞ **Note:** If you need to move a source code integration, contact your CollabNet representative for help.

To help decide whether you need a separate source control server, consider these approximate values:

	Shared TeamForge-SCM server	Standalone SCM server
Daily source code commits	Fewer than 1000	More than 1000

What are the right PostgreSQL settings for my site?

Your site's PostgreSQL settings depend on the conditions your site is operating under, especially the number and size of projects and the number of users.

The default values in the `site-options.conf` file are designed for a TeamForge site running on a system with 8 GB of RAM. This table contains recommended values for systems with various amounts of RAM, based on testing carried out in CollabNet's performance lab. Use your discretion in selecting the right values for your environment.

☞ **Note:** Remember to recreate the runtime environment after changing any value in the `site-options.conf` file.

Recommended values if PostgreSQL and TeamForge are on the same server

site-options.conf tokens	8GB RAM	16GB RAM	32GB RAM	64GB RAM	128GB RAM
PGSQL_EFFECTIVE_CACHE_SIZE=	4GB	6GB	12GB	24GB	48GB
PGSQL_SHARED_BUFFERS=	1GB	2GB	4GB	8GB	8GB
PGSQL_WORK_MEM=	64MB	64MB	64MB	64MB	64MB
PGSQL_WAL_BUFFERS=	16MB	32MB	32MB	32MB	32MB
PGSQL_MAINTENANCE_WORK_MEM=	256MB	615MB	615MB	615MB	615MB

Recommended values if PostgreSQL is on a separate server

site-options.conf tokens	8GB RAM	16GB RAM	32GB RAM	64GB RAM	128GB RAM
PGSQL_EFFECTIVE_CACHE_SIZE=	6GB	12GB	24GB	48GB	96GB

site-options.conf tokens	8GB RAM	16GB RAM	32GB RAM	64GB RAM	128GB RAM
PGSQL_SHARED_BUFFERS=	2GB	4GB	8GB	8GB	8GB
PGSQL_WORK_MEM=	64MB	64MB	64MB	64MB	64MB
PGSQL_WAL_BUFFERS=	16MB	32MB	32MB	32MB	32MB
PGSQL_MAINTENANCE_WORK_MEM=	256MB	615MB	615MB	615MB	615MB

Is it possible to change artifact prefixes in TeamForge?

Artifact prefixes in Teamforge cannot be changed or customized, all artifacts across the site will have an artifact ID of artfXXXXXX.

I have made changes to the James config, how can I enable them without restarting TeamForge?

You can restart James without restarting all of TeamForge with the following command: `/opt/collabnet/teamforge/runtime/scripts/phoenix.sh restart`. Please keep in mind that any changes made to the James config that are not stored in the TeamForge configuration may be overwritten when doing future upgrades.

Why am I getting an error about NoEmailShouldEverBeSentToNoBody@localhost?

This error is caused by the 'nobody' user somehow being set as monitoring one or more items in TeamForge.

You can see if this is the case with your TeamForge install by running the following against your TeamForge database:

```
select count(*) from mntr_subscription where user_id=`user1002`;
```

If the above returns a count greater than zero, then this is the problem. To correct this, do the following:

- Completely stop TeamForge
- Log into the TeamForge database
- Execute the following code: delete from mntr_subscription where user_id=user1002;
- Exit your database session
- Start TeamForge

Note: You should make a backup of your database before attempting this procedure.

How does TeamForge handle third-party applications?

TeamForge relies on many third-party applications to augment or enhance functionality.

TeamForge integrates with additional third party applications, such as Microsoft Office. Support will always make an effort to provide assistance in using third party applications. However, for complete, end-to-end support, customers should consult the application vendor, as the vendor is best equipped to provide the support necessary to use their products.

CVS

Technical Support provides best-effort support for Subversion and CVS client usage issues. TeamForge is not shipped with CVS source control functionality. For best results, contact the vendor for assistance.

The CVS RPM that ships with RedHat Linux Enterprise Server 3 and RedHat Advanced Server 2.1 has a known bug that prevents users who have access to 32 or more CVS repositories from accessing the repositories that are alphabetically after the 31st. This is currently RedHat bug #131124 (*https://bugzilla.redhat.com/bugzilla/*

show_bug.cgi?id=131124). Customers are advised to contact Red Hat for a solution to this bug for any case where users are members of 32 or more CVS repositories on a TeamForge CVS server.

Discussion forum threading

For TeamForge discussion forums to properly thread posts sent in via email, the email message must include either the `References` or `In-Reply-To` header. Email received without both of those headers cannot be threaded accurately and will most likely be treated as a new topic or thread in the discussion. While the lack of either of the headers is not an explicit RFC violation, the inclusion of such headers is considered compliance with section 3.6.4 of *RFC 2822*.

Microsoft Outlook and Lotus Notes are prone to sending mail without at least one of the required headers. There is evidence that Lotus Notes versions 6.5 and newer are capable of sending email that includes at least one of the two required headers. However, older versions of Lotus Notes either do not include the headers, or require special reconfiguration in order to do so.

Microsoft Outlook on its own does include the `In-Reply-To` header. However, mail is sent through a Microsoft Exchange server, that header is stripped off. There are no known versions of Microsoft Exchange server that do not strip these RFC headers from outbound email, and therefore there are no known workarounds.

Contact your IT group or the vendor of your email client with questions or concerns.

When do I run the initial load job?

You can run the initial load job any time after the site is upgraded to TeamForge 8.2. We recommend that you run it before you hand over the site to the users.

You can run the job when the site is in use. On an average, there might be a ten percent degradation in the response time when the job is running. If the job is not run as a part of the upgrade, we recommend that you run it at a time when the site usage is relatively low. The initial load job may consume more time to complete and this depends on the data available in the site. So this job should be triggered when the site is stable after the upgrade and the chances of site shut down is remote.

☞ **Note:** The initial load job can be run immediately in case of a fresh installation.

How do I list all the ETL jobs?

You can list all the ETL jobs by running the following command.

```
./etl-client.py -a
```

How do I run the initial load jobs?

Run the scripts specified below to run the initial load jobs.

```
$ [RUNTIME_DIR]/scripts/etl-client.py -r SCMCommitInitialJob
$ [RUNTIME_DIR]/scripts/etl-client.py -r TrackerInitialJob
```

What happens if I run an initial load job that has already run successfully?

Re-running the initial load job does no harm. The system performs an internal check and the job is aborted.

How do I check the status of the job?

Site administrators will be notified through mails if there is a job failure. The following command will help you identify whether a job is currently running or not.

```
$ [RUNTIME_DIR]/scripts/etl-client.py -a
```

What happens if there is a job failure?

An email will be sent to site administrators in case of a failure. The job should be re-triggered manually. Data will be harvested from the last completed milestone prior to the failure.

Is it mandatory to run the initial load job for a new site?

Yes, it is mandatory to run the initial load job for a new site.

What happens if I miss running the initial load during the installation or upgrade?

You can run this any time after the site is upgraded to TeamForge 8.2. However, the incremental data harvesting is blocked until the initial load is run. This is true for an existing site that is upgraded to TeamForge 8.2 as well as for a new site.

Why does the incremental_etl job fail when the initial load job is not triggered?

The incremental load is made to fail if the initial load is not run. You can ignore this.

Why do ETL jobs fail post TeamForge upgrade?

This could be due to incompatible database and JDBC driver versions. See *Why do ETL jobs fail post TeamForge upgrade?* on page 929

When do I run the Tracker initial load job?

You can run the Tracker initial load job any time after the site is upgraded to TeamForge 8.2. We recommend that you run it before you hand over the site to the users. This task might take few hours depending on the amount of artifact data in the site.

You can run the job when the site is in use. On an average, there is a ten percent degradation in the response time when the job is running. If the job is not run as a part of the upgrade, we recommend that you run it at a time when the site usage is relatively low. The initial load job may consume more time to complete and this depends on the data available in the site. So this job should be triggered when the site is stable after the upgrade and the chances of site outage is remote immediately.

How do I run the job?

```
$ [RUNTIME_DIR]/scripts/etl-client.py -r TrackerInitialJob
```

How do I list all the ETL jobs?

You can list all the ETL jobs by running the following command.

```
./etl-client.py -a
```

How do I check the status of the job?

Site administrators will be notified through mails if there is a job failure. The following command will help you identify whether a job is currently running or not.

```
$ [RUNTIME_DIR]/scripts/etl-client.py -a
```

What happens if there is a job failure?

An email will be sent to site administrators in case of a failure. The job should be re-triggered manually. Data will be harvested from the last completed milestone prior to the failure.

Is it mandatory to run the initial load job for a new site?

Yes, it is mandatory to run the initial load job for a new site.

What happens if I miss running the initial load during the installation or upgrade?

You can run this any time after the site is upgraded to TeamForge 8.2. However, the Tracker incremental data harvesting is blocked until the initial load is run. This is true for an existing site that is upgraded to TeamForge 8.2 as well as for a new site.

Why does the tracker_incremental_etl job fail?

The incremental load is made to fail if the initial load is not run. You can ignore this.

Why do I get an error during tracker incremental load job?

An error (as illustrated below) shows up during the tracker incremental load job in case there is no data to harvest for certain fields.

You can ignore such errors that have no impact on the incremental load job's success or failure.

```
2015-11-02 02:30:05,547 ERROR [TrackerIncrementalJob] SetVar - We can not
 field
 [date_to_harvest_teams] in the previous job entry result!
2015-11-02 02:30:05,940 ERROR [TrackerIncrementalJob] SetVar - We can not
 field
 [date_to_harvest_dependency] in the previous job entry result!
2015-11-02 02:30:06,399 ERROR [TrackerIncrementalJob] SetVar - We can not
 field
 [date_to_harvest] in the previous job entry result!
2015-11-02 02:30:11,278 ERROR [TrackerIncrementalJob] SetVar - We can not
 field
 [date_created] in the previous job entry result!
2015-11-02 02:30:11,493 ERROR [TrackerIncrementalJob] SetVar - We can not
 field
 [date_membership_removed] in the previous job entry result!
```

Why do I get an exception when recreating the runtime?

A JBoss exception was encountered during TeamForge installation (during "recreate runtime"). This could be an issue with the JBoss vault library and is found to occur inconsistently. If you encounter this exception during installation, run the TeamForge install command (recreate runtime) again.

Can I run other java applications in the same JBoss instance as CollabNet TeamForge?

It is recommended to create a separate instance of JBoss or tomcat and deploy your applications there. While it is possible to deploy other applications alongside CTF, you may encounter some errors.

In order to deploy an application into the CTF JBoss instance, you must place the war files into the deploy directory beneath the runtime directory. All directories beneath the runtime directory are recreated each time the application runtime is rebuilt, deleting your applications.

If your applications include the CTF SDK , you may receive class conflicts and errors such as "java.lang.reflect.InvocationTargetException" when attempting to connect to CTF.

Why won't my CollabNet TeamForge virtual machine installation start?

TeamForge won't start, or you receive an error message when trying to access your site.

You may be encountering one of the following issues:

- The TeamForge application server is not running.
- Your organization has exceeded your maximum number of licensed users for the TeamForge download.

- The free trial version supports up to 3 users at no charge.
- The Team edition supports up to 25 users.

  To purchase additional licenses, visit *http://www.collab.net/products/sfee/buyit.html*
  .

- You are attempting to run TeamForge on an unsupported VMware product. The following legacy VMware product versions are not supported:

  - VMware ESX Server 2.x
  - VMware GSX Server 3.x
  - VMware ACE 1.x
  - VMware Workstation 4.x

  For more information about VMware Player and similar products, see *http://www.vmware.com/products/player/*

What does it mean to run CollabNet TeamForge on a virtual machine?

TeamForge can run as a virtual machine image in a VMWare Player. You get all the functionality of TeamForge with the ease of installation and maintenance that comes with VMWare.

To access TeamForge , one user (generally the site administrator) must configure and run the TeamForge application server in VMware Player. When the TeamForge application server is running, other users can access it via a Web browser. These users do not need to run VMware Player.

The TeamForge download may also run on some other VMware products, such as VMware Workstation 5.5. However, these instructions are only for using VMware Player.

Users invoked via su command in TeamForge

TeamForge invokes the super user (su) command during service start or stop.

The following are the operating system users thar are invoked via the su command.

Users	TeamForge component that invokes the user	Purpose
sf-admin	TeamForge integration server (Tomcat)	During service start (collabnet start tomcat)
sf-admin	TeamForge ETL server (Tomcat)	During service start (collabnet start etl)
sf-admin	TeamForge Application Server (JBoss)	During service start (collabnet start jboss)
PostgreSQL	TeamForge database server (Postgres)	During service start/stop (postgresql-9.2 start/stop) and create runtime (install.sh)
bds-codesight	Code search database server (Postgres)	During service start/stop (bdcs-codesight-postgresql start/stop) and create runtime (install.sh)
gerrit	Gerrit integration server (Jetty)	During service start (collabnet start gerrit)

Who is responsible for applying OS updates to the underlying VMware image?

CollabNet does not provide the OS updates. If there is a critical update that needs to be addressed within the VM Image, CollabNet will post instructions on how to update that on open.collab.net.

Why does the the Yum installer display a warning message on Centos 6?

The yum installer displays a warning message on Centos 6 if you are not using the latest version update of Centos or if the new GNU Privacy Guard (GPG) keys are not in place.

On the Centos system, the new GPG keys are generally sent as an update before the expiry of the old keys. You must perform the version update regularly to receive the new GPG keys.

To avoid the warning message, use the "-nopgcheck option" in the yum install command.

```
yum install --nogpgcheck teamforge
```

Why am I getting a Yum repository filename conflict?

This means that you have created a repository filename starting with `collabnet`.

The repository file for the TeamForge installer begins with "collabnet-*.repo". If you have a customized repository with the same name, then the TeamForge installer renames the repository file with ".backup" as an extension and installs the repository file. It is better not to have any repository file name that begins with `collabnet` apart from the one that is installed by the TeamForge installer.

Why is a user restricted and cannot be changed?

If using the TeamForge integration with CVS, the machine running CVS must be using the default Linux authentication method (/etc/passwd, /etc/group) and not a network service (NIS, MS AD, MS winbind, etc).

What's the difference between a TeamForge site in ALM mode and a TeamForge site in SCM mode?

A TeamForge site can offer the full range of application life cycle functionality, or it can be restricted to supporting Subversion source code and the related core functions.

Your site license specifies how many users have ALM licenses and how many have SCM licenses. You can assign these licenses to individual users according to their needs.

*   When your site is running in ALM mode, users with ALM licenses can use the full range of ALM features, while users with SCM licenses can see only the core source code management features.
*   When your site is in SCM mode, all users are restricted to the core source code management features, regardless of their individual license.

As a site administrator, you are prompted to select ALM or SCM mode for your site the first time you log in after installing TeamForge. You can change the mode at any time, if you have the appropriate license.

When you upgrade to TeamForge 8.2 from an earlier version of TeamForge, your site is automatically in ALM mode.

How do TeamForge licenses work?

When you purchase a CollabNet TeamForge license, you get the right to assign licenses to a specified number of users.

What each user can do on your site depends on the type of license you have assigned to that user.

*   Users with ALM licenses can use the full range of ALM features.
*   Users with SCM licenses can see only the core source code management features.

Check with your CollabNet representative if you aren't sure how many licenses or what kind of licenses you have.

*   Your license key contains three important numbers:

    *   The number of users eligible to use the source code management (SCM) features.
    *   The number of users eligible to use both the SCM features and the full set of application lifecycle management (ALM) features.
    *   The IP address of the machine that your site runs on.

For example, if your organization has 80 users who will use only the source code management features and 100 users who need the application lifecycle management features, your license key string may look like this:
`ALM=100:SCM=80:supervillaininc:144.16.116.25.:302D02150080D7853DB3E5C6F67EABC65BD3AC17`

If you are upgrading from TeamForge 5.3 or earlier, all your existing users are considered ALM users.

- Your license key only works for the IP address (or range of addresses) of the hardware that your CollabNet TeamForge is running on, as specified in your order form. If your site uses a separate server for its database or source code repositories, make sure your license key reflects the IP addresses of all the necessary machines. If any of these addresses change, ask your CollabNet representative to generate a new key.
- When you create a user account, you can assign the user an SCM license as long as there are enough available licenses (counting both ALM and SCM). You can assign the user an ALM license if ALM licenses are available. If no more ALM licenses are available, you can only assign SCM licenses.
- You can purchase a TeamForge license for as many years as you want. The validity period is encoded into your license when it is generated by your CollabNet representative.
- How many users your site can support depends on the type of license. Check with your CollabNet representative if you aren't sure what kind of license you have.
- Your license key is attached to the IP address of the server where your site runs. You can get a license key for a single IP address or for a range of IP addresses.
- Your service year starts the first time you log into your site, or the first time you create or edit any item on your site, such as a tracker artifact or a document. Whichever of these events comes first starts the clock.
- The expiration date of your license is shown on the **License Keys** page. (See the *About CollabNet TeamForge* page for the company name associated with the license.)
- When your service year expires, you can still see the project data on your site, but you cannot make any changes to it. However, you can still carry out some critical maintenance functions for your site:

  - Enter a new license key.
  - Deactivate or delete users.
  - Change user passwords.
  - Get forgotten user passwords.
  - Use documentation on *http://help.collab.net*.

What happens when my site license expires?

When your service year expires, you can still see the project data on your site, but you cannot make any changes to it.

Your service year starts the first time you log into your site, or the first time you create or edit any item on your site, such as a tracker artifact or a document. Whichever of these events comes first starts the clock.

The expiration date of your license is shown on the **License Keys** page. (See the *About CollabNet TeamForge* page for the company name associated with the license.)

If your site has switched to read-only mode, you can still carry out some critical maintenance functions:

- Enter a new license key.
- Change users' license types.
- Deactivate or delete users.
- Change user passwords.
- Get forgotten user passwords.
- Use the comprehensive help on *help.collab.net*.

What's the difference between an ALM user license and an SCM user license?

The type of license you have determines what you can do on your TeamForge site.

Your site administrator has assigned you a site license. Your license is of one of two types:

- An ALM license enables you to use the full set of tools on the site, including Source Code, Trackers, Documents, Tasks and Reports.

- An SCM license enables you to use the site for basic source code management. You can browse repositories and participate in Wiki conversations.

If you had an account on a TeamForge 8.2 site or earlier, you automatically have an ALM user license.

How to reinstall a deleted installation directory?

You can reinstall an installation directory if it has been deleted inadvertently.

Before attempting to reinstall the deleted installation directory the remnants must be wiped out completely. Rather, remove the `collabnet-local.repo` file under `/ etc/yum.repos.d` before reinstallation. This is mandatory for a clean and complete installation.

How do I configure the user session timeout?

Configure user session timeout for newly created sessions of the application server and the integration server in site-options.conf by editing the setting SESSION_TIMEOUT.

By default, it is 30 minutes (for security reasons). You must re-create runtime for the changes to take effect.

Where do I configure my client proxy settings?

Configure proxy settings in the servers file (created when Subversion is installed on your system).

The server file is created when you install TortoiseSVN, Eclipse or command-line Subversion. Use the appropriate path from the installation folder to configure proxy settings:

CLI Unix/Linux	/home/<username>/.subversion/servers
CLI Windows	C:\Documents and Settings\<username>\Application Data\Subversion\servers
Eclipse	Window > Preferences > General > Network Connections
IE (6/7)	Tools > Options > Advanced Network > Connection Settings
Firefox	Tools > Internet Options > Connections tab > LAN Settings
TortoiseSVN	Windows Explorer > File > TortoiseSVN > Settings > Network (by default, TSVN uses browser proxy settings)

How do I back up TeamForge?

TeamForge has essentially four data components that require the System Admininstrator to proactively back up in case of system failure. These components are:

- the database
- the filestorage
- the search indexes
- the scm data

As with most large and complex applications, the recommended method of backing up TeamForge involves shutting down the application. While it is possible to back up the application while it is up, the backup itself will not provide 100 percent data consistency. You should undertake one of these live backups only as a last resort and only after ensuring all the potential consequences are fully understood.

If, for whatever reason, a full, offline backup is not possible, you should do the following to ensure that as little data is being changed as possible. While these steps won't completely replicate the offline backup, they can mitigate most of the issues without doing a live backup. Again, we do NOT recommend this. Offline backups are your friend. Just schedule the downtime and take the hit.

For the CVS repositories, you should use whatever normal filesystem backup method your company prefers, much like the FILESTORAGE section above. For SVN repos, it is highly advised that you use the 'svn dump' action to export them to normal files and then back up those dump files.

Please note that due to the nature of SFEE, there are several interdependencies between these various data storage points. As such, you must take great care to ensure that all these components are backed up AT THE SAME TIME. You will NOT be able to use a database backed up yesterday morning with a filestorage from the night before.

Quieting the system

First, you'll want to turn off the web server so users can't get to the UI (assuming you are routing through Apache and not direct to JBOSS): `service httpd stop`

Second, on the SCM server, remove the execute bits on the SCM: `chmod -x /usr/bin/{cvs,svn}`. This will prevent anyone from executing 'cvs' or 'svn' for new checkins/checkouts/tags. Please note however, that it won't affect existing running processes. Either wait for them to finish, or kill them.

Finally, pause the search indexing engine: `touch /opt/collabnet/teamforge/var/searchIndexes/LOCK_INDEXES`

The database

The majority of TeamForge data is stored in the database. To back up the database, follow the recommended procedure from your db vendor. If you're using PostgreSQL, you can use this command: `pg_dump -Fc ctfdb > ctfdb.dump`

The file storage and search indices

Any documents uploaded to TeamForge, or attachments to an artifact or forum, as well as the search indices, are stored on the filesystem as normal system files. Please use whatever normal backup method you use to ensure filesystem restorability (tar, dd, cpio, Ghost, etc). You should back up the following directory and ALL its subdirectories: `/opt/collabnet/teamforge/var`

The SCM data

On the SCM box, you will also need to back up your SCM data. This data is contained in your companies various repositories which are located under: `/cvsroot` and `/svnroot`

Finishing

Once the backup is complete, remove the file: `/opt/collabnet/teamforge/var/searchIndexes/LOCK_INDEXES`

Turn on the execute permissions for the SCM binaries: `chmod +x /usr/bin/{cvs,svn}`

And restart Apache: `service httpd restart`

How do I configure the timeout for Apache in TeamForge?

This can be changed by editing the setting WWW_SERVER_TIMEOUT.

The timeout for Apache (httpd) can be configured in seconds. Default is 300 seconds. You must re-create the runtime for the change to take effect.

How do I make TeamForge work the same when the IP address of the server changes?

Update the /etc/hosts file on the server and the dns record(s) with the new IP address.

When the IP address of the server changes, update the /etc/hosts file on the server and the dns record(s) with the new IP address.

In addition, for TeamForge instance using PostgreSQL database, the IP address used by the PostgreSQL database needs to be changed in the following files:

- /var/lib/pgsql/9.3.6/data/pg_hba.conf - Change the IP of the host entry pointing to the TeamForge server.
- /var/lib/pgsql/9.3.6/data/postgresql.conf - Change the application server's IP for the listen_addresses variable. Restart Postgres service for the changes to take effect.

How do I change the JBoss log level in TeamForge 8.2?

Use the **Configure Logging** page to change the JBoss log level in TeamForge 8.2.

For more information, see *Configure your site's log level* on page 296.

☞ **Note:**

- The default JBoss log level is **INFO**.
- JBoss restart is not required after changing the log level using the **Configure Logging** page.
- Once you change the log level in the **Configure Logging** page, the changes you may do to the `standalone-full.xml` file to change the log level have no effect. In other words, the log level set using the **Configure Logging** page is persisted in the database and therefore overrides the log level in the `standalaone-full.xml` file.

Upgrade to TeamForge 8.0 and later fails if the field 'team' already exists. How do I resolve this?

While upgrading to TeamForge 8.0 (or later) you may encounter an error stating that the field `team` already exists for the tracker(s).

This is because if the trackers in your project already have a field with the name `team`, it conflicts with the Team feature introduced in TeamForge 8.0. This error occurs while running the migration script `migrate.py`. To resolve the issue, do the following and then run the `migrate.py` script again:

1. Identify the tracker(s) where the `team` field is present by running the following script:.

   - For the `folder_layout` table:

     ```
     select folder_layout.id, field_name, title,folder_id tracker_id from
     folder_layout, folder where field_name = 'team' and folder_id like
     'tracker%' and folder.id = folder_layout.folder_id;
     ```
   - For the `field` table:

     ```
     select field.id, name, title, folder_id from field, folder where
     name='team' and folder_id like 'tracker%' and field.folder_id =
     folder.id;
     ```

2. Rename the field `team` as `original_team` by running the following script::

   - For the `folder_layout` table:

     ```
     update folder_layout set field_name='original' || '_team' where
     field_name='team';
     ```
   - For the `field` table:

     ```
     update field set name='original' || '_team' where name='team';
     ```

3. Verify the newly renamed fields by running the following script:

   - For the `folder_layout` table:

     ```
     select folder_layout.id, field_name, title,folder_id tracker_id from
     folder_layout, folder where field_name like '%team' and folder_id like
     'tracker%' and folder.id = folder_layout.folder_id;
     ```

   - For the `field` table:

     ```
     select field.id, name, title, folder_id from field, folder where
     name like '%team' and folder_id like 'tracker%' and field.folder_id =
     folder.id;
     ```

4. Re-run `migrate.py`.

   ☞ **Important:** Before re-running this script, make sure you delete the `ctf72_ctf80` folder (which was created the first time you ran the migration script), because every time you run the script, this folder gets created.

Troubleshooting TeamForge errors

Common errors and solutions in TeamForge.s

Concept definition.

I deleted the email that verifies my CVS password. How can I retrieve my password?

You can sync up your TeamForge login and CVS passwords by changing your password in TeamForge.

To change your password

1. Log in and click on My Workspace.
2. Click on My Settings.
3. Click on **Change Password**.
4. Enter your old password and new password.
5. Click **Submit**.

☞ **Note:** The old and new passwords can be the same if you do not wish to actually change it.

Not all my team members appear in the "assigned to" field. Why is that?

In order to appear in the Assigned To field, the user must belong to a role giving them edit/view permissions on the tracker.

Why don't the branding repo changes get rendered into UI?

It may be due to the property 'subversion_branding.repository_base' pointing to /sf-svnroot instead of the /svnroot directory, which is used by the scm-integration of the csfe installation.

First, check the location of the branding repository in subversion_branding.repository_base=/sf-svnroot' in `/opt/collabnet/teamforge/runtime/conf/sourceforge.properties`.

If it has to be /svnroot, then add an entry that states SUBVERSION_BRANDING_REPOSITORY_BASE=/svnroot

Then re-create a runtime and restart TeamForge.

Why am I getting "Could not connect" status for my email and search server?

On the System Tools page, when you see "Could not connect status for search and email servers," you must stop and start your `phoenix.sh` process.

You may also need to set the *JAVA_HOME* environment variable to the location of your JDK.

The stop/start Phoenix commands:

```
sh /opt/collabnet/teamforge/runtime/scripts/phoenix.sh stop
sh /opt/collabnet/teamforge/runtime/scripts/phoenix.sh start
```

Why am I not able to see the charts for tracker metrics?

You may not be able to see the charts for the tracker metrics if the tracker initial load is not running correctly.

The incremental data collection is disabled until the initial load is run. You can check if the initial load is completed successfully by executing the query below from the Ad-hoc reporting page against the datamart.

```
select status from etl_job where job_name='tracker_initial_etl';
```

You must get the status value as 1 if the initial load is completed successfully. Otherwise, you must trigger the job manually by executing the command: `[RUNTIME_DIR]/scripts/etl-client.py -r TrackerInitialJob`

Post install fails for GIT. What should I do?

Sometimes, the post installation fails during the GIT setup. In such a scenario, you need to manually kill the process using the following command and then do a post install again:

```
kill -9 <ProcessID>
```

Why am I getting a 'Not running' message when the Datamart service is stopped?

When TeamForge and Datamart are running in a single instance, the TeamForge database is stopped when you stop the Postgres services. The message, 'Not running' is displayed when you stop the Datamart service. You can ignore this message.

Why are the dynamic images that TeamForge creates broken?

If you have a fresh install of TeamForge and you're noticing that the dynamic images are not correct, you may be missing a library that is needed to create the images.

The easiest way to find this is to check and see if you have the xorg-x11-depreciated-libs rpm installed:

```
rpm -qva | grep xorg-x11-depreciated
```

Watch for the results. If you see that you have the xorg-x11-depreciated-libs rpm installed, and after a server restart you're still not seeing the images, please open a support request. If you do not have the xorg-x11-depreciated-libs rpm installed, it can usually be installed by performing a simple: `up2date xorg-x11-deprecated-libs` and restarting TeamForge.

Due to firewall restrictions I cannot send email from James. How can I resolve this?

If James is unable to send email directly due to firewall restrictions, or mail being rejected from the application servers IP address, you may have to configure it to use a gateway mail server to send outgoing messages through.

To do this, you will need to add the following to the `<mailet match="All" class="?RemoteDelivery">` directive in the james config file at `/opt/collabnet/teamforge/james/james-<version>/apps/james/SAR-INF/config.xml`:

```
<gateway>smtp.example.com</gateway>
<gatewayPort>25</gatewayPort>
```

You should find these commented out on line 362 of the config file. If your gateway mail server requires authentication to send email, you may also add the following directives:

```
<username>username</username>
<password>password</password>
```

Why is my email taking a long time to arrive?

TeamForge uses the James MTA to send and parse all email coming to and from the system. In this case, the best course of action is to look in the james mailet logfile.

Your logfile will help you to determine what is going on with the emails that are being sent from your system. Your logfiles will look very similar to this:

```
07/02/07 07:54:43
INFO James.Mailet: ?RemoteDelivery:
Attempting delivery of Mail1170135534355-39088-to-domain.invalid to host
 domain.invalid at 192.168.0.1 to
addresses [invalid.user@domain.invalid] 07/02/07 07:55:43
INFO James.Mailet: ?RemoteDelivery: Could not connect to SMTP host:
 192.168.0.1, port: 25; nested exception
is: java.net.?ConnectException: connection to 192.168.0.1 timed out 07/02/07
 07:58:43
```

```
INFO James.Mailet: ?RemoteDelivery: Storing message
Mail1170135534355-39088-to-domain.invalid into outgoing after 7 retries
 07/02/07
07:58:43 INFO James.Mailet: ?RemoteDelivery: Attempting delivery of
Mail1170831482124-2756-to-company.com to host mx.company.com. at 127.0.0.1
 to addresses
[good.user@company.com]
```

As you can see, the James MTA stores outgoing emails to resend at a later time. These files can be located in this directory: /opt/collabnet/teamforge/james/james-<ver>/apps/james/var/mail/outgoing/

When you do a directory listing of the files, you will see a listing of files very similar to this:

```
4D61696C313137303833333231313430313322.Repository.?FileObjectStore
4D61696C313137303833333231313430313322.Repository.?FileStreamStore
```

The FileObjectStore is a binary file, but, the FileStreamStore can be viewed with an editor or your favorite paging program in order to determine the contents. Sometimes, the directory can grow to a large number, where you will not be able to use a standard bash expander to delete all of the files. In that case, use the following shell script to remove all of the objects from the outgoing directory:

```
for i in * ; do /bin/rm $i; done
```

How do I enable the Postgresql log files archiving when the services are not started using the CollabNet startup script?

The Postgresql log files archiving can be enabled by running a simple command.

It is recommended you use the CollabNet init script to start and stop the Postgres services. However, if you use the Postgresql init script to start or stop the Postgres services, the postgresql log files are not archived by default.

To enable the Postgresql log files archiving, run the following command:

```
/etc/init.d/collabnet start pgsql
```

Why am I not getting any error messages when executing the Subversion upgrade script?

Error messages may come when Subversion is installed with a dependent package from an unknown source.

The Subversion working copy script assumes that Subversion is installed with the dependent packages from a proper source repository(RHEL/CollabNet). If you install any dependent packages from any unknown source that is not authorized by RHEL/CollabNet, it will result in inconsistency and this cannot be handled by the Subversion working copy script.

Why do we have errors creating or altering repositories and adding or removing users?

The TeamForge SCM Integration server runs an instance of Tomcat and then launches TeamForge inside the Tomcat container.

If you are experiencing issues creating or altering repositories or adding and removing users from repository access, and the other TeamForge integration logs are not providing any clues, you may wish to review the Tomcat log at: /opt/collabnet/teamforge/log/integration/catalina.out

Sometimes, OS-level errors will be flagged into this log and not others. In our experience, it is pretty rare to find something in this log that is not logged elsewhere.

Why am I getting an email specifying that the ETL job has failed?

You are getting this email because one of the Extract transformation and load (ETL) jobs has failed during the run.

You can see the etl.log for more details to find out the reason for the job failure.

The ETL job failure may happen because of the following reasons:

- Out of memory error.
- No response from the database.

If the ETL job failure is happening for the first time, you can restart the ETL (`[RUNTIME_DIR]/scripts/ collabnet restart etl`) and check if the problem is occurring again. You can increase the JVM heap size by specifying the same in ETL_JAVA_OPTS if the problem keeps recurring. The default value is `-Xms160m - Xmx256m`. You can increase the heap size depending on the memory available in the box.

Check if both TeamForge and Datamart are up and responding to queries if there is no response from the database. Restart the ETL (`[RUNTIME_DIR]/scripts/collabnet restart etl`).

Contact the Collabnet support if the problem persists.

My custom event handler is not working! What is wrong?

If you get a message in server.log (or events.log) that your event handler could not be parsed, these are some points you might check.

Did you change the package of your class containing handler code?	This is not supported.
Did you move `event.xml` to another directory?	This is not supported.
Did you compile your java code with a Java version that is not supported by the JVM running on the TeamForge server?	
Did you use a very long file name for your JAR file?	Only 31 characters are supported.
Did you include libraries in your JAR file that have already been part of the default TeamForge class path?	In this case, your libraries will not be picked up.
Did you try to reference internal TF classes from packages other than the SOAP namespace?	In this case, the TeamForge security classloader will reject your JAR file.
Did you change the DTD location referenced in `events.xml`?	This will not work.
Did you try to trigger a follow up action in a synchronous event handler?	This will most likely result in a time out exception, since your code is already running in a transaction that locks the resources you are trying to obtain a lock for.
Did you trigger a TeamForge SOAP call with the session ID passed to your handler which threw an exception?	See *Using an asynchronous event handler: trigger follow-up events* on page 850 for an explanation why this does not work.

What happens when I have too many users for the available TeamForge licenses?

When there is a mismatch between the number of licenses available and the number of users you assign licenses to, your site will respond in different ways depending on the circumstances.

The number of SCM licenses has been exceeded, but there are unused ALM licenses.	An advisory banner appears on the License page.
The total number of licenses has been exceeded, but there are unused ALM licenses.	The site switches to read-only mode. Users can see all the tools (unless the site was already in SCM mode) but can't do anything with them.
The number of ALM users has been exceeded, but there are unused SCM licenses.	The site switches to SCM mode. Users see only the SCM tools, but can do any operation on them that their permissions allow.

The number of ALM licenses has been exceeded, and the total number of licenses has also been exceeded.	The site switches to SCM mode and read-only mode. Users can see only the SCM tools and everything is read-only.

Why do I get a JBoss error - "failed to start in 240 seconds, giving up now" - while installing TeamForge?

You get this error when the system's RAM is less than the minimum recommended value of 4GB. However, it's most likely that JBoss will start within a few minutes.

To make sure that JBoss starts up, check the `service.log` file using this command:

```
tail -f /opt/collabnet/teamforge/log/apps/service.log
```

If you see messages like the following, the TeamForge application will start in a few minutes.

```
        Check Port Available PASSED: Port 4444 on localhost is available
        Check Port Available PASSED: Port 4445 on localhost is available
        Waiting for application server to start up.. this can take a few
minutes.
```

JBoss crashed with out of memory error, how do I prevent this?

This can indicate that the JVM heap size is set too small.

You can adjust this by changing the -Xms and -Xmx settings of the JBOSS_JAVA_OPTS token in site-options.conf and rebuilding runtime.

This will appear if the JBoss application server has crashed and you find this error in the server.log:

```
INFO [STDOUT] java.lang.OutOfMemoryError: Java heap space
```

The default maximum heap size of 640MB can cause issues on a heavily used site. If the CTF application is the only thing running on the server, you can increase this to half of the total physical ram on the machine. This should still allow enough memory for the OS and other necessary processes. If you are also running the app, database and scm on the same machine a maximum heap size of 1/4 or the total ram maybe a better setting. Determining the right JVM settings for your install will require testing with your particular usage patterns and database.

You can view the current memory usage under the JVM Environment section of the JBoss webconsole at `http://<CTF_SERVER>:8080/web-console/`. You will need to log in using the CTF admin password.

Why would some users not get email?

Check your dnsserver-<date and time>.log.

James records all errors related to resolving DNS for outbound mail to: `/opt/collabnet/teamforge/james/james-<version>/apps/james/logs/dnsserver-.log`. If you find that some of your TeamForge users are receiving email, but significant groups of others are not, you should consult this log to determine if James is experiencing difficulties in resolving their domain or MX records.

Why do I get 'Name is Invalid' error when trying to create a project using `createProject()` method via the SOAP API?

You need to pass certain default parameters to createProject() method.

The default parameters that can be passed to createProject() method are:

```
        (sessionId, name, title, description)
        where, name = URL Name [in UI]
```

```
            title = Project Name [in UI]
```

In TeamForge, URL Name ['name'] accepts only lower case and the numeric characters. If the value of 'name' contains any other character than the aforesaid, user will end up seeing the 'Name is Invalid' error and the project creation would fail.

Why is the password and login shell changed for users on my cvs/svn server?

TeamForge uses local user accounts on the SCM server to provide access to CVS repositories via ssh.

If any local user accounts on the SCM server match user names within TeamForge they will be changed.

If you are planning to use CVS in the SCM server, you should ensure that the local accounts do not conflict with the TeamForge user accounts. Adding a prefix to the local user accounts (local_username) would be one way to resolve this and prevent the usernames from conflicting. Alternatively, if you are not using CVS repositories, the CVS integration can be removed altogether.

Why am I not able to see the status of the Postgres in the collabnet startup script?

You may not be able to see the status of the Postgres if the host name of the *HOST_ token* is set to localhost in a SaaS multibox setup.

The Teamforge installer fails to add the IP address of the database box to the listen address in the `postgresql.conf` file if the host name of the *HOST_ token* is set to localhost in a SaaS multibox setup.

☞   **Note:** You must add the IP address of the database box to the listen address in the `postgresql.conf` file.

What does the "psql: could not connect to the server: No such file or directory" error message mean?

This error indicates that the PostgreSQL database server is not running. You need to restart the server.

Use the following command to start the server:

```
service postgresql start
```

Why do I get a proxy timeout when I try to view certain SCM pages?

If you are getting a proxy timeout error when you try to view a SCM page, you may need to configure the *Apache 2.2 Proxy Timeout* to 300 or less in the *httpd.conf* file.

If you get the following error while attempting to view a SCM page in SFEE:

```
The proxy server received an invalid response from an upstream server.
The proxy server could not handle the request
    GET /integration/viewcvs/viewcvs.cgi/ibe-rules/tags/phases/ibe-
rules_09.02.0-Ph-200902_test_20090105/
Reason: Error reading from remote server
```

Configure your Apache 2.2 Proxy Timeout to 300 or less in the `httpd.conf` file.

Why do search and email server show "Could not connect"?

Typically this means the tomcat container for James and the search service are not running.

You can restart this with the commands shown below. You may need to set the JAVA_HOME environment variable to the location of your JDK.

```
sh /opt/collabnet/teamforge/dist/james/james-2.2.0/bin/phoenix.sh stop
sh /opt/collabnet/teamforge/dist/james/james-2.2.0/bin/phoenix.sh start
```

Why do I get a server status error when I perform a search?

Occasionally, an exceedingly large or complex document causes the search indexing service to abort.

This is typically when all searches in TeamForge return an exid to the user.

Check the server status page and see if the search server is listed as anything other than OK. If it is not OK, then you should restart the search service by logging into the TeamForge application server as root and issuing the following command:

```
/opt/collabnet/teamforge/james/james-/bin/phoenix.sh restart
```

Check the server status page again in TeamForge and ensure that it shows a status of OK. If it shows OK, then searches should now work, and the site will slowly catch up on any indexing requests that were logged while the service was down. If you continue to get exids returned for all searches even with an OK status, then you probably have corrupt search index files.

Why can't TeamForge send my outbound mail?

If you are unable to send email directly due to *firewall* restrictions, or if mail is being rejected by the application server's IP address, configure TeamForge to send outgoing messages through a gateway mail server.

Configure TeamForge to send outgoing message through a gateway mail server by adding the following to the `<mailet match="All" class="RemoteDelivery">` directive in the configuration file at `/opt/collabnet/teamforge/runtime/james/apps/james/SAR-INF/config.xml`:

```
<gateway>smtp.example.com</gateway>
<gatewayPort>25</gatewayPort>
```

If your gateway mail server requires authentication to send email, you may also add the following directives:

```
<username>username</username>
<password>password</password>
```

Why does the SOAP service show "could not connect" on the Server Status page when everything else appears to work?

This can be caused by an incorrect host name in /etc/sourceforge.properties. Rebuilding runtime will correct this, assuming the hostname is set correctly in the site-options.conf file.

This issue can occur when using the restore.py script to restore data from a TeamForge instance with a different hostname.

Why does startup fail or produce errors?

If TeamForge fails to start up, or is starting but is throwing errors on every page, then typically something went wrong during the JBOSS bootstrap process.

Fortunately, JBOSS logs this process to: `/opt/collabnet/teamforge/jboss/jboss-<version>/server/default/log/boot.log`.

TeamForge writes its startup and shutdown info, as well as any system errors to `/usr/local/ soureforge/log/server.log`. If you encounter a system error while using TeamForge, it is logged here. Additionally, if you see an 'exid' string in the application, the Java stack trace for that exid will be logged in this file.

Why do I get a TeamForge system error in the project template creation page?

This may be because of a few stale permissions in the project in which you are trying to create the template.

You can resolve this by identifying and deleting the stale records using this SQL.

```
select role_id from role_operation ro left outer join ia_project_association
 ia
```

```
on (ro.resource_value = ia.id) where ia.id is null and resource_value like
 '%prpl%' ;
```

```
select role_id from operation_cluster ro left outer join
 ia_project_association ia
on (ro.resource_value = ia.id) where ia.id is null and resource_value like
 '%prpl%' ;
```

Why does my CollabNet TeamForge site show a different time than the host machine it is running on?

In some cases it is possible for the clock in the CollabNet TeamForge VMware image to drift from that of the host machine.

If you notice this issue, you can set the CollabNet TeamForge VMware image to synchronize time with an external NTP server.

A script is provided to enable you to configure time synchronization easily. The `configure-ntp.sh` script sets up a manual periodic time sync once per hour between the VMware image and the NTP server.

👉 **Important:** Before running this script, your virtual machine must be able to access an external NTP server. If your virtual machine is running inside a firewall, and is unable to access an external public NTP server, you may need to talk to your system administrator to find an accessible NTP server within your network.

While logged into the virtual machine, run `/root/configure-ntp.sh <ntp server>`.

If you do not enter an NTP server, the script will try to use `pool.ntp.org`, a publicly available time service, by default.

👉 **Note:** VMware advises against setting up the VMware image to use NTP directly because it can interfere with VMware's own built-in time syncing mechanism.

For detailed information about timekeeping in VMware, see *http://www.vmware.com/pdf/vmware_timekeeping.pdf*

How can a user who deleted a document get it back?

It is possible to recover the latest version of a document if someone has accidentally deleted it from TeamForge.

However, note that you can recover the raw document but not undelete it from the app. The user will have to add the document back into TeamForge, and the history of the document will not be recovered. If you need a true undelete so that the history is available, please contact Technical Support and ask for us to schedule our Professional Services group for you.

Let's assume you have a project in TeamForge called Test, and the URL in your browser when you load this project is: `http://your_sfee_server/sf/projects/Test`. Please note the word after projects, this is your project name.

Now, let's assume the file to be retrieved was in a folder called 'foo' in the document manager. Further, let's assume the document was called 'example'. On the database, run the following command:

```
select stored_file.raw_file_id, document.id, item date_last_modified,
 item.title,
            stored_file.file_name, sfuser.full_name from item, folder,
 document, document_version,
            stored_file, sfuser where item.title like '%example%' and
 item.folder_id = folder.id and
            folder.path = 'docman.root.b' and document.id =
 document_version.document_id and
            document_version.stored_file_id = stored_file.id and
 item.is_deleted = 't' and
            item.last_modified_by_id = sfuser.id ;
```

This will return something like:

```
raw_file_id | id | date_last_modified | title | file_name | full_name
```

Examine the rows of output to determine which of the returned results is the document you need, from the title and the name of the file that was uploaded. Once you've determined this, look for the file in `/opt/collabnet/teamforge/var/deleted_files`.

```
yujr0C98B989011083DBFF4F534F | doc1001 | 2007-04-11 08:49:19-07 | example |
foo.txt | Joe User bxma0C98B98901115B9CE99E82C2 | doc1002 | 2007-04-11
08:49:19-07 |
example | test.txt | Joe User rpyf0C98B98901116DD1E40F9F8B | doc1003 |
2007-04-11
08:49:19-07 | example | findme.txt | Joe User ltrq0C98B9890111D76E74920A0E |
doc1004
| 2007-04-11 08:49:19-07 | example | project_logo.jpg | Joe User
```

Let's say the fourth document above was the one you wanted. You can find it at: `/opt/collabnet/teamforge/var/deleted_files/l/lt/ltr/ltrq/ltrq0C98B9890111D76E74920A0E`. Simply copy that file off the system, rename it to project_logo.jpg and return it to the user.

Why do I get a URL "not found" or "moved permanently" error after applying a patch/upgrade?

If you are experiencing a URL "NOT FOUND" or "MOVED PERMANENTLY" error after applying a patch or upgrade, then set Apache ProxyPreserveHost token to ON in the `httpd.conf` file.

If you have applied a patch or upgrade and are now receiving the following error:

```
<The document has moved <a href="https://www.<site>/sf/global/jsp/
buildtime.html"
    format="html" scope="external">here</a>.</p>
<hr> <address>Apache/2.2.3 (Red Hat) Server at www.<site>.com Port 80</
address> </body></html>
Not Found
```

```
The requested URL /sf/sfmain/do/userPicker/projects.pftool//sfmain/do/
listMonitoringUsers/projects.pftool/discussion.announcements was not found
 on this server
```

Or if you are trying to add users to a monitoring list, and are receiving the following error:

```
Not Found
The requested URL
    /sf/sfmain/do/userPicker/projects.pftool//sfmain/do/listMonitoringUsers/
projects.pftool/discussion.announcements
    was not found on this server.
```

Set the `ProxyPreserveHost` token to ON in the `httpd.conf` file.

Unable to embed Jenkins page into an iframe. How to fix it?

From the Jenkins version 1.532 and later, the Jenkins page is not displayed in an iframe.

Do you need a way to permit Jenkins to be visible in selected iframes? Here you go.

To configure X-Frame-Options:

If you try to load content into an iframe and the **X-Frame-Options** header is set to **SAMEORIGIN**, then the content loading is prevented. To override the default **X-Frame-Options** header settings, you have to perform the following:

- You need to have the Jenkins plugin installed to configure the **X-Frame-Options**. You can download the plugin from *https://wiki.jenkins-ci.org/display/JENKINS/XFrame+Filter+Plugin*.
- You must have Jenkins administrative privileges to do this. On the **Jenkins Configuration** page, in the **XFrame Filter Configuration** section, enter the value for **X-Frame-Options Options** as **ALLOW-FROM <ctf domain name>**.

    For example, on the `http://jenkins.maa.relenghub.collab.net/jenkins/configure` page, enter `ALLOW-FROM http://forge.collab.net` for **X-Frame-Options Options**.

    Now, the page can only be displayed in a frame on the specified origin.

The TeamForge-Binary integration is down after enabling SSL. What should I do?

TeamForge-Binary integration is found to be down soon after enabling SSL on sites which initially had SSL disabled. You must update the base URL, go URL and end point URLs stored in Postgres to access your Binary servers.

Run the following query with the psql-wrapper script (/opt/collabnet/teamforge/runtime/scripts/psql-wrapper) to update the base URL, go URL and end point URLs.

```
update integrated_application set base_url='https://<url>',
go_url='https://<url>', end_point='https://<url>' where name='Binaries';
```

☞ **Note:** In the above query, replace `<url>` with valid base URL, go URL and end point URL for your site. The URLs must use `https` as illustrated above.

Why do ETL jobs fail post TeamForge upgrade?

This could be due to incompatible database and JDBC driver versions.

Pentaho, used by TeamForge for data integration and transformation jobs, recommends using compatible JDBC drivers meant for specific database versions. See Pentaho's *JDBC Drivers Reference* for more information.

Do this if the ETL jobs fail post TeamForge upgrade:

1. Refer to Pentaho's *JDBC Drivers Reference* page.
2. Click the JDBC driver reference URL corresponding to your database, Oracle or PostgreSQL.
3. Identify and download the compatible JDBC driver for your database.
4. Replace the JDBC driver found in the following directories with the one you downloaded. (The TeamForge ETL process refers to the JDBC driver available in these directories.)

    - `/opt/collabnet/teamforge/dist/tomcat/commonlib/`
    - `/opt/collabnet/teamforge/runtime/tomcat_etl/webapps/etl/WEB-INF/lib`

☞ **Note:** You can also refer to *this page* for more information about Pentaho-special database issues and resolutions.

Why are statistics charts broken on a wiki page?

Statistics charts such as artifact statistics, task statistics, FRS statistics and document statistics, when added to a wiki page, do not get displayed because the relevant markup formats *[sf:artifactStatistics]*, *[sf:taskStatistics]*, *[sf:frsStatistics]* and *[sf:documentStatistics]* have been deprecated from TeamForge 8.1 and later.

So, if you try to create a wiki page with these values, instead of displaying the charts, the following error messages are displayed:

- For task charts, "The Task Statistics are discarded."
- For all the other statistics charts, "The Statistics charts are deprecated viewing from Wiki. It can be accessible from Project Home Page."

Manage security

Questions about managing security.

Concept definition.

Can I block project data from public search engines?

Yes, edit the robots.txt file to specify the pages or directories that should not be indexed by search engines.

If your site has some content that you may not want to be publicly searched for, or if some search engine hits are causing the site to slow down, you can make this change.

Only one robots.txt file can be created for a site. This file can contain a list of url patterns that should not be indexed, against the web crawler name.

The default robots.txt is available in the SITE_DIR/var folder. The robots.txt file is accessible without logging into CollabNet TeamForge, via the domain/robots.txt URL. You can update and commit the robots.txt file into the branding root directory, and the system then places the file in the SITE_DIR/var/overrides folder.

☞ **Tip:** You must have the commit permissions to create a robots.txt file under the branding repository in the Look Project.

How can I enforce strong passwords?

You can configure the application to reject passwords that do not meet your security criteria.

To enforce password requirements, place the following lines: :

```
system.password.min-length=5
password.requiresNumber=true
password.requiresNonAlphaNum=true
password.requiresMixedCase=true
```

into:

```
/opt/collabnet/teamforge/sourceforge_home/etc/
sourceforge_configation.properties
```

Once these lines are in place, restart TeamForge for them to take effect. The above example would require a password of at least 5 characters that must include at least one (1) mixed case letter, at least one (1) number, and at least one non-alphabetic character, e.g. Us3r!

NOTE: These settings apply only to new passwords. Anyone in the system currently will be able to continue to use their existing, potentially weak, password. You should force all users to change their passwords after changing these.

How do I configure Subversion to authenticate against multiple LDAP domains?

For some configurations, a Subversion server may need to be authenticated against multiple LDAP domains. This is possible by modifying the Apache configuration.

This is now possible due to the mod_authn_alias module for Apache. The external link for the module contains multiple usage scenarios. You will need to confirm that your Apache has been compiled with the module enabled. (This is the case for CollabNet Subversion binary packages since 1.5.4). If it is compiled as a module, make sure it is enabled via the LoadModule directive in your Apache configuration.

Example for configuration usage for authentication against three LDAP servers :

```
<AuthnProviderAlias ldap ldap-US>
    AuthLDAPBindDN cn=ldapuser,o=company
    AuthLDAPBindPassword password
    AuthLDAPURL ldap://ldap-us.company.local/ou=Developers,o=company?sub?
(objectClass=*)
</AuthnProviderAlias>

<AuthnProviderAlias ldap ldap-EU>
    AuthLDAPBindDN cn=ldapuser,o=company
    AuthLDAPBindPassword password
```

```
        AuthLDAPURL ldap://ldap-EU.company.local/ou=Developers,o=company?sub?
(objectClass=*)
</AuthnProviderAlias>

<AuthnProviderAlias ldap ldap-IN>
    AuthLDAPBindDN cn=ldapuser,o=company
    AuthLDAPBindPassword password
    AuthLDAPURL ldap://ldap-in.company.local/ou=Developers,o=company?sub?
(objectClass=*)
</AuthnProviderAlias>

<Location /svn>
    DAV svn
    SVNParentPath /opt/subversion/repos
    AuthType Basic
    AuthName "Subversion Repository"
    AuthBasicProvider ldap-US ldap-EU ldap-IN
    AuthzLDAPAuthoritative off
    Require valid-user
</Location>
```

How do I authenticate multiple LDAP via Apache?

If you need to add multiple OU= values in the LDAP url you must have separate LDAP urls and utilize AuthnProviderAlias to check both LDAP searches.

Use the following AuthnProviderAlias to check LDAP searches.

```
LoadModule authn_alias_module
modules/mod_authn_alias.so

<AuthnProviderAlias ldap ldap-alias1>
AuthLDAPBindDN cn=youruser,o=ctx
AuthLDAPBindPassword yourpassword
AuthLDAPURL ldap://ldap.host/o=ctx
</AuthnProviderAlias>

<AuthnProviderAlias ldap ldap-other-alias>
AuthLDAPBindDN cn=yourotheruser,o=dev
AuthLDAPBindPassword yourotherpassword
AuthLDAPURL ldap://other.ldap.host/o=dev?cn
</AuthnProviderAlias>

Alias /secure /webpages/secure
<Directory /webpages/secure>
Order deny,allow
Allow from all

AuthBasicProvider ldap-other-alias ldap-alias1

AuthType Basic
AuthName LDAP_Protected_Place
AuthzLDAPAuthoritative off
Require valid-user
</Directory>
```

After switching to ADS authentication, why did the Create button disappear from the user admin section?

When using external authentication such as LDAP, creating users from within the application is disabled. All users must be created via LDAP.

See *login-config.xml* for more information.

How does TeamForge authenticate CVS users?

CVS is treated as a special case when managed by a TeamForge site. It is not authenticated in the same way as SOAP API clients.

CVS relies on the Linux operating system to provide access and security. This includes permissions on individual repositories and access to the server itself. To add users, change passwords, create repositories, etc., the TeamForge integration simply changes the appropriate settings in the Linux operating system.

Users access CVS via an encrypted SSH session. To support this, TeamForge creates accounts on the Linux server that hosts the CVS repository. A typical CVS repository is created in the `/cvsroot` directory and is owned by root, with a group assigned by TeamForge. To gain access to a repository, TeamForge will add a user to the appropriate system group.

When TeamForge detects that a user's password has changed, it changes the password for that user on the Linux server too.

Users also have the option to use SSH keys or Kerberos tickets.

When a user is added to the Linux server, the login shell is `cvssh`, which limits their activities to CVS commands.

☞ **Important:** Do not expose cvspserver (the TCP protocol over port 2401) either internally or to the Internet if there is any way you can avoid it.

Password changes under internal authentication

To set the password for the user at the operating system level, TeamForge needs to have the clear-text version of the users password. The only time TeamForge has this is when the user uses the Change Password form in the Web UI. This is because the database-stored version, as an MD5 Password Hash, is a one-way encryption and can't be decrypted.

On a successful password update, TeamForge makes a SOAP call to the integration server that manages CVS. For this reason, the integration server must be SSL-enabled.

Password changes under external authentication

When a password change happens in an external authentication system, TeamForge does not immediately know that the password has changed. TeamForge needs a way to detect that the password has changed.

To accomplish this it keeps a copy of the last password the user successfully logged in with as an MD5 Password Hash in the same database table and field that it normally uses for Internal Authentication.

Now that TeamForge has a reference point, it still needs a clear-text copy of the password to make the change at the Linux operating system level. The only time this is available is when the user logs into TeamForge via the Web UI or SOAP API. So upon a successful login TeamForge compares the password to the encrypted one. If it is different it tells the Linux operating system to change the users password and then saves it in the database (as an MD5 Password hash).

Until the user logs into TeamForge, the CVS server will still have and accept the old password. There is no CVS server-side way to trigger a password update, unless an alternative method is used, such as LDAP or Kerberos.

Alternative Authentication

Because users use SSH to access a TeamForge-managed CVS server, it is possible to configure SSH to accept other authentication features such as SSH keys and Kerberos tickets. It is even possible to disable the use of passwords and require the use of other alternative methods.

- TeamForge supports SSH Keys natively. The user uploads their public key into their profile under "My Settings" in the TeamForge Web UI. The key will automatically be copied to each CVS server that TeamForge manages.
- If TeamForge is using External Authentication and the method is Kerberos, then SSH can be configured to use the same Kerberos server. This allows users to use Kerberos tickets for CVS operations.

LDAP

Linux supports LDAP as an authentication source. If TeamForge is using External Authentication and the source is LDAP, then SSH can also use that same source. When you do this, passwords and user account status are observed in real time instead of as a mirror of TeamForge.

Can the users be forced to change their passwords at first login?

Yes, as a site administrator you can configure the CollabNet TeamForge site options to force the users to change their passwords at first login.

Setting the `REQUIRE_USER_PASSWORD_CHANGE` attribute as `true` in the `site-options.conf` file enforces password change on first login into CollabNet TeamForge.

☞ **Tip:** You can not force password change on a user who had self-created the user account, or if a password-request had been raised for the user or if an administrator had reset the login password for that user.

Does TeamForge work with LDAP?

Yes, you can have your TeamForge installation authenticate against an LDAP server.

This is handy when users want to use a variety of different resources without having to maintain credentials for each one separately.

Overview

CollabNet TeamForge is a JBoss2 based application and relies on the JBoss JAAS service for user authentication. This enables a TeamForge site to authenticate users internally or externally.

Internal user authentication	Out of the box, TeamForge relies on its local database to manage user accounts. This includes username, password, full name, email address and a variety of other meta data values. Passwords are stored in the database using the standard MD5 Password hashing algorithm1. The database is only accessible by the application itself and a user with root access to the physical server. While running in this default configuration users are allowed to change their passwords in TeamForge, and any user with site administration privileges can create and approve new user accounts.
External user authentication	The JAAS service comes with several standard providers that allow TeamForge to be integrated with services such as LDAP, Active Directory and Kerberos. The JAAS service allows more than one source to be configured in the event several sources are needed.

☞ **Note:** It is possible to use both types of authentication with a single TeamForge installation. See your CollabNet representative for details.

To ensure that you are not locked out of your site, the site administrator account is always validated by TeamForge, not by LDAP.

LDAP accounts must conform to the TeamForge rules for user names and passwords. For example:

- If a password is used in LDAP that is shorter than the minimum allowable password length in TeamForge, you cannot create the user in TeamForge.
- A user name that starts with a special character, such as an underscore, will not be accepted by TeamForge, even if it is valid in LDAP.

(For detailed TeamForge user name and password rules, see *Create a new user account* on page 271.)

How is life different for the user under external authentication?

- When you turn external integration on, every user account (except the site administrator account) must have a matching LDAP entry to log in. This may require changing some existing accounts to match their corresponding LDAP records. (Accounts created after LDAP is in place are validated with the LDAP server when they are created, so you don't have to worry about this.)
- Every login attempt (Web UI and SOAP access) is passed to the external provider. This means that any changes to the user status in the external system take effect immediately. Users who have already logged in and have valid sessions are not affected.
- When TeamForge is using internal authentication, a site administrator can change a user's password. This is disabled for external authentication.
- Under external authentication, passwords can't be changed in the TeamForge web UI. Users have to use the interface provided by the third-party authentication source to change their password. Such password changes are available immediately to TeamForge for the next login attempt.
- Site administrators can no longer create user accounts. The end user must create their own account by logging into TeamForge just like a user who already has an account. At that point TeamForge detects that a new account needs to be created and presents the new user with a registration form, which requests the user's password n the external authentication system. On submit, TeamForge verifies the user account with the external system, and only if the username/password is verified does TeamForge create the new account.
- Once a new user has created their account, TeamForge can optionally be configured to put every new account in a pending status so that a site administrator can approve the new account. By default, new users will have immediate access to the system.

LDAP for source control

LDAP is integrated into your TeamForge source control services.

- For Subversion, the integration server queries TeamForge as needed.
- CVS authentication is not managed directly by LDAP, but each TeamForge user's SCM password is synchronized automatically with the user's LDAP password upon logging into TeamForge.

What can go wrong?

When TeamForge is configured to authenticate against an LDAP server and the LDAP server is down, all TeamForge authentication is disabled until the LDAP server is restored.

If a user does not exist on the LDAP server, or is deleted from the server, that user cannot log into TeamForge.

Why do I get the "Invalid command 'AuthLDAPAuthoritative'" error when I try to set LDAP for SVN users?

The invalid command `AuthLDAPAuthoritative` error may occur if you need to upgrade Apache from version 2.0 to 2.2.

CollabNet Subversion 1.5 is bundled with the latest version of Apache (currently 2.2.x). It includes the module mod_authnz_ldap and does not include mod_auth_ldap. Hence compatibility issues arise due to missing directives. Upgrade your Apache version to 2.2 if you get the following error when trying to install CollabNet SVN:

```
bash-3.00# /etc/init.d/collabnet_subversion start
   Starting CollabNet Subversion:
   Syntax error on line 29 of
      /etc/opt/CollabNet_Subversion/conf/collabnet_subversion_httpd.conf:
   Invalid command 'AuthLDAPAuthoritative',
      perhaps misspelled or defined by a module not included in the server
configuration
   FAILED
```

How does TeamForge handle multiple redundant LDAP servers?

When configuring LDAP authentication for a TeamForge instance, there may be a business need for using multiple LDAP servers.

Follow the guidelines below for configuring.

The additional LDAP servers can be added to the `java.naming.provider.url` option in `login-config.xml`:

```
login-config.xml:
        <module-option name="java.naming.provider.url">
        ldap://primary/ ldap://secondary/</module-option>
```

Once the primary and secondary servers have been defined, they will be consulted in order of definition for every authentication request. First the primary, and if the primary fails, then the secondary. This prevents specifying multiple servers for round-robin handling of authentication, but it can still be used for redundancy needs.

What user activities are tracked?

In case of a data security compromise, a record of who is performing what activities will help resolve some of the security issues.

Typically web servers log every page (or URL) being accessed, including the IP address of the user, date and time of access, etc. These logs are very useful in tracking the source of any security violations that may occur.

CollabNet TeamForge auditing tools are a powerful way to track unwanted and/or unauthorized changes within the system.

What is a CERT advisory?

CollabNet Product Support monitors the CERT coordination center (http://www.cert.org/) for notification of vulnerabilities or exploits against applications that CollabNet TeamForge provides.

If CollabNet Technical Support identifies an advisory that may indicate potential challenges for users who have deployed CollabNet TeamForge , Support proactively releases a notification and a statement of action. CollabNet will provide product updates as it deems appropriate or necessary.

J2EE Architecture and security

CollabNet TeamForge is a J2EE application that employs three-tier architecture to provide a secure environment for mission-critical data.

In a multi-tier architecture, access to each tier is restricted to the tier above it, effectively securing the tiers behind the firewall. For example, while clients (users accessing the system through a web) access the web server, they neither have access to the application and backend servers nor are they aware of their existence.

Similarly, the web server itself does not have access to the backend servers (database, SCM, mail etc.)

Exceptions to this rule include:

- Direct client access provided to the SCM servers. SCM servers are accessed across the firewall typically through SSH protocol (for CVS), or HTTP or HTTPS (for Subversion). SCM server data is also accessible in a view only mode through the web interface.
- Clients must have access to the mail server for posting messages to mailing lists.
- Mail server must have access to deliver messages across the firewall.

Clients can also access the SOAP APIs through the web server. The web server in turn forwards SOAP requests to the application server for processing.

Which ports are required to be open for a TeamForge site?

The components of a CollabNet TeamForge installation listen on a number of operating system ports.

A small subset must be exposed externally to enable users to access TeamForge services. Any port that is not absolutely needed must be closed.

⚠️ **Caution:** Expose only the JBOSS and Tomcat ports that are required for integration with another application, and open them only to that specific host IP address, even within your internal network.

You can select your open ports in one of these ways:

- Use the firewall configuration GUI tool that comes with your operating system. It's usually launched with a command like `system-config-selinux`.
- Open the `/etc/sysconfig/iptables` file and specify your open ports by hand.

Ports open to the Internet

Open the following operating system level ports. All other ports must be firewalled off to maintain security.

☞ **Important:** Do not open port 7080 or port 8080 to the Internet. These ports are only for communications between the TeamForge application and the source code integration service, when those two site components are running on separate boxes.

22 (SSH)	Port 22 is the default port for the secure shell (SSH). This is required for basic SSH administrative functionality and for CVS, as all CVS transactions occur over SSH. If all Teamforge repositories are in SVN (the default for Teamforge), then this port should be closed to the public and only accessible to the system administrators.
	If you have to expose SSH to the Internet, the best way to protect it is to require SSH keys and not allow password authentication, and do not permit root logins over SSH. If you must use local authentication for SSH, enforce regular password changes and password complexity.

☞ **Note:**

- If you have to expose SSH internally, limit access to the port to a bastion host if you can; otherwise limit it to specific trusted hosts or subnets.
- Do not expose cvspserver (the TCP protocol over port 2401) either internally or to the Internet if there is any way you can avoid it.

25 (SMTP)	Port 25 is the default port for SMTP (email). CollabNet TeamForge discussion forums include mailing list functionality that allows users to send email to the TeamForge server. The James mail server included with TeamForge listens on port 25 to accept this mail for processing.
80 (HTTP)	Port 80 is the default port for Web data transfer. We strongly recommend that you set up SSL and use port 80 only to redirect to port 443.
443 (https)	Port 443 is the default port for encrypted Web data transfer (HTTPS). The Apache web server should be configured to encrypt all data so that it cannot be compromised by a third party with malicious intent. Apache can be configured to force all traffic to be sent over HTTPS, even when a request is sent via port 80 (HTTP).

TeamForge can help you take care of this, if you tell it to. See *Set up SSL for your TeamForge site* on page 581 for details.

Ports for internal use only

Ports 7080 and 8080 have special internal uses for your site, but should not be exposed externally.

7080	On the source code integration server, if it is a separate physical server from the TeamForge application server, expose a port by which the application server can communicate with the SCM integration server. The default is port 7080.
8080	If you are running the source code (CVS or Subversion) integration server on a separate physical server from the TeamForge application server, set port 8080 on the TeamForge application box to accept connections from the server where your source code integration service is running.
8500	The default port for binary apps (such as Nexus OSS) integrated with TeamForge.

☞ **Important:** Do not open port 7080 or port 8080 to the Internet. These ports are only for communications between the TeamForge application and the source code integration service, when those two site components are running on separate boxes.

Open the *REPORTS_DATABASE_PORT* if you are granting direct access to the datamart from specific IPs using the REPORTS_DB_ACCESS_HOSTS `site-options.conf` token.

Ports to be open in the firewall environment for TeamForge 8.2

Name	Source Box	Target Box	Port	Notes
Apache	ALL	TeamForge App	80 or 443	443 for ssl
TeamForge Database	TeamForge App	TeamForge Database	5432	
SVN Integration	TeamForge App	SVN	80 or 443	443 for ssl
GIT Integration	TeamForge App	GIT	80 or 443	443 for ssl
GIT ssh	ALL	GIT	29418	
Indexer	TeamForge App	Indexer	2099	
Binaries	TeamForge App	Binaries	8500	
Reports DB	TeamForge App	Reports DB	5432 or 5632	5432 is used by default as Reports DB is co-hosted with TeamForge database. 5632 can be used if you want Reports DB on a separate port.
Reports ETL	TeamForge App	Reports ETL	7010	
Codesearch	ALL	Codesearch	9180	
EventQ	TeamForge App	EventQ App	8844	
EventQ Adapters	ALL	EventQ MQ	5672	

Name	Source Box	Target Box	Port	Notes
EventQ Messages	EventQ App	EventQ MQ	22, 2572, 15672	15672 is for admin
EventQ DB	EventQ App	EventQ DB	22, 27017	

> ☞ **Note:** No manual port configuration is required for the following services if they are all installed on the TeamForge App server:

- TeamForge Database
- SVN Integration
- Indexer
- Binaries
- Reports DB
- Reports ETL

Roles and permissions

Questions about setting permissions and using roles in TeamForge.

Concept definition.

Can I block a user's access to source code?

From time to time, you may need to prevent a user from using a source code repository.

SCM access can be withdrawn from a user by doing one of the following:

- Change the user's access to specific paths in the repository. (Subversion only.)
- Remove the role providing the desired source code permissions from the user.
- Remove the user from the project.

How this works depends in part on whether the repository is managed by TeamForge.

- If the source control server that hosts the repository is managed by TeamForge, the user's access is removed immediately.
- In the case of a CVS server that is not managed by TeamForge, all TeamForge administrators are sent an email requesting that the change be made manually. The request appears in the **SCM Access Requests** section of the **Integrations** page for approval.

Both the project administrator and the user receive email notifications when the change is made.

Can I assign a role to all users of the site at once?

If you are a site administrator in CollabNet TeamForge, you can assign a role to all the system users (non-site administrators) at one go.

Now, CollabNet TeamForge provides a system created user group called "All Users", including all site users. All role assignments that can be made to a user group can be made even for the "All Users" user group.

> ☞ **Note:** The "All Users" group members can not be manually added, edited, viewed or deleted. This group is automatically updated each time a user is added or deleted from the site.

Can I request a role?

If you are a project member in CollabNet TeamForge, you can request a role in your project.

You can use the **Project Home** > **My Roles** page to request for a role. Your request is submitted to the project administrator for approval. You will receive an email notification when your request is either approved or denied.

> ☞ **Note:** Based on your project administrator's discretion, some role requests may be granted immediately, while the other role requests may need approval.

Why don't I have access to the "Reported in Release" information in my artifact?

If you cannot see the "Reported in Release" information in your artifact, ask your project owner to add the role required to obtain access to this information.

Users with tracker artifact submit/view/edit permissions will not be able to see "Reported in Release" information in an artifact, as it is directly associated with the File Releases section.

You can request that the project owner add a role permission of View Only for All Packages in File releases for you to obtain access.

How do user permissions in CollabNet Enterprise Edition map to permissions in TeamForge 8.2?

Permissions in TeamForge 8.2 are different from permissions in CollabNet Enterprise Edition.

Permission mapping details

Care has been taken to map the permissions of a CollabNet Enterprise Edition role to an equivalent role in CTF. However, after migration it may be similar, but not same. See these details to get a better understanding of the mapping between CollabNet Enterprise Edition and TeamForge 8.2 permissions: *User permissions in CEE and CTF 8.2 projects* on page 320

☞ **Note:** Site-wide roles in TeamForge 8.2 are equivalents of Domain Roles in CollabNet Enterprise Edition. Global project roles in TeamForge 8.2 are the equivalents of Project roles with "All projects" visibility in CollabNet Enterprise Edition.

Exception reports

In cases where it is not possible to retain the permissions mapped to a role in CollabNet Enterprise Edition, the exceptions are reported and the exception report is provided for your reference in TeamForge 8.2. Each project will have an exception report with the change details at the project level. The exception report for the project is placed in the "Documents" tool under the `CollabNet Enterprise Edition_migration_exception_report` folder. The report file is named as `<project-name>-REPORT.txt`.

The Look project contains another exception report with site-level details. This would also have exception report of individual projects in the system. The site level report is placed in the look project under the same folder as `GLOBAL-REPORT-CONSOLIDATED.txt`.

Please go through the global exception report for the site, and see that you are okay with it, or make necessary adjustments as needed. To use the exception report effectively, see *How to read an exception report?*

Resource Patterns

Resource patterns associated with the CollabNet Enterprise Edition roles are not migrated. This means that the users belonging to these roles will have access to the complete repository after migration, depending on their access level in CollabNet Enterprise Edition. For example, if a role had "VersionControl - Modify" access to only "trunk/src", post-migration, the role will have commit access to the entire repository. Similarly, if the role in CollabNet Enterprise Edition had "VersionControl - Read" access to only "trunk/docs", post-migration, the role will have view access to the entire repository.

TeamForge 8.2 uses Path-based permissions (PBP) as an alternative. You can transfer the regular expressions to PBP to control the repository access. Setting PBPs is easier when compared to using the Resource Patterns; however, PBP has some limitations compared to CollabNet Enterprise Edition's regular expressions.

PBP limitations include:

1. PBPs are not inherited.
2. Site-wide and global project roles do not allow PBP associations with permissions.

See these details to get a better understanding of the mapping between CollabNet Enterprise Edition Resource Patterns and TeamForge 8.2 permissions: *TeamForge 8.2 Permissions for Resource Patterns* on page 332

The exception report contains the permission-resource pattern mapping that was dropped, against the affected role.

Role associations with **Host/*Domain users

If your project had any role associations with *Domain users and **Host user's user groups, they are not retained upon migration. At this point, TeamForge 8.2 doesn't have an alternate mechanism for this.

👉 **Note:** If any such associations are dropped, role association of default user groups in Domain and project level are captured in domain and project level exception reports respectively. The default user group's subscription to discussions are captured in respective project's exception reports.

Registered User role and permission of a logged in user

Permission of a CollabNet Enterprise Edition's Registered user role are mapped to Default Access Permissions of individual projects. The difference here is that, there is no central place that controls the access rights but it has the advantage that each project can specify different permissions.

In cases where the CollabNet Enterprise Edition permissions are not retained, they are exception reported. Please refer the global exception report for details.

Anonymous access

You may skip this section if guest access was not enabled in your CollabNet Enterprise Edition site.

There is no central control in TeamForge 8.2 to enable/disable guest access. Each project can specify what an anonymous user can access through the DAP of the project. Permissions of CollabNet Enterprise Edition's Anonymous Guest role are mapped to Default Access Permissions of individual projects. If guest access was enabled in your CollabNet Enterprise Edition site, please look at the global exception report to see whether there are permissions that changed during migration.

👉 **Note:** It is recommended that you try out this in stage before doing in production.

Who can access an application?

Application permissions help you minimize the need to create and assign many similar roles for individual users. Instead, you can permit or restrict access to individual applications within the project for whole classes of users.

Application permissions supplement role-based access control (RBAC.) For each application's concepts, documents, file releases, trackers, and discussion forums, you can assign permissions globally based on user type.

For example, if you know that you want all project members to be able to view and submit to all project trackers, you can specify this application permissions. You need to configure these settings only once. All current and future project members will be able to view and submit to all trackers without having the tracker view/submit permission assigned to them individually via a role.

Before you do this, you should have identified your project as private, gated community, or public. Configuring permissions is a finer-grained level of control that operates within this hierarchy of project types.

Some applications may be invisible to some users based on the roles you assign. If you give a user a role that does not grant access to a particular application, that user cannot see the button for that application in the Web interface. (However, if that user also has some other role that does grant access to that application, the user can see the application button.

👉 **Remember:** A user's license type also influences what the user can see and do on your site. A user's license type supersedes any role assignments.

- An ALM license enables the user who holds it to use the full range of TeamForge features: both the core source-code management tools and the extended application life cycle management functionality.
- An SCM license enables the user who holds it to use the core TeamForge source-code management tools.

Ask your site administrator how many licenses of each kind are available for your users.

Exceptions

- If a user has an SCM license, that user can see only the tools that support the core source control functions of the site. The Tracker, Documents, and Tasks tools are not visible.
- If a user has any level of permission on a tracker or a task folder, that user can see the **Reports** button.
- If a user has tracker administration permission on any tracker, or task administration in any task folder, that user can see the **Project Admin** button.

User classes

These are the classes of users to which you can assign application permissions:

User class	Description
All users with Role Permissions	Only project members with appropriate RBAC permissions.
All project members	All project members.
All project members and unrestricted users	All unrestricted users, whether or not they are project members, plus all project members.
All logged-in users	All restricted and unrestricted users (all logged-in users,) whether or not they are project members.
All users	All users, whether or not they are logged in or have CollabNet TeamForge accounts.

Restrictions by type of site

On some types of sites, you can't assign application permissions to certain classes of users. In such cases, you must use role-based access control (RBAC) permissions.

- On a private site, you cannot set application permissions for these classes of users:

  - All project members and unrestricted users
  - All logged-in users
  - All users
- On a "Gated Community" site, you cannot set application permissions for these classes of users:

  - All logged-in users
  - All users

Can I delete an item if I have 'Administer' permission?

No. You cannot delete an item if you have the administer permission.

The administer permission does not allow you to delete an item. You can create, edit, submit, and view items if you have the administer permission. For deletion, you must have the delete permission.

Can I disable creation of user accounts?

As of TeamForge 4.1 SP3, it is possible to disable the creation of new accounts by users so that only a 'site admin' can create new users.

To enable this mode of operations, simply add the following line to /opt/collabnet/teamforge/ sourceforge_home/etc/sourceforge_configuration.properties:

```
sf.disableUserSelfCreation=true
```

Once this line is in place, restart TeamForge for it to take effect.

Why do I see the project home page though I lack the necessary permission?

This has been set so that when users are provided access to a project, they do not see an empty page.

Irrespective of whether the view permission on project pages is checked, users will be able to see the contents in the project home page.

Why do you need the authentication and authorization plugin for Hudson?

The Authentication and Authorization plugin allows you to set up your Hudson installation to authenticate against a CollabNet server and specify access control for CollabNet users.

☞ **Note:** Authentication and authorization are independent actions. You could set up your Hudson installation to authenticate against a SourceForge or TeamForge site, but not use that CollabNet site for authorizing users. Authorization is available only with CollabNet TeamForge 5.2, but authentication is possible with earlier CollabNet SourceForge Enterprise versions as well.

Authentication

Authentication determines user names and passwords. You establish user credentials when you enable your Hudson site to use the CollabNet security realm.

Authorization

Authorization determines what users can do on the Hudson site.

Your Hudson server can be shared between many CollabNet projects, and you can grant TeamForge users permissions at the site level. You specify site-level permissions when you configure the site to use a CollabNet server for authorization.

A job on your Hudson server may be involved with more than one TeamForge project. For example, a job that builds software can pull in source code from multiple project repositories. For the purpose of authorization, however, each job is associated with one CollabNet project, and you can give users project-level roles.

When your Hudson site is set up to use CollabNet authorization, TeamForge project administrators can assign these roles to project members:

- Hudson Build/Cancel
- Hudson Configure
- Hudson Delete
- Hudson Promote
- Hudson Read

Users get the highest permissions they are entitled to. For example, if a user is part of a group that has administration permissions for the Hudson site, that supersedes any Hudson-related role the user might be assigned within a specific CollabNet project.

Can I control user access to an integrated application?

TeamForge can integrate the permissions scheme of a separate application into the TeamForge role-based access control system.

To look at how this works, we'll use the Pebble blogging tool as an example. Pebble is an application that you can quickly integrate with TeamForge.

Pebble brings with it a set of pre-determined roles that you can assign to project users. The roles are defined in the XML application configuration file.

Blog Reader	You can only read blogs and make comments, the comments are sent for moderation.
Blog contributor	You can add blog posts, but they will be sent for moderation.

Blog publisher	You can add blog posts, moderate comments and blog posts.
Blog owner	You can do all that a Blog publisher does as well as change the blog properties and security options.

Any site user with one or more of these roles can see the **Pebble Blog** button in their project toolbar. Clicking that button allows them to operate Pebble according to their access rights.

☞ **Note:** Site Administrators don't need any specific permissions; they have all permissions on all projects on the site.

How does inheritance work?

Site or project administrators can create hierarchical relationships between projects so that one project can inherit members, roles and permissions from a parent project.

When you define users, user groups and roles with specific permissions in one project, they can be inherited in one or more subprojects. This helps you avoid duplicating the effort of defining users, user groups and roles across projects.

You can still, if required, create roles specifically for this project and add direct members to any project. The direct members can be assigned inherited roles. The inherited members can be made direct members and/or assigned direct/inherited roles too.

While inheriting roles, only the permissions associated with all (top-level) folders are inherited.

In a subproject, you can select the inherited project members from the **Assigned to** lists or from the user picker, as the case may be. Inherited users may not be part of these lists if their role inheritance was prevented in the parent project.

☞ **Note:** At the time of role creation, you can choose to allow or disallow the role inheritance into private subprojects.

How does inheritance work for project groups?

Site or project administrators can create project groups and add member projects to the group, to manage several projects as a single unit.

Moreover, hierarchical relationships may exist between projects so that one project can inherit members, roles and permissions from its parent project.

Quite like a project can inherit roles and permissions from its parent project; projects can inherit only permissions via the project group.

Project Groups permissions when granted to users via site-wide roles, also allows the users to access project groups in all cases. Prevent inheritance option for roles does not apply in those cases.

The project in which a user gets permissions through role assignment in project groups appears on the **My Workspace** > **Projects** > **All Projects** page.

The roles created specifically for a project group are not made available as inherited roles in the group's member projects for assignment. Project group roles can only be assigned directly in the project group context. These roles are not requestable.

Can I create new user accounts as "unrestricted?"

You can make all new accounts be "unrestricted" by default instead of the more secure "restricted."

This is controlled by the *USER_ACCOUNT_RESTRICTED* variable in the `site-options.conf` file. Change that value to false and restart runtime.

☞ **Note:** This does not change any existing accounts. If someone was restricted before this flag was turned on, they remain restricted until the site admin edits their account.

Why can't Oracle connect to my TeamForge installation?

The simplest way to correct this is to overwrite the.jar included with TeamForge with the one from $ORACLE_HOME.

TeamForge uses the thick Oracle JDBC driver, which has two parts. One of these is provided by TeamForge, the other is in $ORACLE_HOME. If these two components are incompatible, TeamForge will be unable to make a connection to the database.

Follow these steps to overwrite the.jar included with TeamForge with the one from $ORACLE_HOME:

```
cp $ORACLE_HOME/jdbc/lib/ojdbc14.jar
        /opt/collabnet/teamforge/jboss/jboss-3.2.6/server/default/lib/
```

A restart of the application will be required to use the new.jar.

Why can't I move these artifacts into this planning folder?

To avoid confusion, you must observe a few basic rules when you assign artifacts to planning folders.

When an artifact in a planning folder has a child artifact, the child artifact can only be assigned to the same planning folder as the parent artifact, or a sub-folder of that folder. For example:

- Before you assign an artifact to a planning folder, make sure the artifact and all its children are assigned to a single planning folder and its sub-folders.
- Don't assign an artifact to a planning folder that is above the artifact's parent artifact in the planning folder hierarchy.
- Before promoting a planning folder to a higher level in the hierarchy, make sure its member artifacts (and its members' parent artifacts) are all assigned to the same planning folder or a sub-folder of it.

An artifact can only be assigned to a planning folder in its own project.

As a best practice, consider instead creating a matching artifact in this project and associate the two artifacts.

Before you assign an artifact to a planning folder, be sure you have permission to edit every artifact affected by the move.

Who can access a project?

You control access to your CollabNet TeamForge project by a combination of project settings, membership rules and user restrictions.

Who should I allow into my project?

To decide how to control access to your project, think about what the project is for and who will be using it.

Consider these elements:

- The project's access setting.

  A project can be public, private, or gated community.

  ☞ **Note:** The site administrator can change the default project access permissions.

- The project's membership.
- Each site user's user type.
- Each site user's license type.
- Each TeamForge user's user type.
- Any parent projects from which members, user groups or roles are inherited.
- Any subprojects that inherit members, user groups or roles from this project.

User type

Users can be restricted or unrestricted.

- Restricted users can access only public projects and projects of which they are members.
- Unrestricted users can access all projects except private projects of which they are not members.

License type

Users can have an ALM license or an SCM license.

- An ALM license enables the user who holds it to use the full range of TeamForge features: both the core source-code management tools and the extended application life cycle management functionality.
- An SCM license enables the user who holds it to use the core TeamForge source-code management tools.

License type supersedes user type. For example, if you give a user an SCM license, and then declare that user an unrestricted user, the user can see only the core source code management tools in any project they can access.

Project access setting

A project can be private, gated, or public.

- **Private** - Private projects can only be accessed by project members. Private projects do not appear on the **Home** page, in the **All Projects** list, or in search or reporting results to users who are not project members.

  Create a private project when you want to strictly limit project access.
- **Gated** - Gated community projects can be accessed by project members and by unrestricted users. As with private projects, gated projects are not visible to users who are not allowed to access them.

  Create a gated community project when you want to exclude restricted users, but do not need to exclude other, unrestricted users. For example, your organization might wish to designate all contractors or outsourced staff as restricted users. They will not be able to see any gated community projects, but all of your full-time, regular staff will have access.
- **Public** - Public projects can be accessed by all users.

  Create a public project when you have no need to restrict access.

  ☞ **Note:** Project access is not the same as project membership. Project access allows a user to see the project in the **All Projects** list, visit the project home page, and browse selected project data. A restricted user may be able to access a project without being a project member.

This table shows which user types can access projects with each project access setting.

Project access type	Project member (Unrestricted user)	Project member (Restricted user)	Non-project member (Unrestricted user)	Non-project member (Restricted user)
Private	Yes	Yes	No	No
Gated	Yes	Yes	Yes	No
Public	Yes	Yes	Yes	Yes

How do I give read-only anonymous access to cvs/svn repository?

Set the default access permissions of the project to public and allow Source Code to view permission for all or specific repositories to All users while restricting other permissions.

To give read-only anonymous access to cvs/svn repository within a project while still restricting write access, you could set the default access permissions of the project to public with Source Code view permission open to all users, while restricting other permissions to specific user classes.

The steps to set this up are as follows:

1. Go to 'Project Admin' tab in the required project.
2. Click Permissions in the left nav.
3. Click on Default Access Permissions tab.

4. Set Project Access Permissions as Public.
5. Under Application Permissions, choose All users from the drop-down for Source Code View permission.
6. For other application permissions, choose a user class based on your access requirement.

   Note: In Step 5 above, it is possible for you to choose whether you want to give View access to All users for all repositories or a specific repository.

Who can access source code?

Project administrators can give project members specific kinds of access to a whole Subversion repository or any path within that repository.

Overview

You can specify access permissions for users with a given role at the repository level, or at the path level within a repository.

- When you set a permission for a role on any directory in the repository, all directories and files under that directory get the same permission.
- When you set a permission on an individual file in the repository, there is no effect on the permissions assigned to paths above the level of that file. (A file is just a path that ends with a file name.)

How you use path-based permissions will depend on whether you view permissions primarily as a way to grant access or as a way to restrict access.

Full access, with exceptions	Give your company's own employees commit access to your whole source code base, while allowing developers from contracting firms to commit only to those parts of the code base that they are expected to work on.
No access, with exceptions	Assign all developers "No access" by default, then assign each type of developer access to certain directories and files according to their responsibilities.

☞ **Note:** You can control access to a path or to an individual file. This is different from normal Subversion checkout and commit operations, which are performed on directories but not individual files.

No Access

When you deny all access to a repository for a role, users with that role cannot see that the repository exists, except if:

- The role has "View and Commit" access to some directory within the repository. In this case, users with this role can see the directories that contain the directory they have access to.
- The user has another role that grants access to some part of the repository.

☞ **Note:** An individual user can have multiple roles. When two roles have permissions that conflict with each other, the role with the more expansive permissions takes precedence.

View Only

Users with a role that has "View Only" access to a path can browse the contents of the repository on the Web site or by connecting directly to the repository from a client, such as Tortoise, CollabNet Desktop for Eclipse, or CollabNet Desktop for Visual Studio.

View and Commit

Users with a role that has "View and Commit" permission to a path can browse and download code, and can also check code into the repository.

Wildcard-based access control and path-based permissions

You can create rules that use wilcards to control access to specific paths in a repository. See *Wildcard-based access control and path-based permissions in TeamForge* for more background information about wildcard and path based permissions.

Scenario 1

Controlling access to a specific path in a branch using `authz` rules can be very time consuming and inefficient. Assume you have three branches, `[/branches/foo/build]`, `[/branches/bar/build]`, and `[/branches/baz/build]`. If you create `authz` rules, you may have to write a separate rule for every branch in the repository, let alone the fact that you need to write such rules for a branch you may create in the future.

```
[/branches/foo/build]
@build = rw
@dev = r
```

```
[/branches/bar/build]
@build = rw
@dev = r
```

```
[/branches/baz/build]
@build = rw
@dev = r
```

Instead, you can write a rule using wildcards such as:

```
/branches/*/build
```

Later, you can create and assign roles to users such as "build engineers" and "developers" with "read-write" and "read-only" access permissions respectively.

Scenario 2

In this scenario, assume that there is a particular file or folder pattern that needs access control in a repository. For example, you may want to restrict all users but release managers from modifying `.iso` files. It is impossible to define such a rule using the `authz` file.

With TeamForge, you can write a rule that partially uses wildcards in file or folder names such as:

```
/trunk/build/*.iso
```

This rule applies to all files and folders that end with `.iso` under the path `/trunk/build`.

Scenario 3

In this scenario, assume that you want to control access to a particular folder no matter where the folder is in a branch. For example, you may want to control access to the "build" folder wherever it is in a repository. You can write a rule using wildcards such as:

```
/**/build
```

The "******" matches any number of folder levels in a repository. For example, this rule applies to the following paths:

```
/trunk/build
```

```
/branches/feature1/build
```

```
/trunk/external_module/build
```

```
/build
```

In addition to these scenarios, you can use wildcards to create rules that suit your requirement. A few examples of how you can create wildcard-based rules:

- `/**/*.iso` - Match any `.iso` file anywhere in a repository.
- `/branches/RB*` - Match any branch if the name starts with RB.
- `/branches/*/*.txt` - Match all `.txt` files one level under any branch.

Notes about path-based permissions

If two paths have different permissions, the permissions on the lower-level path take effect. For example, consider a role that has "No Access" set for the path `/branches/version3/users`, but has "View and Commit" access to `/branches/version3/users/vijay`. A user with this role can:

- Check code in and out of the `vijay` directory.
- Click down through all the directories in that path, including `users`. (Directories that are not included in the user's permissions are not shown.)

Users with this role cannot:

- Check files in and out of the `users` directory.
- Monitor commits to `users`.
- Execute Subversion copy, move or delete operations that involve resources in the `users` directory (or any other paths where the user has View Only or No Access).

Commit messages

When users monitor a repository, they receive commit announcements by email that include the resources that their role permits them to see.

Users with a "No Access" role on a repository cannot monitor that repository to receive commit messages by email.

By default, commit emails provide all the details about the commit that a logged-in user can view in the **Source Code** application. A repository owner can elect to have the notification emails omit most of the detail and provide only the commit ID and the committer's user name.

Toolbar button

If none of the available permissions (**View Only**, **View and Commit**, or **Path-based Permissions**) is selected for any repository, and none of the options under **Source Code Permissions** is selected, users with this role do not see the **Source Code** toolbar button.

Can I ensure that only site admins can create projects?

You can do this using Velocity.

Create `/opt/collabnet/teamforge/sourceforge_home/templates/body_footer.vm` and populate it with this:

```
#set($superUser = ${PAGE_INFO.isSuperUser()})
#if($superUser == false)
<script>
```

```
if(document.getElementById("createProject") != null){
var e = document.getElementById("createProject").parentNode.parentNode;
while (e.firstchild) {
e.removeChild(e.firstChild);
}
}
</script>
#end
```

Save the file, and TeamForge will start hiding the button for non-site admins immediately.

How do user roles work?

Project administrators can define specific access permissions for individual project members. They do this by using global project roles and/or creating roles and assigning the roles to project members.

Under role-based access control (RBAC), permissions are not assigned directly to an individual user. Instead, each user has the permissions that are attached to any role that is assigned to that user.

A project member can be assigned multiple roles.

In CollabNet TeamForge, site or project administrators assign roles to the site users or project members. Besides this, a project member can submit a role request to the project administrator. The project administrator can approve or reject such requests.

When you define users, user groups and roles with specific permissions in one project, they can be inherited in one or more subprojects. This helps you avoid duplicating the effort of defining users, user groups and roles across projects.

☞ **Note:** Permissions are cumulative. If a project member is assigned a role that provides a specific permission, and another role that does not, the user has that permission.

A role defines these things:

- The applications that project members with that role can and cannot access.
- The resources on which project members with that role can use the applications.
- The actions that project members can take in each application and on each resource.

☞ **Note:** If a user has an SCM license, that user can see only the tools that support the core source control functions of the site, even if the user has a role that would otherwise grant access to other resources.

When a user's roles do not include access to an application or resource, that application or resource is not visible to that user. For example, imagine that you are assigning roles to Jason, a software developer. Jason needs to check source code in and out in order to fix bugs, develop features and create software releases. However, Jason does not need access to the wiki. If you set up Jason's roles according to those requirements, Jason's experience is like this:

- On any page in the project site, Jason can see and click the **Trackers**, **Source Code**, and **File Releases** buttons along the top of his screen.
- Jason does not see the **Wiki** button.
- If someone sends Jason a link to a page in the Wiki application and Jason clicks the link, he gets an error message. (The message does not specify whether the page exists or not.)
- When he accesses the project directly from Eclipse or Visual Studio, Jason can expand the project node and browse the **Trackers**, **Source Code**, and **File Releases** nodes, but not the **Wiki** node.

☞ **Remember:** A user's license type also influences what the user can see and do on your site. A user's license type supersedes any role assignments.

- An ALM license enables the user who holds it to use the full range of TeamForge features: both the core source-code management tools and the extended application life cycle management functionality.
- An SCM license enables the user who holds it to use the core TeamForge source-code management tools.

Ask your site administrator how many licenses of each kind are available for your users.

Applications

An application is a collection of related features designed to enable a user to collaborate on particular kinds of tasks. For example, the Documents application helps users create documents, share in document reviews, and publish documents, among other things.

In the Web interface, each application is represented by a button in the navigation bar at the top of any project page. A given user can see the buttons corresponding to applications they have access to by virtue of the roles assigned to them.

Applications are also known as "tools."

Resources

- The tracker application might contain a bugs tracker and a feature request tracker. These are the tracker resources.
- A project can contain multiple SCM repositories. These are the SCM resources.

Permissions

View	Allows users to view and download items, but not to create or edit items, administer folders, or edit application settings.
Create or Submit	Allows users to create new items, but not to edit items, administer folders, or edit application settings. Users with the create or submit permission also have the view permission.
Edit	Allows users to edit items, but not to administer folders or edit application settings. Users with the edit permission also have the view permissions.
Administer	Allows users to create and edit items, administer folders, and edit application settings. Users with the administer permission also have the edit, create or submit, and view permissions. To delete items, the user needs to have the delete permission.
Delete	Allows users to delete items, but not to administer folders or edit application settings. Users with the delete permission also have view permissions. Without the delete permission, users with the administer permission are not allowed to delete items.

How does CollabNet TeamForge help protect data access?

Access to data must be strictly controlled to meet the security requirements of the enterprise. Strict data access control is achieved through a combination of firewalls, authentication, and authorization.

Firewalls and network configuration

A firewall provides the first level of protection by restricting access to the private network from the Internet. Sophisticated firewall configuration can provide strong security for all enterprise resources.

All CollabNet TeamForge application server nor the backend servers should ever be exposed to the Internet.

The CollabNet TeamForge application to function effectively, the following conditions must be met.

- Across the firewall, clients (users) must have access to:

- • The web server through a secure protocol such as HTTPS (port 443). The web server typically handles both the browser requests as well as the SOAP requests and forwards them to the CollabNet TeamForge application server.
  - • Send mail to CollabNet TeamForge mail server via SMTP (port 25).
  - • The SCM server through a secure protocol such as SSH (port 22).
- • The web server must have access to the application server (typically port 8080).

  ☞ **Note:** This port is not exposed outside the firewall.

- • The web server must have access to the SCM server for repository browsing functionality.
- • The application server must have access to the backend (SCM, database, mail, etc.) servers.
- • The SCM server must be able to access CollabNet TeamForge for commit notifications.
- • The mail server must be able to deliver messages across the firewall.

Authentication and authorization

To secure sensitive data, CollabNet TeamForge provides access control tools to restrict unauthenticated and non-member access.

User authentication is supported through verification of username and password during login. Project administrators can completely restrict access to authenticated members by marking projects as gated communities or private. A gated community is only accessible to unrestricted users, while a private project is only accessible to its members.

How does CollabNet TeamForge help protect my data?

Sensitive data must be protected from illegal access at various points in the system. Key areas where security is typically compromised include data transmission and data storage.

Data transmission

Network traffic is not encrypted by default. The HTTP protocol (non-SSL) does not protect data during transmission. HTTPS provides Strong Encryption using the Secure Socket Layer and Transport Layer Security protocols (SSL/TLS).

☞ **Note:** The web server employed by a CollabNet TeamForge installation must be reconfigured to employ the HTTPS protocol.

Data storage

Sensitive data, such as credit card numbers, financial information, etc., must be stored securely. Usually this is done by encryption. In the context of an application like CollabNet TeamForge only stores password digests with an MD5 based cryptographic hash to guarantee adequate data protection.

MD5 is a one-way hash function that is used to verify data integrity through the creation of a 128-bit digest from data input. A one-way hash function is designed in such a way that it is hard to reverse the process, that is, to find a string that hashes to a given value. MD5 is currently a standard, Internet Engineering Task Force (IETF) Request for Comments (RFC) 1321. According to the standard, it is "computationally infeasible" that any two messages that have been input to the MD5 algorithm could have as the output the same message digest, or that a false message could be created through apprehension of the message digest.

How do site administrator roles work?

Site administrators are default site managers who can create additional site administrators and delegate few site administrative tasks to them.

They can also allow some CollabNet TeamForge users to use one or more CollabNet TeamForge tools across several projects, by creating site-wide roles with specific project permissions, minus site administrative permissions. They can also provide ready-to-use roles as global project roles, creating uniformity across the site.

In CollabNet TeamForge, site or project administrators assign roles to the site users or project members. Besides this, a project member can submit a role request to the project administrator. The project administrator can approve or reject such requests.

A role defines these things:

- The applications that project members with that role can and cannot access.
- The resources on which project members with that role can use the applications.
- The actions that project members can take in each application and on each resource.

When a user's roles do not include access to an application or resource, that application or resource is not visible to that user. For example, imagine that you are assigning roles to Jason, a software developer. Jason needs to check source code in and out in order to fix bugs, develop features and create software releases. However, Jason does not need access to project wiki. If you set up Jason's roles according to those requirements, Jason's experience is like this:

- On any page in the project site, Jason can see and click the **Trackers**, **Source Code**, and **File Releases** buttons along the top of his screen.
- Jason does not see the **Wiki** button.
- If someone sends Jason a link to a page in the Wiki application and Jason clicks the link, he gets an error message. (The message does not specify whether the page exists or not.)
- When he accesses the project directly from Eclipse or Visual Studio, Jason can expand the project node and browse the **Trackers**, **Source Code**, and **File Releases** nodes, but not the **Wiki** node.

Applications

An application is a collection of related features designed to enable a user to perform tasks and collaborate with other users. For example, the Documents application helps users create documents, share in document reviews, and publish documents, among other things.

In the Web interface, each application is represented by a button in the navigation bar at the top of any project page. A given user can see the buttons corresponding to applications they have access to by virtue of the roles assigned to them.

Applications are also known as "tools."

Resources

- The tracker application might contain a bugs tracker and a feature request tracker. These are the tracker resources.
- A project can contain multiple SCM repositories. These are the SCM resources.

Site Administration Responsibilities

The additional site administrators can be granted administrative rights for any of the site administrator responsibilities related to the following:

- Projects (includes project templates)
- Project Groups
- Users
- Groups
- Roles
- Categories
- System Tools
- Integrated Applications

Site Administration Permissions

View Only	Allows users to view and download items, but not to create or edit items, administer or edit application settings.

Create or Submit	Allows users to create or submit and edit items, but not to administer or edit application settings. Users with the create or submit permission also have the edit and view permissions.
Edit	Allows users to edit items, but not to administer items or edit application settings. Users with the edit permission also have the create / submit and view permissions.
Administer	Allows users to create, edit and administer items plus edit application settings, if required. Users with the administer permission also have the edit, create or submit, and view permissions. To delete items, the user needs to have the delete permission.
Delete	Allows users to delete items, but not to administer items or edit application settings. Users with the delete permission also have view permissions. Without the delete permission, users with the administer permission are not allowed to delete items.

Which role is assigned to me?

There are several ways in which you could have been assigned certain roles in CollabNet TeamForge. Your access to the projects depends largely on the permissions granted to you via your roles.

You could have been granted access to any CollabNet TeamForge project in any of the following ways:

* Through site-wide roles (assigned directly to the user)
* Through global project roles (assigned directly or inherited via another project, either individually or as a group member)
* Through project roles (assigned directly or inherited via another project, either individually or as a group member)
* Through permission inheritance (via project hierarchy)

There are two ways you can check which roles are assigned to you in any of your projects:

☞ **Note:** If you are a site administrator, you can view any user's access rights to the system.

To view the roles assigned directly to you:

Use the **My Workspace** > **My Page** > **Projects** page and click the **My Projects** tab.

☞ **Note:** You can click the role to view the folder-level permissions assigned to you in a project.

To view all the roles assigned to you:

Use the **My Workspace** > **My Page** > **My Settings** page and click the **Roles** tab.

You can select **Roles Created For a Project**, **Roles Inherited From a Parent Project** or **Site-wide roles** to view the corresponding roles assigned to you.

☞ **Note:** A global project role that has been assigned to you in a project where you are a direct member will be listed as a directly assigned role.

☞ **Note:** You can click the role to view the folder-level permissions assigned to you.

Who can access project planning information?

A project manager should consider carefully who is able to change the project plan.

You can specify the people who can change the product plan based on their roles in the project. Only users with the role you select can view, create, modify, and delete artifacts in the planning folder hierarchy, and populate those artifacts with content.

Every project must strike its own unique balance between openness and control. These broad guidelines may help you decide how to set up the appropriate roles for your project.

- Err on the side of visibility. Most of the time you will want all project members to at least see the contents of all planning folders. This helps developers avoid duplicating their efforts, and it can enable team members to volunteer their help to each other when appropriate.

  However, occasionally you may want to hide all or part of the planning folder hierarchy from selected people. For example, when you bring on a contract developer for a short period, you may want to restrict that person's view to the specific artifacts they are working on directly.

- Limit the number of people who can modify or populate planning folders. In many cases, only the project manager really needs to do this.

  However, it may make sense to allow some people with a direct stake in the project to modify some folders. For example, if you are working with a product owner, it might be useful for that person to be assigning artifacts to future planning folders while your team is working on the current one.

Who can see a project page?

A user can see a given project page if the page is not hidden and the user's role includes permission to see the page.

☞ **Note:** When a page is hidden, it is only visible to users with the Project Admin role, and then only when the user has clicked **Configure: On**.

If a person can see your project, they can see the project home page and any of its subpages. To see any other page, they must be assigned permission explicitly.

Permissions determine two kinds of access:

- Anyone with access to the project can see the project homepage.
- Users whose role includes permission to see a given top-level project page can see all the subpages of that page.

Subpages inherit their permission setting from the top level page they belong to.

Who can see a project page component?

To see a given project page component, you must have permission to see both the project page where the component is and the tool that the component represents.

Your ability to see and edit the contents of a project page component is controlled by your permissions with regard to the tool the component represents.

For example:

- If you have permission to create documents in a selected folder in the Document Manager tool, you have that same permission when viewing that folder in a Documents component on a project page.
- If you do not have permission to view a folder in the Document Manager tool, you cannot see a document component that contains that folder on a project page.

☞ **Note:** Users with the Project Admin role can see project page components even if they do not have access to the related tool.

Who is an owner of the sub component?

The user who creates the sub component is the owner of the sub component.

As a project admin, why don't I have permissions to the wiki?

Most likely, your site was installed before TeamForge contained the wiki component.

When the wiki was added to TeamForge, it was decided that there was no way for us to know the security requirements at customer sites, so permissions for the new wiki component were not assigned to project admins by default. As a project admin, you can alter any existing role to grant this permission, or create a new role for this permission and then assign to the appropriate project members as needed.

Concepts and terms in TeamForge

Descriptions of concepts and terms in TeamForge.

Concept definition.

What is an association?

TeamForge allows users to easily associate, or link together, any objects in the system to simplify knowledge sharing and provide traceability throughout the lifecycle.

For example, a discussion post regarding a customer problem could be linked to a document specifying the feature requirement. An issue might be created to track the defect, the source code commits that fix the issue, and the release that contains the fix. All of these records can be linked together with an association. When any item is associated with another, the link appears on the item's **ASSOCIATIONS** tab.

Associations enable development organizations to improve information sharing, capture institutional knowledge, and simplify regulatory compliance.

What is change management?

Change management is about monitoring and addressing work items, product defects, feature requirements, and support requests.

What is a document in CollabNet TeamForge?

For TeamForge purposes, a document is any file that you track with a document artifact.

Any project involves some number of documents that have to be written, edited, reviewed and published. You can use a document artifact to facilitate and standardize the work you do with documents. TeamForge automatically maintains the version history of each document, including the person who posted each version, its status, and any version comments.

If you are the author, use the Review feature to track the process of getting others to evaluate your document.

To make sure everyone who needs to know about a document knows about it, associate it with a task, a code commit, or another kind of artifact.

What is a feature tree?

A feature tree is a way to look at planned product features hierarchically, so you can quickly understand the relationships among product features you have specified.

A feature tree is what you see when you click the **Tree** button on a tracker overview screen. (You can return to the traditional single-level tracker view by clicking **list**.)

The feature tree shows you the nested descendants of an artifact at whatever level of detail you need, all the way down to the tasks supporting your lowest-level user stories. When you define a user story that depends on several tasks for its completion, and you declare each of those tasks a "child" of the user story, they appear as subordinate nodes under the "parent" user story artifact in the feature tree.

User stories are the most common components of a feature tree, but any kind of artifact can appear. For example, when the completion of a user story depends on fixing a bug, the artifact representing the bug appears in your feature tree as a "child" node under the artifact representing the user story.

For each artifact in the feature tree, look for this key information:

Priority	A relative measure of how important the feature is, compared with the other features you want in the product.
Artifact ID	The permanent alphanumeric identifier for the artifact. Use the **Jump to ID** box to quickly find artifacts by their artifact ID.
Title	A short phrase identifying the work that the artifact describes.
Status	The artifact's position in the work flow of its project. Standard status values are "Open," "Under development," "In QA," and the like.
Estimated Effort	A measure of the work required to deliver the work described in the artifact. The units used are up to you -- just make sure they are used consistently across the project, so that you get accurate totals when you need them.
Remaining Effort	A measure of the work that's still to be done on the artifact, in the same units used in **Estimated Effort**. Project members must update this value whenever they advance the work in the artifact.
Planning Folder	The planning folder that the artifact is assigned to, if any.

What is a discussion forum?

TeamForge discussions provide workspaces where project members can work together online or by email.

Discussion forums and mailing lists are closely integrated. Forum administrators can choose to enable a mailing list for each project forum. A mailing list extends the discussion forum functionality to allow project members to post messages to the forum using email.

Discussion forums can be public or private, depending on the forum's objective and desired level of access into the forum. Private discussion forums restrict anyone without specific access permissions from viewing or posting into the forum.

In a moderated discussion, messages from anyone except "trusted" users are screened by a moderator before they are posted. Messages posted by trusted users do not require the moderator's approval.

It's a good idea to use a discussion forum instead of direct personal email whenever privacy or security concerns don't prevent it, even when the communication only involves two project members.

*   Members not directly involved often can make unexpected contributions if they are aware of the discussion.
*   TeamForge archives all news items, forum posts, and mailing list communications, so you can go back and find valuable information later.

What is an integrated application?

An integrated application is a stand-alone application that can seamlessly integrate into any TeamForge project.

You can use integrated applications to incorporate these types of applications into your TeamForge project:

*   Third party applications
*   Internally developed applications
*   Integrations developed using the TeamForge SOAP APIs
*   External websites

When you add an integrated application to your project, an icon is added to your project navigation bar. Clicking the icon displays the integrated application in the main TeamForge project window.

TeamForge site-administrators can register site-wide integrated applications that project administrators can opt to use across projects.

Site administrators or users with site-wide roles with the administration permissions for integrated applications can enable/disable integrated applications.

☞ **Tip:** Disabling an integrated application restricts it from being added to projects. However, disabling an integrated application does not affect the projects where the integrated application might already be in use.

After your site administrator registers an integrated application on the site level, on adding it to your project, an icon is added to your project navigation bar. Clicking the icon displays the integrated application in the main TeamForge project window.

☞ **Note:** You can register and integrate as many applications per project as you wish. However, because each integrated application adds an icon to the project navigation bar, creating a large number of integrated applications can cause horizontal scrolling.

What is an iteration?

An iteration (also called a sprint) is the period of time it takes to produce something that works well enough to show to the user.

At the end of an iteration, users get to try out the functionality. They can and accept it or send it back for more work.

- Acceptance means the user agrees that the acceptance criteria stated in the user story have been met, and the capabilities as implemented support the needs that the user identified.
- Rejection means the product, as implemented, does not support the user's requirements. Sometimes this means the development team has not understood the user story the way the product owner meant it. Sometimes it means the user's initial ideas turned out not to be so helpful in practice. In either case, this leads to more analysis and development work.

Keeping iterations as short as possible helps reduce the time spent on corrections when features don't meet the users' expectations. It is less costly to redo two weeks of development than six or eight weeks.

Shorter, more modest iterations also tend to generate more reliable burndown data for estimating further work.

What is a linked application?

A linked application is an external application or site that users can get to from inside a TeamForge project.

You can use linked applications to incorporate these types of applications into your TeamForge project:

- Third party applications
- Internally developed applications
- Integrations developed using the TeamForge SOAP APIs
- Company intranet sites
- External websites

After you create a linked application, a button is added to your project navigation bar. Clicking the button displays the linked application in the main TeamForge project window (or in the case of a site-wide linked application, in a separate window).

☞ **Note:** You can create as many linked applications per project as you wish. However, because each linked application adds a button to the project navigation bar, creating a large number of linked applications can cause horizontal scrolling.

TeamForge administrators can also create site-wide linked applications that appear in all TeamForge projects.

What is the look project?

The `look` project contains special files that can override your site's default appearance and content, such as the default icons, fonts, colors, and labels.

Unlike most projects, the `look` project has no members. It is only visible to users with site administration permission. Its only purpose is to control your site's look and feel, including such things as fonts, background colors, icons, and the wording of the onscreen labels that appear throughout your site.

Any project on your TeamForge site can have one or more Subversion repositories associated with it. The `look` project has just one Subversion repository. That repository is named `branding`.

When a user requests a page from your site, TeamForge checks the `branding` repository to see if any files there specify custom fonts, colors or text strings. If such specifications are found, TeamForge displays the page according to those specifications. If not, the page displays according to the default design.

Having your custom look-and-feel specifications in a Subversion repository enables you to roll back changes, track contributions, and use all the other features of a source code versioning system.

What is a parent project?

A parent project is the base from which a subproject's members, user groups and roles, with their corresponding permissions, are derived. A subproject can inherit project members, user groups and roles from its parent project.

You can create a parent project to track and manage several smaller assignments as subprojects.

When you define users and roles with specific permissions in a project, those users and roles are passed down to any subprojects that belong to that project. This helps you avoid the repeated effort of defining users, user groups and roles across projects.

What is a patch?

A patch is a package of code that fixes or adds to the functionality of a CollabNet product. Patches are also known as "component upgrades."

Things to know about patches

- Patches are cumulative. You don't need to apply multiple patches sequentially to get to the desired patch level. You can move up (or down) one or more patch levels with a single operation.
- The Level option (-l) allows you to downgrade or upgrade to any patch level (within the maximum available in the cumulative patch).
- The Rollback option (-r) allows you to revert the site to the previous patch level it was at, before the current patch was applied.
- The Uninstall option (-u) allows you to downgrade the patch level on the site by one.
- When a patch installation fails you can use the Force option (-F) to proceed, without manually uninstalling previous patches.
- The system displays a summary of what happens during the patch installation.
- Before proceeding with the patch installation, you can use the "dry run" mode (-t option) to see the summary of actions that will be performed during the installation.

Best practices

Before applying a patch, note the following principles.

- The upgrade scripts are usable only with an existing installation.
- No data migration will occur if any changes have been made to the database schema.
- You must use the `sudo` command or have an account that is equivalent to root in order to complete a patch installation successfully.

👉 **Important:** Before installing a patch, verify that it has been fully tested and qualified.

What is a planning folder?

A planning folder is a virtual container that helps you organize and plan the work that goes into delivering a product.

You can create a hierarchy of planning folders to organize artifacts by product, release, iteration, etc. You can store artifacts from multiple trackers in a planning folder. This allows you to plan various stages of your project (i.e. releases, iterations, etc.).

Selecting an individual planning folder provides a view of all artifacts from all trackers within the selected planning folder.

The **Planned For** field identifies which product, release, or iteration the artifact is planned for, based on the planning folder that it is assigned to.

For example, in an agile development environment, a project manager breaks down the prospective product into its component parts and looks at what it would take to deliver each one. When all the parts planned for a given iteration are finished, the product is considered complete for that iteration.

Some parts of a product can be developed more or less in isolation, but most depend on other parts. Tracking these relationships is one of the trickiest aspects of product development.

For example, you can only provide a graphical user interface for a shopping cart application if you also come up with a database for the customer's payment data to be stored and accessed. That in turn requires a data storage and backup solution of some kind. And so on.

Use a planning folder to track the dependencies among the parts of your project as each moves toward completion. As you work through the question of what depends on what, you'll move artifacts representing user stories into the appropriate iteration in the appropriate release. As you proceed, you'll find a pattern like this emerging:

- Product 1
    - Release 1
        - Iteration 1
        - Iteration 2
    - Release 2
- Product 2

☞ **Note:** "Move" is meant figuratively. When you "move" an artifact into a planning folder, it is still a member of the tracker where it lives, and you can still do all the things with it that you can do with an ordinary tracker artifact.

Your planning folder lets you see at a glance the pieces of the work that support other pieces, and the pieces that depend on other pieces. Think of this as a "planning tree." If you are responsible for a development project, you can use this tree view to understand and predict the time and effort required to deliver a given set of features.

What are Planning, Task and Kanban boards?

When you have set up your planning folders and teams, you have four views available to work with them: **List**, **Plan**, **Track** and **Kanban**.

TeamForge user roles and permissions that are in place for planning folders apply to all the four views.

Planning Board

The Planning Board is an important tool for your TeamForge project's Agile planning activities. It enables you to plan and monitor the features that are required in each sprint (or iteration), and assign them from the product backlog to specific sprints. The planning board view complements the list view. While the latter offers you capabilities to accomplish various actions such as create, edit, and delete artifacts, planning folders and teams, the former offers product owners (or similar users) the ability to view, rank and move artifacts across the three planning folders (swimlanes) in a physical board-like user interface. In the Planning Board, planning folders are represented as swimlanes. In each swimlane, the tracker artifacts for the selected planning folders are represented as cards. You can also have a team's view of artifacts (backlogs and tasks), which is a swimlane representation of artifact cards for the selected team in the selected planning folder.

Task Board

The Task Board is an important tool in the Agile process. It helps the team to focus on the work at hand in the current sprint and feed progress data back into the system. Unlike the list view and planning board view, which can be used for agile project planning, the task board view is for tracking tasks in a sprint. TeamForge project administrators can configure the Task Board. Once that's done, projects members can use the Task Board to break down the stories into tasks and then progress the tasks, and the story, towards completion.

The Task Board can have at least two and up to seven swimlanes depending on how your project administrator configures it. Every swimlane in the Task Board represents a task status. Once configured, team members can use the Task Board to view tasks in a selected planning folder or filter tasks for a specific team within the selected planning folder, add new tasks for a backlog item (epic, story, etc) and move tasks from one swimlane to the other as tasks progress from one status to the other.

Kanban Board

The Kanban Board is an agile project management tool, which gives you a snapshot of the statuses of work items within a planning folder, how your project teams are placed in terms of work distribution and directs you to re-distribute the tasks to ensure optimal resource utilization.

Kanban Board uses the kanban method for project implementation. Kanban is built on two basic concepts: value stream mapping and WIP (work-in-progress) limits.

- **Value stream map**: A value stream, as defined in the *APICS (American Production and Inventory Control Society) dictionary*, comprises "the processes of creating, producing, and delivering a good or service to the market". In software development, a value stream map is an end-to-end mapping of the flow of activities (tasks/work items) from one state to another, starting from conceptualization to delivering the product to the customer.
- **WIP limits**: These are constraints (minimum and maximum) applied on each point or state (Planning, In Progress, etc.,) in the value stream to ensure optimal WIP. This defines the minimum and maximum artifact count that ought to be present in each state so that if these constraints are violated, Kanban Board flags the issue for you to fix the bottleneck.

  For example, let us assume that you want to view the status of the work items for a planning folder called 'Iteration 1' and assess the distribution of work between teams, Team A and Team B within this planning folder. You want to see a maximum of 8 artifacts in the 'In Progress' state and you configure your Kanban Board accordingly. When you go to your Kanban Board and view the artifacts for the selected planning folder, you see a constraint violation because there are 14 artifacts in this status. In addition, when you see the work split-up between the teams, Team A has 14 artifacts whereas Team B has none. This is clearly an indication that not only is the overall count of artifacts more than the maximum constraint, but even between the teams, Team A is overloaded and Team B is underutilized; this calls for re-distribution of work items to avoid any delay in the development process.

You can configure the Kanban Board based on your project activities and you can configure as many boards as you require.

List, Plan, Track and Kanban views

You can click the **LIST**, **PLAN, TRACK** and **KANBAN** buttons to toggle between these views.

When you make a planning folder or team selection in the respective list view, (either from the planning folder/team tree or through the **Jump to ID** box), the selection is retained when you move to the task board or kanban board view. Similarly, when you select a planning folder or team in the task board or kanban board view, the selection is retained as well when you move to the list view. These selections are retained across browser sessions as well.

> ☞ **Note:** An exception to this case is when you select the root planning folder or team ('Project Teams') in the list view, this selection is not retained when you move to the task or kanban board view; instead, the selection of a specific planning folder or team you had made prior to this one is retained.

In the Planning Board, multiple planning folder and team selections are retained when you navigate to other pages in TeamForge and return to your planning board.

What is a product owner?

The product owner works with the users, or customers, to define how a product can address their needs and wishes.

As the intermediary between the user and the development team, the product owner defines the product scope and requirements.

The product owner is sometimes called a product manager or a marketing manager. If your organization has a project office, that group may include people who function as product owners.

The product owner's primary focus is on the content of the product. The details of implementation planning and scheduling will affect and be affected by the product owner's work, but they are not the product owner's direct concern.

In TeamForge, the product owner's main focus is the feature tree. This is the part of the interface where discussions of product features are organized and tracked.

What is a project administrator?

The project leader takes user stories created by the product owner and works with the project team to manage the process of analyzing, scheduling and executing them.

The project leader coordinates the development team.

The project leader is sometimes called a scrum master, a team lead, or some other variation.

The project leader's primary focus is on the development of the product. This work provides feedback that's useful for defining and revising the scope of the product and for estimating the effort involved, but those are not what the project leader is directly concerned with.

In TeamForge, the project leader's main focus is the planning tree. This is the part of the interface where you define and schedule the work items and personnel needed to deliver the required features.

What is a project?

A project is a workspace where people can use the CollabNet TeamForge applications to collaborate and to create, store, and share data.

All the work you do with CollabNet TeamForge is organized into projects. Any registered CollabNet TeamForge user can create a project, subject to approval by a CollabNet site administrator. After a new project is approved, the project creator can configure project applications, add project members, and create and assign roles to govern each user's individual access permissions and the access permissions of groups of users. A registered CollabNet can also request membership in any CollabNet project. Requests to join projects are submitted to the project's administrators for approval.

How CollabNet TeamForge projects are organized is up to you and your organization. You might choose to create one large, centralized CollabNet TeamForge project in which to manage all of your organization's development work. Or you might choose instead to create a number of smaller projects for each team or sub-project.

Any registered user on the site can create a project, subject to approval by a CollabNet TeamForge site administrator.

You can use a project template to pre-populate new projects with the structure and configuration of an existing project.

When you create a project, it is submitted to the CollabNet TeamForge administrator for approval. You will receive an email notification when the site administrator approves or rejects your project. When your project is approved, you are assigned the Founder Project Admin role and made a project administrator. You can access the project from **My Projects** or **View All Projects** menu option under **My Workspace**.

What is an agile project team?

As a member of a project team, you implement product features according to user stories created by the product owner and planning input from the project leader.

Project team members must use their experience and judgment to estimate the effort involved in work that's been specified by the product owner and the project leader. These estimates will often lead to changes in the product scope and the project plan. In that sense, project team members are an important part of the agile planning process.

The project team members may be known as developers, programmers, engineers, individual contributors, or something else that's apporiate to the type of work being done.

In TeamForge, the project team member's main focus is the tracker. Project members can manage and track their work by observing and maintaining tracker artifacts such as user stories, defects, or tasks.

What is in a project template?

When you create a template from an existing project, each project tool contributes its own structure to the template, and its content if you want it.

For each tool, you can include or omit the actual content that was created with that tool.

For example, suppose you have a project for which you have created some tracker artifacts, and these artifacts have proved useful to members of the project.

- You can include those artifacts in your template, so that people who create a new project from that template will have access to the same artifacts that you developed for your original project.
- You can leave those artifacts out of the template and let future users create new artifacts to fit their own needs.

Imagine that you have documented company-wide process standards on wiki pages in an existing project.

- You can include those wiki pages in a project template, so that the manager of a project created from that template won't have to go find those pages and copy them into the new project.
- Or you can leave the wiki tool empty and let the new project's users create new wiki content for themselves.

👉 **Note:** You can choose not to include content from the original project, but you can't choose not to include a project tool. Every project template you create must include all the project tools, even if the project from which you created the template does not use all the tools. Every project you create from a template will include all the tools.

Tool	Always in Template	Can Be in Template
Tracker	You can have any number of separate trackers in a project. When you create a project template from that project, all the trackers that exist in the original project are copied into the project template. Any default column configurations, select field dependencies or text field validation rules are also included.	You can include the artifacts that were created in the original project, and any parent-child dependencies among them. If users of the original project have shared saved searches, these can be included too.
Planning folders		You can include all the planning folders in the original project, or none of them.
Documents	When you create a project template from an existing project, all the document folders that were created in the original project are copied into the project template.	If users have created documents and attachments in the original project, you can choose to include these documents in the project template.
Tasks	A project can have any number of task folders. All of those folders will be included in any project template that you create from that project.	You can include the tasks that were created in the original project, and any dependencies among them.

Tool	Always in Template	Can Be in Template
Discussions	All discussion forums that have been created in the original project will be included in any project template that you create from that project.	You can choose to include discussion topics, posts and attachments in your template, or leave them out.
File Releases	Any packages or releases that have been created in the original project will be included in the project template. If you have mapped a planning folder to a file release, that mapping is also included.	
Wiki		When you create a template from an existing project, all wiki pages that users have created in the original project can be included in the project template.
Integrated applications	All external applications you have integrated into the existing project become part of the project template.	Which elements are included depends on the application that is integrated. See the **Content includes** section of the **Projects** > **Templates** tab.

What is a project template?

Project templates enable you to capture and re-use the structure and content of existing projects, including project pages, custom tracker fields, and work flow definitions, to speed new project creation and standardize lifecycle processes.

Project templates allow you to create and configure new projects quickly, enforce organizational standards, and facilitate process improvement.

To create a new project, you can use any project template created by any project administrator.

☞ **Note:** You can also create a new project without using a template.

The template name and description appear on the **Templates** tab of the **Projects** list, accessible from the navigation bar on your **My Workspace** page. If your site administrator has made it possible to preview the contents of project templates, click the name of a template to see what's in it.

A template can include the structure of the original project without any of the content, or it can include both the structure and the content of the original project.

- "Structure" means the folders and sub-folders in the original project.
- "Project content" means any work items you or any other project member have created as part of the project. For example, when you create a tracker artifact to manage a piece of work, that tracker artifact is part of your project's content. Any documents you upload to the project and any wiki pages you create are part of the project's content.

What is a release?

A release is a group of one or more files that are published as a unit.

Each release can have a maturity level attribute to describe its state of completeness. Maturity levels are predefined and include development build, alpha, beta, and general availability.

The TeamForge file release system enables users to publish files and groups of files to selected audiences. Using role-based access control, administrators can control which project members can access each package or release.

What is a story point?

A story point is a measure of effort that expresses the relative difficulty of implementing a user story. You can use story points, also referred to simply as "points", to help estimate how much work can be done in a sprint.

Story points are useful for relative measurement. A story point has no specific value in hours, or lines of code, or anything else. When you are estimating work, use story points only to compare one piece of work with another.

When you record a value in the **POINTS** field of an artifact, that value is added to the total points (story points) in the planning folder that the artifact belongs to. You can then use that figure to support forecasting.

In the parent artifact, you can view the total of story points assigned to each of its children. The calculator icon indicates that the artifact's points is a sum of its child artifacts' points within the project. If the parent artifact has children in other projects across the TeamForge, you can include those points in the total as well. The icon, 🗡 indicates that the artifact's points include its foreign child artifacts' points.

👉 **Note:** The Tracker Admin needs to set the tracker to include foreign children in points calculations.

What is a task dependency?

Task dependencies show what must be done before a given task and what will be done after the task is complete. Project managers can create dependencies between any two or more tasks in a project.

Task dependencies allow you to establish relationships between tasks. This can help to frame a task in the context of a larger project plan.

There are two types of task dependencies:

- *Predecessors* are tasks on which another task depends.

  For example, before you can complete the task "Finalize project plan," you might need another project member to complete the predecessor task "Hire new staff."
- *Successors* are tasks that depend on your task.

  In this example, "Finalize project plan" is a successor task to "Hire new staff."

👉 **Note:** Task dependencies and tracker artifact dependencies are different things.

- For tracker artifacts, an artifact with dependencies (a "parent" artifact) can't be considered closed unless all of its dependent artifacts ("children") are closed.
- For tasks, a dependency means one task can't start until another task is completed.

👉 **Important:** A task cannot be dependent on itself. You cannot make a task both a predecessor and a successor to another single task. If you attempt to do so, you will receive a circular dependency error.

What is a tracker?

A tracker is a collection of records that follow the development of a unit of work from conception through to completion. You can create a tracker to manage almost any kind of work that your project calls for.

In each TeamForge project, you can create any number of trackers. Each tracker tracks a different type of data.

For each tracker, you can define values for status, category, and other default fields. You can create your own fields to capture additional data that is specific to your project or organization. You can also create tracker workflows to specify the criteria necessary for users to change tracker status values.

Individual tracker entries are referred to as tracker artifacts. The role-based access control system enables you to control which project members are allowed to view, create, and edit tracker artifacts. Within a project or across the projects, a tracker can be cloned along with the workflow.

Summary information about the number and status of project tracker artifacts is provided on each project's **Tracker Summary** page. The Project Dashboard also provides an at-a-glance overview of the status of each project member's projects, including information on the number and status of the tasks and tracker artifacts in each project, and project overrun and underrun statistics.

What is tracker work flow?

To help users handle their tracker items effectively, you can set up some work flow rules. Workflow rules require a user to do something to a tracker before they can reassign it or update its status.

Administrators can define these kinds of workflow rules:

Status transitions that a user can make based on an artifact's current status.	For example, if an artifact is **Open**, you can specify that it can be changed to **Pending** or **Cannot Reproduce**, but not to **Closed**.
Status transitions that a user can make based on his or her role.	For example, if an artifact is **Pending**, you can specify that only users with the role **QA Engineer** can change it to **Closed**.
Field values that a user must provide when making a specific status change.	For example, if an artifact is **Closed**, you can specify that a user must enter a comment in the **Comments** field before changing the artifact's status to **Open**. You can also require an attachment.

You can create a workflow for each combination of tracker status values in a tracker A tracker can be cloned within a project or across the projects along with the workflow. If a user role is not existing in the destination project, a new role is created with the same name. The permissions associated with the role are not copied from the source project.

What is a user group?

To manage multiple users at once, create a group to represent them.

You can create a group to facilitate managing many users who share one or more characteristics.

For example, giving all the users in the accounting division access to all financial projects might be laborious if you assigned the permissions one at a time. Instead, create a group and assign it access to the financial projects category.

A user group can have any number of roles. When a role is assigned to a group, every member of that group has that role.

If a project is a subproject of another project, it may inherit user groups and their associated roles from the parent project.

☞ **Note:** A user who has a role in a project by virtue of group membership is not necessarily a member of that project. Becoming a member of a project is a separate process.

Group permissions are cumulative. This means that each member of the user group has all the access permissions allowed by all of the assigned roles, plus any permissions that may have been assigned by other methods, such as application permissions or individually assigned roles.

Velocity and average velocity

Your team's velocity is the amount of work the team has shown it has completed in an iteration. You can use this information to help estimate how much work can be completed in future iterations.

In the planning folder summary page, you can see your team's velocity for the specific Iteration planning folder.

☞ **Note:** Velocity is displayed only on the Iteration planning folder summary page.

Velocity is calculated based on the sum of points of closed artifacts in an iteration.

Note that velocity calculation takes into account the following:

• Points of the parent artifacts, that is, only closed artifacts which are at the top level of the planning folder. For example, in the following illustration, only the points of the highlighted artifacts and not their child artifacts are taken into account for velocity calculation.

- Autosummed points of these closed parent artifacts do count as well.

Velocity only makes sense as a relative measure. There is no specific velocity that is good or bad or standard, because no two teams take on work of exactly the same scale or complexity. However, if your team's velocity is increasing from sprint to sprint, you can surmise that you are on the right track.

Average velocity

The average velocity is calculated for completed and ongoing iterations and is displayed only on the Release planning folder summary page. Average Velocity calculation is as follows:

Average velocity = Sum of iteration velocity of all iterations within the specific Release/Total number of iterations

Note the following for velocity and average velocity calculation:

- Nested iteration planning folders (child iteration planning folders or subfolders) are not included for velocity or average velocity calculation for the parent planning folder. For example, in the following illustration:

  Average velocity for Release 1 = Sum of velocity of Iteration 1 and Iteration 2 / 2

  In the above calculation, Iteration 1.1 (being the subfolder) is not included. Similarly, velocity calculation for Iteration 1 does not take into account its subfolder Iteration 1.1.

What is an activity table component?

An activity table is a special kind of text component that gives project members quick access to a focused set of artifacts, action items and work products.

An activity table includes organized links to up-to-the-minute information about work that is going on in the project. You can add, remove or change these links with the HTML editor that comes with every text component.

You can add static links to particular artifacts, or you can link to a search query. When you link to a query, you enable project users to get all the results of the query with a single click.

The activity table template is just a suggestion for how you might want to organize your information. You can change any of its properties to support the way your project members work.

What is a documents component?

A documents component allows team members to work with documents directly from a project page.

A documents component is like a window into your Documents tool. Instead of making users go to the documents tool and search for a document on their own, you can create a documents component on your project page that is linked to a folder in your documents tool. Each page can have its own list of relevant documents.

Users still go the documents tool, but now you guide them to what they need there.

What is a global project role?

A global project role is a ready-to-use role available in all projects. Only site administrators or restricted site administrators can create and manage a ready-to-use role.

As a project administrator, you can use global project roles provided by the site administrators instead of creating and managing roles tailored to your projects.

☞ **Note:** You can use the ready-to-use roles to set up your team faster and with little fuss. However, you may not be able to edit the ready-to-use roles.

Before you create a role in your project, it is a good idea to check all the available ready-to-use roles. You are likely to get ones that grant the desired set of permissions.

Global project roles serve a different purpose from that of a site-wide role. The site-wide roles enable site administrators to create restricted site administrators for providing assitance in site management. Besides that, site-wide roles can be used to grant tool/application access across the site to a user.

What is a project page?

A CollabNet project page is a place where users can see and add information about the project, such as messages from the project manager, open issues or documents you want people to read.

You can build your own project pages to design, manage and track your project's lifecycle. When you create a project page, users automatically see it in their navigation panes.

To post information or provide functionality on your project page, you add a page component of the appropriate type for the information or functionality you are working with.

For example, to let people know about how the project is coming along, add a Project Statistics component to the page. To let project members upload documents for other users to read, you add a documents component.

If the page is a part of a parent page, it appears as a node in the tree in the left navigation area.

You can add a page directly to the top of the project, or as a subpage.

☞ **Tip:** When you have your project pages defined and your process honed, you can export your project pages in the form of a project template, so other projects in your organization can reuse the resources you created. See *Create a project template* on page 250.

What is a project page component?

A project page is a collection of simple portlet-like components that enable you to add custom HTML content, reports, project tracker queries, and much more to the project home page so your team quickly access what is important.

What is an Available upon Request role?

The Available upon Request role is a role which a project member in TeamForge can submit a request for. You must be a site administrator or project administrator to create an Available upon Request role.

While creating global or project roles, you can select the **Available upon Request** option to allow the project member to see the role in **Request Additional Roles** section on the **Project Home** page.

When a project member requests a role, the request is submitted to the project administrator for approval. The project member receives an email notification when the request is either approved or denied.

☞ **Note:** Based on the project administrator's discretion, a few role requests may be granted immediately on request, while the other role requests may need approval.

What is a news component?

A news component is a way to get developing information to project members via a project page.

You can use a news component to maintain a journal or blog about your project, to announce milestones met, or to share information about a rapidly changing situation.

There is a single set of news posts for your project. If you put a news component on more than one project page, the same new posts will show on all of them.

What is a project page component for?

Project page components add functionality to a project page. You can use components to communicate with users or project members, or to invite them to contribute information or resources to the project.

- Promote a standard lifecycle for all of your projects by using project pages to lay out information the way that works best for your business.
- Increase developers' productivity by putting the information most relevant to your project right on the homepage.
- Improve project coordination by combining related documents, graphs, artifacts, and content into single, digestible web pages.
- Maintain information security. The project manager controls who has access to each of the pages and components.
- Keep project members informed with live data and content that can be quickly updated as often as you like.

Project pages put the project manager in the driver's seat. You define how you want your project to look and what information you wish to share with your members and customers. Using a simple portlet-like layout, project pages let you add custom HTML content, reports, project tracker queries, and much more to the project home page so your team can quickly access what is important. All your pages are organized and displayed in a simple navigation menu on the left side of the home page.

In addition to customizing the project home page itself, you can also create and integrate additional web pages on the fly. Whether you need a page to display daily status reports or want to publish an HTML blog, a few clicks is all you need. No need to use any fancy tools or even know HTML.

For example, on a project page titled **QA**, you might have:

- A trend graph showing the open bugs in your project over a period of time.
- A Tracker query list showing you what hot bugs are currently in the system.
- A block of custom HTML content that the QA lead updates daily giving users his or her take on how the quality of the product is doing that day.

What is a Tracker Search Results component?

A Tracker Search Results component provides quick access to Tracker information from a project page.

Some tracker searches are used so frequently that you want people to be able to see their results without exploring. For example:

- People working on your project may need to know at the beginning of every work day which artifacts have been updated overnight by a remote team.
- A cross-project leadership team may need a daily count of started and completed work items to support forecasting and planning.

Such heavily-used information might be a good candidate for a tracker search component on a prominent project page, such as the project home page. When you set up this component based on a popular tracker search that you devised and saved, project members and users can save time by viewing the information at a glance, without having to go find the information themselves.

☞ **Note:** If your site administrator has set up Project Tracker for your project, you can also use a project page query component to search your Project Tracker data.

What is a project statistics component?

A *project statistics component* gives users a graphical view of recent statistics for tasks, trackers, documents, and file releases from the project page.

For more information about the project statistics component see *Create a project page component* on page 106.

What is TeamForge?

TeamForge integrates software configuration management (SCM), continuous build and test, issue tracking, project management, lab management, and collaboration tools into a single Web-based application lifecycle management platform that empowers distributed teams to deliver great software.

1. *My Page* on page 317
2. *Join a TeamForge site* on page 9
3. *Manage source code* on page 76
4. *My Page* on page 317
5. *The Project Dashboard page* on page 319
6. *What is a project?* on page 961
7. *Watch for changes* on page 100
8. *Join a TeamForge site*
9. *CollabNet Desktop - Microsoft Windows Edition*
10. *CollabNet Desktop - Eclipse and Visual Studio Editions*
11. *Help for project managers*
12. *Work on tasks* on page 48
13. *Manage source code* on page 76
14. *Plan and track anything* on page 21
15. *Build and test your software* on page 96
16. *Provision hardware* on page 88

What is a text component?

A text component is a self-contained editor with which you can write anything you want on your project page.

You don't have to know anything about HTML.

Text components include everything you would expect in a fully functional HTML editor. You can add tables, upload images to embed into the page, create hyperlinks, and assign font colors and styles. Advanced users can work directly with the HTML source.

What is a wiki component?

A wiki component lets you link to an existing wiki page from a project page.

What is the complete wiki syntax that TeamForge supports?

Since TeamForge makes use of the JSPWiki rendering engine internally, you may reference the JSPWiki syntax documentation at the link below for a complete list.

http://www.ecyrd.com/JSPWiki/wiki/TextFormattingRules

Please note however that JSPWiki has progressed beyond the release available in TeamForge currently, so there may be some discrepancy between what JSPWiki supports and what TeamForge supports. TeamForge also does not currently support the JSPWiki plugin framework, so any formatting plugins referenced on the URL will not be available for TeamForge.

What is Team?

Teams helps you create logical groups of cross-functional members comprising architects, developers, testers and so on.

A team's view of backlogs complements the planning folder view. While planning folders represent backlogs (release, iteration, etc), a team's list view represents the team's view of the backlog (across releases, iterations, etc).

In an agile environment, project activities are broken into smaller units of work items and are assigned to various project teams. The Teams feature allows you to create these physical teams in TeamForge so that you get the specific team's view of backlogs. This knowledge of the team's view of work items helps you facilitate effective communication and execute project activities in a more structured and organized fashion. For example, you can easily view and filter artifacts which are taking longer than estimated within a team, analyze the scenario, and identify the impediments. Once identified, you can communicate them to the relevant team(s), re-assign it to other appropriate team(s) or team member(s), and resolve them quickly.

What is a team owner?

Any project member can be designated as a team owner.

The team owner does not have the permission to create or delete teams. They can view their team information and edit them; while editing their team details, they can add or remove members, or designate another member as the team owner.

General FAQs

Common questions on TeamForge usage.

What's the best way to search the CollabNet User Information Center?

Searching this site with the Eclipse search tool is a little different from searching the Web with a service such as Google.

Here are some guidelines for getting the most out of a search on this site.

Setting the search scope

To quickly locate topics on a particular subject in the documentation, enter a query in the **Search** field. You can narrow the scope of your search by selecting only the sections you are interested in. To set the search scope:

1. Click **Search scope** next to the **Search** field.
2. Select the **Search only the following topics** option.
3. Click **New**.
4. Type a name in the **List name** text box.
5. Select one or more guides (select the check boxes) from the **Topics to search** list. For example, you can select the check boxes corresponding to the CollabNet TeamForge 7.0 user and administration guides.
6. Click **OK**.

Once the search scope is set, you can enter your search query and click **GO**. Your search is restricted only to the guides you selected.

Search query guidelines

- Use AND to require the term on each side of the AND operator be present in the topic.

  > **Note:** When you don't put anything between terms, the search tool assumes you mean AND. Topics that contain every term in the query are listed in the search. For example, if you enter database service, topics that contain both the term database and the term service display. Topics that contain only the term service or only the term database do not display.

- Use OR before optional terms. For example, if you enter database OR service, topics that contain either the term database or the term service display.
- Use NOT before a term that you want to exclude from the search results. For example, database NOT service displays topics that contain the term database, but do not contain the term service.
- Use ? to match any single character. For example, plu? displays topics that contain plug.
- Use * to match any set of characters, including an empty string. For example, plu* displays topics that contain plug or plugin.
- Use double quotation marks to enclose a term that is to be treated as a phrase. For example, "edit menu" displays topics with this entire term, not topics with only the term edit or the term menu.
- Case is ignored. For example, database service displays database service, Database Service, and DATABASE SERVICE.
- Punctuation acts as a term delimiter. For example, web.xml displays topics that contain web.xml, web, and xml. To display only topics that contain web.xml, enclose the term in double quotes.
- In a search query, if you enter create, topics that contain create, creates, creating, and creation are displayed. To only see the term create, enclose the term in double quotes.
- The following English words are ignored in search queries: a, and, are, as, at, be, but, by, in, into, is, it, no, not, of, on, or, s, such, t, that, the, their, then, there, these, they, to, was, will, with.

Is the CollabNet TeamForge SOAP API backward-compatible?

The CollabNet TeamForge Enterprise Edition 5.x and CollabNet TeamForge Enterprise 6.x APIs (including Service Pack updates) are fully compatible with CollabNet TeamForge.

Applications developed using these earlier APIs will continue to function after you upgrade to CollabNet TeamForge. A release of CollabNet TeamForge may be accompanied by updates to the API. These changes are always backward compatible with earlier versions of the API. However, the calls from the different API versions are not interchangeable.

- For updates and patch releases, existing methods are incremented, not overwritten. For example, if an update is made to the `createDocument` method, it is reflected in a new method createDocument1. You do not have to update your existing applications to reflect a new API with any Service Pack or Hot Fix release.
- For a major CollabNet TeamForge release, such as this 8.2 release, updates are merged into a new API version. At this point, if you want to use the new API calls, you must update your existing applications.

What can I do with the SOAP API?

You can use the TeamForge SOAP API to do almost anything a user can do in the TeamForge web user interface.

TeamForge exposes a subset of the APIs defined by the application server as web services, through the SOAP protocol.

A SOAP proxy server and a SOAP API layer, both running on Apache Axis, expose a set of web services representing each TeamForge application. The SOAP server serves the following functions:

- Provides web services by accepting SOAP requests from the clients.
- Performs SOAP client authentication.
- Implements TeamForge role-based access control (RBAC) and caching service.
- Accesses the application server via RMI stubs.

You can get the API here: *http://www.collab.net/community/teamforge*

The SOAP API can provide more types of functionality than the Web UI alone. For example, it can be time-consuming for a user with Tracker Admin permissions to copy a tracker workflow from one tracker to another in the Web UI. However, the SOAP method that provides a workflow copy function makes this task easy.

Access permissions

For access permissions (authorization, as distinct from authentication), the SOAP API follows the same rules imposed by the TeamForge role-based access control (RBAC) system. A user using a SOAP API-based client has no more access to data than they have in the Web UI.

The CollabNet TeamForge architecture allows you to quickly and easily develop integration points between CollabNet TeamForge and applications you develop.

User-centric services

All services and APIs are user#centric, meaning that all integrated applications must establish an individual connection to the SOAP server for each user. This differs from programming directly with an application server where one connection can be established for any number of users.

Activities that can be performed using the SOAP interface are by definition user#based, such as retrieving a list of a user's projects, tasks, or assigned tracker artifacts. These activities therefore require an individual connection for each user.

Requiring individual connections also ensures that role#based access control is checked for each action performed by each user. To ensure that security is enforced, RBAC checks are performed on each SOAP API call and cannot be disabled at the client level.

Consistent interaction

While each CollabNet TeamForge service has its own SOAP interface, interaction with each service is designed to be as consistent as possible. The calls for each service are similar, although the data format and specific call parameters may be different.

For example, the following calls are consistent across all services:

- `list`
- `get`
- `set`
- `delete`

- `create`

The call parameters, however, are different. For example:

- When working with the CollabNet service, you might call `getProjectList(string sessionID)`.
- When working with the TaskApp service, you might call `getTaskList(string sessionID, string taskFolderID)`.

How should my API client store user passwords?

Any reputable method of storing passwords will work, as long as your site is protected by SSL.

All client tools rely on the TeamForge SOAP API for authentication, and therefore use whatever authentication method TeamForge is using.

☞ **Important:** Because SOAP is simply XML transmitted over HTTP, all values are sent in clear text. For that reason it is very important that your TeamForge site be SSL-enabled and protected by server-side SSL certificates. This will ensure that any usernames or passwords sent from a client tool will be encrypted.

Many standalone client tools are able to cache a copy of the user's credentials to make it easier for them to access the site. The CollabNet Eclipse Desktop stores passwords in the encrypted Java keystore, and the CollabNet Windows clients use the Windows keystore.

CollabNet's Subversion clients and other Subversion clients, such as Tortoise and Subclipse, are also able to store user credentials. While CollabNet has no control over how third-party tools store such credentials, it our experience that the mainstream tools all use an appropriate keystore for secure storage of user credentials. CollabNet recommends that customers independently verify the storage methods of those tools and set a policy appropriate with their own security guidelines.

Subversion users on Linux systems have the option to use the Gnome keyring to securely store user credentials. CollabNet recommends that customers set their own policy for how their users should use the Linux Subversion client.

How does an application interact with TeamForge SOAP services?

TeamForge exposes a subset of the APIs defined by the application server as web services, through the SOAP protocol.

SOAP APIs

A SOAP proxy server and a SOAP API layer, both running on Apache Axis, expose a set of web services representing each TeamForge application.

The SOAP server provides the following functions:

- Provides web services by accepting SOAP requests from the clients.
- Performs SOAP client authentication.
- Implements TeamForge role#based access control (RBAC) and caching services.
- Accesses the application server via RMI stubs.

While each TeamForge service has its own SOAP interface, interaction with all the services is designed to be as consistent as possible. The calls for each service are similar, although the data format and specific call parameters may be different.

For example, the following calls are consistent across all services:

- `list`
- `get`
- `set`
- `delete`
- `create`

The call parameters, however, are different. For example:

- When working with the CollabNet service, you might call `getProjectList(string sessionID)`.
- When working with the TaskApp service, you might call `getTaskList(string sessionID, string taskFolderID)`.

User-centric services

All services and APIs are user#centric, meaning that all integrated applications must establish an individual connection to the SOAP server for each user. This differs from programming directly with an application server where one connection can be established for any number of users.

Activities that can be performed using the SOAP interface are by definition user#based, such as retrieving a list of a user′s projects, tasks, or assigned tracker artifacts. These activities therefore require an individual connection for each user.

Requiring individual connections also ensures that role#based access control is checked for each action performed by each user. To ensure that security is enforced, RBAC checks are performed on each SOAP API call and cannot be disabled at the client level.

How do I require approval for new user accounts?

You can configure the system so that new users can create their own accounts, but the accounts are not activated until a site admin approves them.

To enable this mode of operation, add the following line to `/opt/collabnet/teamforge/sourceforge_home/etc/sourceforge_configuration.properties`:

```
sf.approveNewUserAccounts=true
```

Once this line has been added to the file, restart TeamForge for it to take effect.

Please note that site admins can still create accounts for new users and they will not be held for approval. Also note that the user will receive an email from TeamForge telling them to confirm their password by clicking on the given link, and the link will not work. The password is properly set on account approval.

How does TeamForge automatically sum up effort estimates?

As you refine your agile project plan, you break down user stories into smaller stories, and eventually into tasks. TeamForge can watch the changing effort estimates all the way down the hierarchy, and give you a running total for each parent artifact and all its children.

Example

Imagine that you originally created a user story to describe the need for a "Shopping Cart" feature in your e-commerce application. You gave this story a rough effort estimate of 20 units, and entered that figure in the **Effort** field.

Upon further discussion, your project team recommends breaking the story down into three tasks:

- Database (5 units)
- Back End (15 units)
- UI (8 units)

You create artifacts for those three tasks, identifying them as children of the user story artifact, and enter their respective effort estimates in the **Effort** field in each task artifact.

Back in the "Shopping Cart" story artifact, you select **Autosum effort** to indicate that effort number for this artifact should be rolled up from its children.

Now, rather than showing the manually entered number (20 units), your user story artifact shows the figure derived from adding the effort estimates for all three child artifacts (28 units).

Double-counting?

Effort numbers are never "double-counted" when auto-summing is on. For example, if the sum for planning folder A would include effort for child C and parent P, and P has autosumming turned on, C's effort number is only counted once.

However, when auto-summing is off, the effort numbers from parent P and child C can both contribute to totals. So if a planning folder A includes parent P (autosum=off, effort=3) and child C (effort=5), then the pair contributes a total of 8 to A's total.

An artifact's effort number is calculated from the effort numbers of its immediate children artifacts only. This is true whether or not those children's numbers are themselves automatically derived.

☞ **Note:** When a tracker is disabled, artifacts from that tracker do not contribute to the effort totals calculated for any planning folder they are in.

Permissions

If you are a long-time user of TeamForge tracker features, be aware that autosumming can lead to some situations you haven't seen before. For example, picture these scenarios:

*   Artifact A is assigned to Sanil. He therefore has permission to edit that artifact, but he does not have permission to edit the artifact's parent, artifact B. Artifact B is assigned to Sergey, who has edit permission for both artifacts. Sergey turns on autosumming for artifact B. When Sanil updates his effort estimate for artifact A, the content of the **Effort** field in artifact B is updated as a result, even though only Sergey is explicitly authorized to change values in artifact B.
*   Connie creates artifacts X and Y, and declares both of them children of artifact Z. Then she assigns artifact Z to Thiru. Thiru now has edit permission for artifact Z. But he cannot edit artifact X or artifact Y, because he does not have access to the separate tracker where they live. When Thiru turns on autosumming for artifact Z, that artifact's **Effort** field includes the sum of artifacts X and Y, even though Thiru is not explicitly authorized to see any data from those artifacts. (In fact, he may not even know that artifacts X and Y exist.)

How does TeamForge use Velocity templates?

Velocity is the templating language that CollabNet TeamForge uses to render areas of the site with dynamic information.

You can override the instructions contained in any of these Velocity templates by placing a file of the same name in the equivalent path in the `branding` repository in the `look` project on your site.

Velocity templates are located in the `templates` directory in the branding repository.

Velocity file	Description
`menu_bar.vm`	Controls the rendering of the top bar across all pages in the system. Displays a small login form, the site logo and current user information as well as the search and projects drop down menus.
`blank_menu_bar.vm`	Contains only the top logo, without the menu that appears below it.
`body_header.vm`	Rendered immediately after the opening body tag. If a site requires everything to be contained in some other container, this template can be used.
`body_footer.vm`	Rendered immediately before the closing body tag. If a site requires everything to be contained in some other container, this template can be used.
`button_bar.vm`	Controls the rendering of the bar beneath the menu bar, which contains the 'Quick Jump" link as well as

Velocity file	Description
	the buttons that appear on any project page (the one containing the applications). Site admin pages, user settings pages (e.g. my workspace, dashboard) and project pages use different sets of buttons that are passed into this template for rendering.
`content_header.vm`	Rendered after the button bar; wraps the actual contents of the page being viewed.
`content_footer.vm`	Rendered before the body footer; wraps the actual contents of the page being viewed. Contains the Copyright notice.
`sfmain/home.vm`	Velocity template that generates the site home page.
`sfmain/project_home.vm`	Velocity template that generates the default project home page.

How do URLs in CollabNet TeamForge 8.2 compare with URLs in CollabNet Enterprise Edition?

The URLs in CollabNet TeamForge 8.2 and CollabNet Enterprise Edition are not exactly the same, but they can be made to work together.

Any bookmarks you have saved in CollabNet Enterprise Edition projects can now be accessed from within CollabNet TeamForge 8.2.

CollabNet Enterprise Edition URLs that specify item identifiers, such as artifact ID, document ID or discussion post ID, are specified are routed to the exact item page in CollabNet TeamForge 8.2. However, CollabNet Enterprise Edition URLs that do not have exact item identifiers may be routed to the tool's main page or project home page. As an example, link to CollabNet Enterprise Edition's specific post ID is redirected to the exact post ID page whereas, the link to a discussion forum may be re-directed to the **Discussions>Forum Summary** page.

☞ **Note:** If any of your bookmarked CollabNet Enterprise Edition URLs are not routed to the CollabNet TeamForge 8.2 URLs, contact your system administrator.

Only the URLs listed here will work the same way in TeamForge 8.2 as in CollabNet Enterprise Edition. You may need to change some other URLs manually and communicate the changes to your site members. For example, there may be a form submission to enter Project Tracker artifacts which may look like: `/servlets/tracking/action/CreateArtifact?type=defect`. The equivalent URL in TeamForge 8.2 would be: `/tracker/tracking/linkid/prpl1042/action/CreateArtifact?type=defect`. You should get the corresponding URLs from the CollabNet TeamForge site and update your HTML or project pages accordingly.

☞ **Note:** If you are making the transition via a staging site, do this in production, after the upgrade is complete, because changes you make on the stage site are not preserved.

Issue Tracker

CollabNet Enterprise Edition URLs, where issue tracker IDs or PCNs are specified, are re-directed to the exact artifact or tracker appropriately. Any other Issue Tracker tool-specific URLs are re-directed to the CollabNet TeamForge 8.2 Tracker application's home page.

- After migration, enter a CollabNet Enterprise Edition URL of an Issue Tracker issue in the browser address bar like: `https://project.domain/ issues/show_bug.cgi?id=xxx`. This gets re-directed to a URL like: `https://host/sf/go/artfxxx`.

Discussions

CollabNet Enterprise Edition URLs, where forum IDs, topic IDs or post IDs are specified, are re-directed to the exact forum, topic or post appropriately. Any other discussions tool-specific URLs are re-directed to the CollabNet TeamForge 8.2 Discussions application's home page.

- Bookmarked forum URLs of the form `https://project.domain/ds/viewForumSummary.do?dsForumId=xxxx` are correctly redirected to the corresponding forum in the CollabNet TeamForge's Discussions tool like `https://host/sf/discussion/do/listTopics/projects.project/discussion.forumId`.
- Bookmarked message url's of the form `https://project.domain/ds/viewMessage.do?dsForumId=xxxx?dsMessageId=xxxx` are correctly redirected to the corresponding topic or post page depending on whether the message represents a post or a topic, like: `https://host/sf/discussion/do/listPosts/projects.project/discussion.forumid.topicid...`
- Any other url's of the form `https://project.domain/ds/....` are redirected to the Forum summary page as:`https://host/sf/go/projects.project/discussion..`

Documents

CollabNet Enterprise Edition URLs, where document IDs are specified, are re-directed to the appropriate exact document. Any other document tool-specific URLs are re-directed to the CollabNet TeamForge 8.2 Documents application's home page.

- CollabNet Enterprise Edition URL: `https://project.domain/servlets/ProjectDocumentReservationAdd?documentID=xxx` is re-directed to `https://host/sf/go/docxxx`.

CVS

- For repository browsing purposes, CVS URLs are appropriately redirected to CollabNet TeamForge 8.2 ViewVC URLs. `https://project.domain/source/browse/project/dd.txt?r1=x&r2=y` is re-directed to `https://host/integration/viewvc/viewvc.cgi/dd.txt?root=project&system=exsyxxxx&r1=x&r2=y`
- For purposes of checking out or committing with a client, see the **Source Code** tab of the CollabNet TeamForge 8.2 project for the new repository path. CVS working copies must be recreated (checked-out afresh).

SVN

Subversion URLs for specific revisions of a file, log messages or difference between two revisions are appropriately redirected to CollabNet TeamForge 8.2's ViewVC URLs.

- CollabNet Enterprise Edition URL: `https://project.domain/source/browse/project/trunk/test.txt?view=markup&pathrev=x` is re-directed to `https://host/integration/viewvc/viewvc.cgi/trunk/test.txt?root=project&system=exsyxxxx&view=markup&pathrev=x`
- CollabNet Enterprise Edition URL: `https://project.domain/source/browse/project/trunk/test.txt?view=log&pathrev=x` is re-directed to `https://host/integration/viewvc/viewvc.cgi/trunk/test.txt?root=project&system=exsyxxxx&view=log&pathrev=x`
- CollabNet Enterprise Edition URL: `https://project.domain/source/browse/project/trunk/test.txt?r1=x&r2=y&pathrev=x` is re-directed to `https://host/integration/viewvc/viewvc.cgi/trunk/test.txt?root=project&system=exsyxxxx&r1=x&r2=y&pathrev=x`

PD/Task

Any URL of CollabNet Enterprise Edition's Dashboard/Task pages are redirected to the CollabNet TeamForge 8.2 Tasks application's home page.

- CollabNet Enterprise Edition URL: `https://project.domain/servlets/activity/*.do` is re-directed to `https://host/sf/taskmgr/do/viewSummary/projects.project/taskmgr`.

Wiki

All CollabNet Enterprise Edition Wiki URLs are redirected to the CollabNet TeamForge 8.2 Wiki application's home page.

- CollabNet Enterprise Edition URL: `https://project.domain/wiki/...` is re-directed to `https://host/sf/wiki/do/viewPage/<projectname>/wiki/HomePage/sf/wiki/...`

Project Metrics

CollabNet Enterprise Edition URLs of Report tool are redirected to the CollabNet TeamForge 8.2 Reporting application's pages.

- CollabNet Enterprise Edition URL: `https://project.domain/servlets/Report*...` is re-directed to `https://host/sf/reporting/do/listReports/<projectname>/reporting`.

Announcements

All CollabNet Enterprise Edition News URLs are redirected to CollabNet TeamForge 8.2's Announcements.

- CollabNet Enterprise Edition URL: `https://project.domain/servlets/ProjectNewsAdd...` is re-directed to `https://host/sf/sfmain/do/myPage?selectedTab=news`.

Help

CollabNet Enterprise Edition Help page URLs are redirected to appropriate help pages on "help.collab.net".

- CollabNet Enterprise Edition URL: `https://project.domain/nonav/servlets/HelpTOC` is re-directed to `http://help.collab.net/index.jsp`.

Moving to a CollabNet TeamForge project from a CollabNet Enterprise Edition project

If you have used the tools in CollabNet Enterprise Edition, you'll find the equivalent tools in TeamForge familiar.

Most of the tools and their functionality are similar between CEE and CTF 8.2. However, within each tool, some work-around might have been done to bring over your CEE project information.

This table shows how tools in CEE map to CTF 8.2 tools:

CEE Tool	CTF 8.2 Tool
Projects	Projects

CEE Tool	CTF 8.2 Tool
Category (Project)	Category (Project Grouping)
Discussion Services	Discussion
Docman	Document
Issue Tracker	Tracker
PCE	PCE
Wiki	Wiki
Project Tracker	Project Tracker (Integrated Application)
Framework	Roles and Permissions
SCM	SCM
Remote Publishing	Remote Publishing
Project Dashboard	Project Dashboard
Announcements	News

Do I have to use the administrator-given password always?

No, you don't have to use the administrator-given password beyond your first login into CollabNet TeamForge.

The site administrator may have provided you with a user name and password after creating your user account in CollabNet TeamForge. When you login using those credentials, you might be asked to change your password for security reasons. At this time, you can set a password of your choice.

☞ **Tip:** You will not be asked to change your password if you had created your own account, or if a password-request had been raised for you or if an administrator had reset your password.

What software configuration management tools are available in CollabNet TeamForge ?

In CollabNet TeamForge, users can browse the contents of a project's source code repositories and view detailed information about code commits, changed files, and associations with other CollabNet TeamForge items.

CollabNet TeamForge is not intended to replace your SCM tool. Code must be checked in using Subversion, CVS or GIT.

How do I determine which version of TeamForge is currently running?

To determine the version of TeamForge you are running, click on the question mark icon in the upper-right corner of the user interface and choose About TeamForge from the drop-down menu.

You can also find the version number inside the following files:

```
 * $SOURCEFORGE_HOME/etc/version
 * $SOURCEFORGE_HOME/etc/integration-version
```

Why can't I reply when someone comments on my review?

By default, TeamForge doesn't add the review initiator to a review, which would be needed to facilitate this.

To work around this, simply add yourself as an optional reviewer when creating the review. Alternatively, some customers have chosen to create a forum in TeamForge, and then include the forum's posting address in the review notes.

How does TeamForge support dynamic planning?

TeamForge helps you maximize your team's effectiveness by keeping you in close touch with the multiple moving targets facing your project.

In a development project, each piece of the picture constantly changes in response to changes in the other parts. As you continue to iterate through the process, a feedback loop like this takes shape. (Click a node for more detail.)

The new Dynamic Planning features in TeamForge 8.2 give you a more open and extensible platform that integrates and centralizes the software development tools necessary in modern application life cycles.

👉 **Note:** The Dynamic Planning features helps you effectively utilize Agile--like processes, but you can use any process model you like.

Tracker summary screen

TeamForge 8.2 incorporates a new Tracker Summary screen display to accommodate planning folders, treeviews, and multiple tracker viewing. The Tracker Summary section is available at the top of the screen with summaries of open and closed artifacts as well as a summary of open artifacts by priority. The Planning Folders section is located at the bottom of the Tracker Summary screen. This section includes summaries of open and closed artifacts, a summary of open artifacts by priority, and a summary of Effort for each planning folder.

Tree view

TeamForge 8.2 incorporates an expandable and collapsible tree view of the planning folder hierarchical structure to display parent/child relationships of artifacts.

The tree view allows for the viewing of artifacts in a hierarchical structure and displays parent/child relationships across multiple trackers.

How do you measure "effort?"

Effort is a uniform span of time for measuring work on your product. Defining the unit of effort is an essential part of planning your work.

TeamForge 8.2 shows you the effort that has been estimated, and actually expended, for each artifact on the tracker summary screen. Parent artifacts can automatically add up these effort figures from their child artifacts' effort figures. The calculator icon indicates that the artifact's effort is a sum of its child artifacts' effort within the project. When the effort from children artifacts in projects across the TeamForge (foreign children) is included in calculations, the icon, 🖐 appears.

👉 **Note:** When a tracker is disabled, artifacts from that tracker do not contribute to the effort totals calculated for any planning folder they are in.

The unit of effort should be something that makes sense in your environment. TeamForge does not require any particular unit. Your unit of effort might be hours per person, days, weeks, or something else. (If you are using a scrum-based project methodology, you may have opted to measure effort in relative terms, using points (story points), in which case you can leave the **Effort** fields blank.)

For example, some teams use the "ideal hour" as their standard unit of effort. To define an ideal hour, consider all the activities in a standard work day that must be done but don't directly contribute to development: installing and configuring tools, eating lunch, responding to email and instant messages, providing customer support, etc. For every hour of direct development work, how much time goes into these activities, on the average? If the answer is about a half hour, then your ideal hour is 1.5 clock hours.

Now consider a task that you judge to represent about four hours of direct development work. The value you'll enter in the **Effort** field is 6, because for each hour of direct development you'll need an extra half hour to make that development work possible.

In an environment with a lot of overhead -- for example, a group that relies on a very complex tool set -- your ideal hour might equal two clock hours, or perhaps much more. This is not in itself a problem: the point is not to suppress needed activities, but to plan realistically, in order to reduce the need for routine scheduling adjustments and to forecast more reliably.

(However, when units of effort seem radically out of line with reality, this may be a clue that something in the environment could be better optimized.)

For a quick view of the effort values you are working with, check the **Planning Folder Summary** page.

- The estimated, remaining and actual values are listed for the planning folder as a whole.
- The **Est** column shows the original estimated effort value for each artifact in the folder.
- The **Rem** column shows the value remaining for each artifact after the latest update to the artifact.
- The **Act** column shows the actual effort expended for each artifact, calculated after the work is done.

☞ **Tip:** To make your tracking easier, consider having TeamForge automatically generate a running total of estimated effort for a whole hierarchy of user stories or tasks.

Why is TeamForge Explorer forgetting my settings and not logging me in?

Certain conditions can cause TeamForge Explorer to stop functioning properly and to start behaving oddly. This can manifest itself as the app not logging you into TeamForge, or forgetting your settings, or other general strange behavior.

To correct this behavior, you can delete your stored profile from within TeamForge Explorer.

If deleting the profile doesn't correct the behavior, try deleting the profile again, then uninstall and reinstall TeamForge Explorer.

If that still doesn't fix it, delete the profile, uninstall TeamForge Explorer, and delete the registry key `HKEY_CURRENT_USERSoftwareVA ?SoftwareSourceForge ?ExplorerWindowsMainWindow` (and all subkeys). Then reboot, then reinstall TeamForge Explorer.

What is the file size limit on uploads or attachments in TeamForge?

TeamForge has a 2GB file size limit on uploads or attachments.

If you attempt to upload a file larger than 2GB using a Windows-based browser, you will not be able to upload the file and you may not receive an error message.

How can I import tasks into TeamForge Task Manager?

Tasks can be imported into the Task Manager by using the CollabNet TeamForge Task Management Plugin for Microsoft Project.

Can I associate objects of different projects?

Yes, you can associate an object (for example from document to subversion commit) in one project to an object in another project if you have access or are a Site Admin.

You must have permission to view the object that you're associating with before you can associate with that object, unless you are a Site Admin.

How can I find the number of files in a repository without checking it out?

To list the files in a repository use the command: `svn ls -R file:///svnroot/REPONAME| wc -l` (Requires local access to server)

What happens when log files get too big?

Log files can grow very large over time. To maintain reasonable log file sizes, you can rotate logs on a schedule.

When you rotate logs automatically, live logs are archived every day at 00:00.

Archived logs are stored in compressed form in a directory alongside the live log. For example, if live logs are stored at `<LOG_DIR>/{apps, apache,...}`, then compressed log archives are stored at `<LOG_ARCHIVE_DIR>/{apps, apache,...}`.

The directory structure of the log directory is preserved in the log archive directory.

☞ **Note:** Empty log files are not compressed.

How can I make the project pages invisible for any user who is authorized to see the project?

This can be customized by a Project Administrator.

The Project Administrator can set this using these steps:

1. Click the link 'On' next to Configure, on the top right-hand side of the project home page.
2. Scroll to the page you want to hide and click the Edit Properties icon. This can be found on the right side of the title bar for the page you want to edit, next to Edit or Create buttons. The tooltip displays if you hover over the tool.
3. Choose 'Hidden - Visible only to project administrators in configure mode'.
4. Click Save.

How do I make the monitoring messages be sent from Forge Administrator?

You can change the default behavior for site options by changing the value from "false" to "true" in this statement: # MONITORING_EMAIL_FROM_ADMINISTRATOR=false

If the site option MONITORING_EMAIL_FROM_ADMINISTRATOR=true, then "From:" field is the Forge Administrator, else it is from the user who made the change that initialized the monitoring email.

How can I access the native install version of TeamForge? I can only find the VMWare version.

You will need to contact your account manager; the TeamForge VMWare license does not include access to native install media.

How do I enable/disable path-based permissions via SOAP?

The Path Based Permissions (PBP) are handled via the *roleList, *Cluster methods of rbacAppSoap. To enable PBP, use the "scm_fgp" (fine grained permissions) argument to addCluster.

See the below psudeocode.

```
from com.collabnet.ce.soap50.webservices import *
        from com.collabnet.ce.soap50.webservices.ClientSoapStubFactory
 import getSoapStub
        from com.collabnet.ce.soap50.types import *

        hostname = "http://server/"
        username = "admin"
        password = "admin"
        project = "proj1007"
        roleName = "tracker"

        sfSoap = getSoapStub(cemain.ICollabNetSoap, hostname)
        sfSession = sfSoap.login(username,password)
        rbacAppSoap = getSoapStub(rbac.IRbacAppSoap, hostname)

        roles =
 rbacAppSoap.getRoleList(sfSession,project).getDataRows().tolist()
        roleId = None

        for row in roles:
        if row.getDescription() == roleName :
        roleId = row.getId()
        print "found tracker role, %s" % roleId

        if roleId == None:
```

```
        raise Exception("Cant find role ")

        clusters = rbacAppSoap.listClusters(sfSession,
roleId).getDataRows().tolist()
        for row in clusters:
        print row.getFolderId(), row.getOperationClusterName()
        if row.getOperationClusterName() == "scm_commit":
        print "found target!"
        rbacAppSoap.removeCluster(sfSession, roleId,
row.getOperationClusterName(), row.getFolderId())
        rbacAppSoap.addCluster(sfSession, roleId, "scm_fgp",
row.getFolderId())

        sfSoap.logoff(username,sfSession)
```

How do I enable post-commit logging?

You do this by editing the post-commit.py file.

Edit the `/opt/collabnet/teamforge/runtime/sourceforge_home/integration/post-commit.py file.`

Search for log.setLogging(False) and modify the value from False to True.

How do I put a notice to my users on the project home page?

To do this you will need to create a news item within the project. News items are posted and displayed on the project home page. News items are also displayed on the TeamForge home page.

To post a news item:

1. Navigate to the home page of the project in which you want to post the news item.

   • From within the project, click **Project Home**.
   • From anywhere in TeamForge, choose the project from the list pf projects available under **My Workspace** > **My Projects**.
2. In the Project News section, click **Add**. The **Create News Post** page is displayed.
3. Enter a title for the news item.
4. Enter the text for the news item in the message field (up to 4000 characters including spaces), then click **Add**.

The news item is submitted. It is displayed on the project home page and the TeamForge home page immediately.

What is a private IP address and what are the private IP ranges?

Any IP address that falls specified ranges is a private IP address and is non-routable on the Internet.

These addresses are reserved for use only within private/corporate network and cannot be seen outside the private networks. These private addresses are translated at the company's firewall into an external (public) IP address, which will be some address that does 'not' fall within these ranges.

10.0.0.0/8=10.0.0.0 – 10.255.255.255
192.168.0.0/16=192.168.0.0 – 192.168.255.255
172.16.0.0/12=172.16.0.0 – 172.31.255.255

An address is Private if it starts with:

10 or
192.168 or
172.[16-31]

It is possible for anyone to see their external (public) IP by going to any one of a number of sites that provide this information as a free service. One example that's easy to remember is http://whatismyip.com.

What is the suggested log configuration for a production system?

To troubleshoot installation issues, the default log4j configuration is set to DEBUG. This can cause the log files to become quite large. Once your system is successfully installed and in use, you should drop the log levels down to INFO.

See *Change the logging level on your site* on page 584 for how to do this.

If you still have a problem with over-large log files, you may want to set up log rotation. See *Rotate TeamForge log files* on page 585.

What is the difference between a stagger and normal header in query result heading settings?

The normal and the stagger header preference setting allows you to set a standard header for your query view or a staggered header.

For more information about queries in Issue Tracker see: *Query database of issues*

Can the query result be listed without the issue id?

No. Query results can never be listed without the issue id because issue id is the only field which identifies each issue uniquely.

How do I remove a news item?

You can delete any news item that you no longer want displayed on the project home page. Deleting a news item from a project also deletes it from the TeamForge home page.

To delete a news item:

1. Navigate to the project home page.

   - From within the project, click **Project Home** in the project navigation bar. From anywhere in TeamForge, choose the project from the **Projects** menu in the navigation bar.
2. In the **Project News** section, click **Delete** next to the news item that you want to delete. When prompted, confirm that you want to delete the news item, and it will be deleted.

How do I remove the OCN link from my TeamForge instance?

Remove the OCN link from your TeamForge instance by commenting out the header.ocn file in menu_bar.vm.

Remove the OCN link from your SourcForge instance by editing the `menu_bar.vm` file in the following directory:

```
        $SFHOME/runtime/jboss/server/default/deploy/saturn.ear/sf.war/WEB-
INF/classes/templates/menu_bar.vm file
```

Open the file and comment out or remove the following lines:

```
        <li class="yuimenubaritem">
        <a class="yuimenubaritemlabel" id="openCollabNet" href="/sf/sfmain/
do/ocn">${MESSAGE.get("common", "header.ocn")}</a>
        </li>
```

How do I send email from a specific sender address instead of the member address?

To send mail from a specific sender address change the `MONITORING_EMAIL_FROM_ADMINISTRATOR` token to true in the `site-options.conf` file.

TeamForge uses the member's email address as the sender address when it delivers the mail. To reconfigure this setting in the `site-options.conf` file, set the `MONITORING_EMAIL_FROM_ADMINISTRATOR` token to true, as shown in the following code sample:

```
MONITORING_EMAIL_FROM_ADMINISTRATOR=true
```

Where can I download the TeamForge DTD?

You can download the TeamForge Document Type Definition (DTD) from the link below.

How do I remove the build and test link from TeamForge pages?

The site administrator can remove these links by checking out and editing the branding repository from the look project.

To remove the build and test links from your TeamForge navigation panel, check out the branding repository from look project and reconfigure the links as shown in the following code sample:

☞ **Note:** You must be logged on as administrator to perform this task.

```
        [branding_stage]$ cd branding
        [branding]$ mkdir -p i18n/com/vasoftware/sf/i18n/apps/sfmain
        [branding]$ echo "configure_build_and_test.systemUrl.default=" >
  i18n/com/vasoftware/sf/i18n/apps/sfmain/application.properties
        [branding]$ svn add i18n
        A i18n
        A i18n/com
        A i18n/com/vasoftware
        A i18n/com/vasoftware/sf
        A i18n/com/vasoftware/sf/i18n
        A i18n/com/vasoftware/sf/i18n/apps
        A i18n/com/vasoftware/sf/i18n/apps/sfmain
        A i18n/com/vasoftware/sf/i18n/apps/sfmain/application.properties
        [branding]$ svn commit
        " Old: "enter paragraph information and use html tags for bullet
  points."
```

Can I query the CollabNet TeamForge database?

Yes, you can query the database if you are a site administrator or have been given access to **System Tools** by another site administrator. You can raise a database query by using the **Admin** > **System Tools** > **Ad Hoc Database Query** page.

In the **Ad Hoc Database Query** page, select a data store, Operational Datastore or Datamart, type the "select" query and click **Run Query**. The query is executed against the CollabNet TeamForge database and the results are dispayed.

For security reasons, the `sfuser` and `password_history` tables of the operational datastore are restricted for ad hoc querying from the **Admin** > **System Tools** > **Ad Hoc Database Query** page. Alternatively, use the following views, `sfuser_view` and `password_history_view`, for retrieving all other data but passwords.

☞ **Note:** The results of your query may be limited or your query session may time-out as per the settings in the `site-options.conf` file.

You can submit read-only queries of the format: `SELECT [FROM][WHERE][GROUP BY] [HAVING] [ORDER BY]`.

☞ **Tip:** You can use the following special keywords while drafting the query:

- "\d" or "show tables" - To list all the tables.
- "\d <table name> - To view the description of a specific table.

- "select * from <table name> - To view all the contents of a specific table.

☞ **Note:** This feature is not supported by Oracle at the backend.

Advantages of using the Apache TIKA parser library for indexing

Starting TeamForge 7.0, the underlying parser library for indexing has been changed from Stellent to Apache TIKA.

The Apache TIKA parser library has the following advantages over the Stellent parser library:

Issue	Stellent	Apache TIKA
Stale process issue	Parsing of corrupt or unrecognized files by the Stellent parser libraries often result in stale processes that consume swap space and add to the load on the system, which may at times lead to site outage. To manage such processes, you may choose to create and deploy stale process monitors and the stale processes, when detected, must be removed manually to prevent site outage.	Parsing of unrecognized or corrupt files by Apache TIKA libraries is robust and needs no manual intervention as there are no stale process issues.
Search queue processing speed	It takes five minutes to timeout when the Stellent parser library encounters a corrupt or unrecognized file that it knows not how to parse. If there are more such corrupt or unrecognized files, more time is wasted by the indexer waiting for a response (or a timeout) from the Stellent parser, which in turn adversely impacts the search queue processing speed.	The Apache TIKA parser library is capable of determining whether a file it encounters can be parsed or not. As no time is wasted by the indexer waiting for a response (or a timeout) from the parser, the search queue processing speed is better with the Apache TIKA.
Multiple processes Vs Single JVM	For parsing files, the Stellent parser library spawns one subprocess per file. Meaning, the number of subprocesses is equal to the number of files to be parsed and it is possible that we may end up with the stale process issue as discussed earlier. As a result, if the Stellent processes consume more resources, other processes and applications are left with scarce resources.	The Apache TIKA, being a Java-based parser library, works within the JVM and makes the external resource pool available exclusively for other processes and applications. As the search JVM, where the Apache TIKA library lives, can also be separated starting TeamForge 7.0, it can be managed better.

How do I connect to the Teamforge Postgres database?

You can use the *psql-wrapper* script to connect to the TeamForge application database.

Usage

Run this script as below:

```
sudo [RUNTIME_DIR]/scripts/psql-wrapper
```

How do I capture the output of "top" command?

Top is the realtime monitor of the running processes in a Linux system. To log the top running processes, use the following command: `top -b -n 1`.

-b = Batch mode operation - Starts top in 'Batch mode', which could be useful for sending output from top to other programs or to a file.

-n = Number of iterations limit as: -n number specifies the maximum number of iterations, or frames, top should produce before ending.

How do I use the TeamForge updater to manage backups of old versions of TeamForge?

You can safely delete the items in your <sourceforge_base_install_directory>/sourceforge_home/backups as long as you are comfortable with your version of TeamForge , and have no desire to go back. This directory may also be safely omitted from your backup plan.

How do I resolve timeouts when calling web services?

This is due to the requested operation taking longer then your client SOAP stack is configured to wait before throwing a timeout. You will need to reference your client documentation to see how to update the timeout properties of the connection.

For AXIS in java, you can do this via the sun.net.client.defaultReadTimeout property.

```
System.setProperty("sun.net.client.defaultReadTimeout", "600000"); //10
 minute timeout, in ms
```

What resources can be searched on a TeamForge site?

The resources that show up in search results depend on the context in which you are working.

Here is a summary.

Tool	Searchable resources
Projects	Project id, title/description, created by, project status
Project categories	Title, description
Discussions	• Web UI: author, subject, content, attachment. • Email: we index the original email after we process it. • Topics: author, title and description. • Forums: title, description, author.
Documents	• Document folders: title, description • Documents; version comment, title, status, description, the attachment itself (all versions), authors
SCM	Commit message, title, author
Tracker	• Tracker title, description, author • Artifacts: title, group, category, customer, status, description, authors, tracker, all text flex fields, single-select fields, multi-select fields • Artifact attachments: the attachment itself and comments
News	Body, title, author
File Releases,	• Packages; title, description, author • Releases: title, description, author, maturity, status • Files: description, filename

Tool	Searchable resources
Tasks	Title, description, authors, planned
Users	Username, full name, email, status, details
Pages	• HTML components: title, content • Subpages: page title, component title
Wiki	Content of wiki page, using wiki syntax

Customization

Questions on how to customize the TeamForge site.

Concept definition.

Which arbitrary HTML elements can I customize?

CollabNet TeamForge Snippets are customizable HTML text fragments that can be inserted in predefined locations in CollabNet TeamForge pages.

Snippets can be added in the following locations:

*   Below the page title on all CollabNet TeamForge templates.
*   Above the footer on all CollabNet TeamForge templates.

Snippets get checked into the `templates/html/` directory in the branding repository.

*   The file name for the header looks like this:

        <applicationname>-<actionname>-Top.html
*   The file name for the footer looks like this:

        <applicationname>-<actionname>-Bottom.html

For example, to put your own custom snippet at the top of the Tracker ViewArtifact page, name the file `tracker-viewArtifact-Top.html`.

👉 **Note:** The file names are case-sensitive.

Tip: You can find the application name and the action name in the URL of the page you are customizing. For example, consider a URL such as `http://sf5latest/sf/sfmain/do/createProject`. Here, the application name is 'sfmain" (the main CollabNet TeamForge application) and the action name is 'createProject." If this page had a custom snippet near the header, the file containing the custom snippet would be `templates/html/sfmain-createProject-Top.html`.

What elements of a site can I customize?

You can customize the site home page and the default home page of every project on the site. You can also customize the menu bars, headers and footers of any page.

For more information about branding, see: *Customize anything on your site* on page 290.

How does TeamForge use stylesheets?

The look and feel of much of TeamForge is controlled by cascading style sheets (CSS).

All default CSS styles can be customized to alter the look of the application. You can customize fonts (color, size, font face, etc.), links, backgrounds, headings, tables, tabs, and anything else that CSS can control.

The default TeamForge CSS file is `css/styles_new.css`.

New CSS files can be added to the css/ directory and reference them via `templates/body_header.vm`.

> ☞ **Tip:** If you override an existing CSS file, it will be used instead of the default CSS file. So you must be sure to include all the default styles in your customized file. A best practice is to add any new or overridden styles to the bottom of the CSS file so that they can be easily identified.

Can I substitute my own images for the default TeamForge images?

Every image in TeamForge can be edited or replaced by a new image file.

Images are stored in an `images` directory in the `branding` repository.

> ☞ **Note:** The default TeamForge image files are included in both the Quick Start and Advanced branding zip file.

Examples

- To replace the masthead graphic, replace the image file `images/masthead/logo.gif`
- To replace the folder graphic, replace the image file `images/my/all_projects.gif`

Can I use my own custom JavaScript on my site?

The JavaScript scripts used in TeamForge can be customized.

Existing JavaScript scripts are located in the `js/` directory in the branding repository.

New scripts can be checked into the `js/` directory then referenced from Velocity templates.

Can I customize the web interface?

You can customize the way the web interface looks and functions through the use of Velocity templates.

Please note, however, that while Velocity as a technology is supported by CollabNet, Inc., the customizations themselves are not supportable. Any future upgrade of TeamForge may, in fact, break your customization. Furthermore, making these types of changes to your installation is considered a customization and will impede our ability to support you. Should you experience issues and open a ticket with Technical Support, you may be asked to remove these customization to debug your issue.

Integrated applications

Questions about integrated applications in TeamForge.

Concept definition.

How is an integrated application described?

An integrated application is described using two XML files - a deployment configuration file and an application configuration file - that provide information to TeamForge about the configuration options exposed by the application.

In TeamForge version 6.1.1 and later, you have the ability to configure some integrated application settings using the user interface. You can also export these settings in XML format and make changes. To edit configuration settings, you would upload the XML file containing the updates.

Integrated application settings

> ☞ **Note:** Some of the tags are internationalized so that the application will display languages based on the browser locale. See *Internationalize your integrated application* on page 746 for more information.

<name>

This is the title of the integrated application. When the integrated application is added to a project, the button that appears on the project pages has this name. This name must be unique -- you cannot use it for any other integrated application on the same TeamForge server.

This tag is used in both deployment and application configuration files.

<adminurl>

When an application has an administration screen for configuring its parameters, this field contains that URL. It is optional.

This tag is used in the deployment configuration file.

<baseurl>

This is the URL to which a user will be directed on clicking the integrated application button in a project.

This tag is used in the deployment configuration file.

<endpoint>

This is the SOAP endpoint for the integrated application. The endpoint contains the various methods exposed by the integrated application that are called during the lifecycle of TeamForge.

This tag is used in the deployment configuration file.

<gourl>

This indicates which URL must be used when an object id for an integrated application is specified (either via Jump_to_id or on the URL as /sf/go/<objectid>). This URL can support a couple of dynamic parameters.

- `%o` -- The object id entered by the user will be dynamically replaced here.
- `%p` -- The project id for the object entered will be dynamically replaced here.

For example, if the Go URL is `http://go.tourl.com/tracking?id=%o` and the object ID entered is `XYZ123`, then the URL will be replaced and redirected to `http://go.tourl.com/tracking?id=XYZ123`.

This tag is used in the deployment configuration file.

<category>

This is used to display the integrated application under a separate **Category** tab. It is an optional parameter that is manually inserted based on the requirement. If the category is not defined, the integrated application appears under the **Generic** tab.

This tag is used in the deployment configuration file.

<isbaseapp>

This validates whether the integrated application is a base application configured for the specified category. It is an optional parameter that is manually inserted based on the requirement.

This tag is used in the deployment configuration file.

<is-search-supported>

This validates whether the integrated application needs to be displayed as an object category in the drop-down list for search options in TeamForge such as **Jump to ID** and **Advanced Search**. It is an optional parameter that is manually inserted based on the requirement.

This tag is used in the application configuration file.

<config-parameters>

There can be any number of configuration parameters for an integrated application and they are displayed when associating the application to a project. These parameters are filled in by the project administrator and are available

in the integrated application SOAP interface as configuration parameters. The integrated application gets a chance to validate these parameters and indicate back to TeamForge whether this project association is successful by passing in a "TRUE". It can return a "FALSE" if it doesn't want this project association to succeed. Each configuration parameter is placed inside the "param" tag, which can contain multiple elements to describe the parameter.

<title>	The internationalized title that appears for a project administrator to fill in while associating the integrated application to a project.
<name>	The Java variable under which the value for this parameter will be available on the integrated application.
<description>	The internationalized description that appears when a project administrator fills in or enters a configuration parameter.
<default value>	The default value for the parameter that will appear in the user interface during the association of an integrated application to a project.
<display type>	This is the type of display control used for the configuration parameter. We support "TEXT" for text fields, "CHECKBOX" for checkbox type controls, "RADIO" for radio buttons, and "SELECT" for select dropdowns. This field can also take an attribute that says what the value type for the field should be -- whether it should be an "Integer", "String" and so on. So if the field is expecting numbers, then entering "foo" as a value will throw a validation failure.
<option>	If the display type is "RADIO" or "SELECT", then these fields contain the individual options available for the display controls. This will contain a "name" attribute that will be sent to the integrated application when that option is selected from the UI. The value of this option should be an internationalized field as it is the value visible to the user.
<editable>	This specifies whether the configuration parameter should be editable once the integrated application is associated to a project. These configuration parameters are available when you add or edit an integrated project. If a parameter should not be "edited" post association, setting this to "false" will make it non-editable.

This tag is used in the application configuration file.

<description>
This is an internationalized string for the integrated application's description. It contains information for TeamForge project and site administrators to know what the application does.

This tag is used in the application configuration file.

<id-pattern>
When trying to link to an integrated application id, this regular expression gets used for mapping. By default (if no value is provided), it looks for alphanumerical characters; in case you need specific characters to be matched (for example, JIRA, which has hyphens in ids), this value is used.

This tag is used in the application configuration file.

<page-component>

These settings are used for Project Content Editors. The integrated application content can become part of the standard Page Component data that appears in project home pages. The settings indicate the type of information that will be available from the integration application.

<input-type>	This is the input type control for an integrated app Page Component. We only support 2 types now. Either "select" so that the inputs can be shown from a "SELECT" dropdown and the users will be able to pick a value from there. Else, it can be a "text" where a simple "text" field will be entered for taking the user input.
<result-format>	This is the format in which the output of Page Component is returned. This can be a "list" which indicates that it will be a Table like output. The integrated app will send the results in an XML format and the Integrated app framework converts this into a list of records. The other option is "html", where the output from the Integrated application is just displayed on the screen.
<page-component-description>	The description that will appear when you add an Integrated application Page Component (Link to the page where " Add component" is available)
<page-component-title>	The title that will appear when adding an Integrated application Page Component (Link to the page where "PCE Add component" is available)

This tag is used in the application configuration file.

<permissions>

This is a collection of permissions that are exposed by the integrated application. There could be any number of such permissions. These permissions will appear as a part of the project's roles (existing ones, as well as ones newly added) and can be assigned along with other tool permissions. You can map one of these permissions with a "dapMappedTo" attribute -- this indicates the permission to be used when a user logs in without authentication (for example, for public projects). Typically, this is the permission to read data so that it doesn't need a login name; it varies from one application to another.

This tag is used in the application configuration file.

<prefix>

If the "require-per-project-prefix" attribute is false, the value of this tag is used for identifying the integrated application in Go URLs, associations, and linkifications. If the "require-per-project-prefix" attribute is true, the value is used only for the "Host" project. Each project must fill in its value as part of adding the integrated application. Click here for *steps to add integrated applications to a project*.

This tag is used in the application configuration file. The prefix can contain alpha-numeric characters and cannot be more than six characters in length. For more information about prefix, see *How does an integrated application interact with other TeamForge tools?*

<require-per-project-prefix>

An integrated application can indicate to TeamForge whether the object ids that it generates are uniquely identifiable across the entire application (if yes, the value for the attribute is "false") or whether they need to be project-specific (in this case, the value for the attribute is "true"). If an integrated application needs per-project prefix, you must enter the prefix value when the integrated application is added to a project.

This tag is used in the application configuration file.

<require-scm-integration>

This indicates whether SCM commits need to be vaildated. Some applications might have business rules which indicate that a commit can be made only if certain conditions are met. If the integrated application has any such rules, the value for the attribute should be "true". There are also a couple of methods to be implemented in the SOAP endpoint.

This tag is used in the application configuration file.

<require-page-component>

Some integrated applications choose not to expose details as Page Components. For those that don't, set this tag to "false" and for those that do, set it to "true". If the value is "true", you must provide the "page-component-details" tags as well.

This tag is used in the application configuration file.

<servicetype>

TeamForge 6.1.0 and earlier releases supported only SOAP as the mechanism to talk from TeamForge to the integrated application. TeamForge 6.1.1 and later support REST calls. The servicetype tag indicates whether the protocol used for communication is REST or SOAP.

This tag is used in the deployment configuration file.

For examples of how these tags are used in the integration of the Pebble bogging application, see *pebble-dep.xml* and *pebble-app.xml*.

How does an integrated application interact with other TeamForge tools?

When you integrate an external application into your TeamForge site, the application can take full advantage of object IDs, links and Go URLs.

To look at how this works, we'll use the Pebble application as an example. Pebble is a blogging tool that you can quickly integrate with TeamForge.

Object IDs

Integrated application object IDs are of the form "`prefix_objectId`". Object IDs uniquely identify a TeamForge object so that you can access and use it in different contexts. For example, to get to artifact `artf1234` quickly, you just enter `artf1234` in the Jump To ID box. In the Pebble tutorial application, the date of a blog post, in YYYYMMDD format, is used as the object ID.

A prefix is an alphanumeric string attached to the beginning of an object ID that TeamForge uses to manage object IDs from different tools. For example, in the Pebble app, `<prefix>_20100601` gets you a page showing all the blog posts in the project that were published on June 1, 2010.

In an object ID such as "`prefix_objectId`", the "`prefix`" is case-insensitive, whereas the "`objectId`" is case-sensitive. For example, the two object IDs, "`PT_SC1`" and "`pt_SC1`", refer to the same object in TeamForge. Whereas, the two object IDs, "`PT_SC1`" and "`PT_sc1`", refer to two different objects in TeamForge. Here, `PT` and `pt` are case-insensitive and the `SC1` and `sc1` are case-sensitive.

The prefix can either be the one specified when an integrated application is added to a project by project administrator, or the one in the XML Application configuration file depending on the "require-per-project-prefix" setting. The "require-per-project-prefix" setting can be true or false. If it is false, each project integration would not need to provide a project prefix; so the one provided in the XML application configuration file takes effect. If the "require-per-project-prefix" setting is true, a prefix needs to be provided by the user during every project association.

The amount of information the prefix carries depends on the kind of application you are integrating into your TeamForge site.

- With applications that use object IDs, such as Project Tracker and JIRA, you can identify the project that the object belongs to from its object ID.

- For applications that don't have uniquely identified objects, or don't have the notion of "project," such as MoinMoin or Review Board, you can choose a prefix that's specific to the project where the integrated tool is used.

Setting up multiple prefixes for integrated applications

At times, you may want to use more than one prefix for an integrated application such as the TeamForge Orchestrate. It is possible to have multiple prefixes set up for integrated applications. You must have the prefixes, separated by commas, included in the XML application configuration file and upload the file to your TeamForge site. For more information about uploading the application configuration file, see *Edit an integrated application*.

Consider the following while setting up multiple prefixes for an integrated application:

- Prefixes, once set up using the XML application configuration file, cannot be modified.
- A prefix can be up to six alpha-numeric characters in length. However, the combined length of all the prefixes cannot exceed 128 alpha-numeric characters.
- The "require-per-project-prefix" must be set to false in the application configuration file. In case it is set to true, an error message appears when you upload the application configuration file.
- Do not use existing prefixes. You cannot upload an application configuration file consisting of one or more prefixes already in use in TeamForge.

Go URLs

Go URLs allow a user to get to a particular object ID with a short, handy URL. To use this for Pebble, construct a URL like this: `https://mysite.com/sf/go/<prefix>_<date in format YYYYMMDD>`.

For example, if the Pebble tool in your project has the prefix `PA`, and you want to send someone all the blog posts published on app June 1, 2010, send them this link: `https://mysite.com/sf/go/PA_20100601`.

Associations

The object ID can be used to associate objects with other TeamForge objects. For example, if you want to associate a document with the blogs published on June 1, 2010, go to the document's **Associations** tab and add an association to `PA_20100601` as the object ID.

Automatic links

When you type text of the format `<prefix>_<date in YYYYMMDD>` in any TeamForge text field, the text is converted to a link. When you click the link you see the blog posts for that date, if any.

SCM commits in TeamForge with Orchestrate integration

Consider the following while integrating TeamForge and Orchestrate.

Uploading the event handler JAR file

After adding *Orchestrate* to your site, it is mandatory to upload the custom event handler JAR file to get the pre and post commit notifications. *Click here* for steps to add a custom event handler.

Implications of TeamForge-Orchestrate integration on SCM commits and associations

Pre and post commit notifications will be sent from TeamForge to Orchestrate only if the "*Association Required on Commit*" is enabled for the repository and if the "*require-scm-integration*" is set to "true" for the Orchestrate application.

Post TeamForge-Orchestrate integration, SCM commits will fail in TeamForge if all the following conditions are true and if the Orchestrate application is down or if there are errors while processing the commit request (on the Orchestrate's side):

- The **Association Required on Commit** is enabled for a repository.
- The **require-scm-integration** is set to "true".

- The commit message contains the integrated application's ID. In other words, the Orchestrate application's ID in this case.

SCM commits can also fail if the custom event handler exists in the TeamForge even after the removal of the Orchestrate application from the TeamForge site. It is highly recommended to remove the custom event handler once you remove the Orchestrate application.

What is in the integrated application's configuration file?

Besides the basic information regarding the application, you can specify several important parameters in the configuration file.

The xml file can contain the following information about the integrated application:

- name - Name of the integrated application.
- description - Description of the integrated application.
- baseurl - Location where the integrated application resource locator is expected to go, when invoked.
- permissions - Set of permissions associated with the integrated application.
- prefix - The prefix identified for the integrated application.
- gourl - The Go URL of the integrated application.
- endpoint - The URL that contains the webservice descriptor language(WSDL).
- adminurl - The URL that provides integrated application administrator privileges through TeamForge site administration.
- Category - This is used to display the integrated application under a separate **Category** tab.
- isbaseapp - To validate whether the integrated application is a base application configured for the specified category.
- is-search-supported - To list the integrated application as an object category under the TeamForge search options, **Jump to ID** and **Advanced Search**.
- require_prefix_project - Confirmation if the integrated application requires a per-project prefix.
- require_scm_integration - The indication for the source code integration between TeamForge and the integrated application.
- Configuration Parameters - The parameters that the integrated application needs for initializing itself.
- editable - Confirmation if the value specified in the configuration file is editable using the **Edit Integrated Application** page.
- servicetype - The servicetype tag indicates whether the protocol used for communication is REST or SOAP. The protocal could either be REST or SOAP.

You can specify permissions for integrated applications and map the same to all the roles, as required.

☞ **Note:** The associations between CollabNet TeamForge and an integrated application can be created only from CollabNet TeamForge to the integrated application and not vice-versa.

☞ **Note:** To associate an object in an integrated application from within CollabNet TeamForge, use the `[<prefix_objectid>]` format. Successful associations appear hyperlinked.

☞ **Tip:** Each integrated application displays its prefix on moving the mouse over the application name in the tool bar.

Lab Management

Use this background knowledge to help you support TeamForge Lab Management projects.

Concept definition.

Can I get more systems for my project?

If your project needs more systems allocated to it, you must contact your Domain Administrator to increase your limit and allocate new systems to your project.

Your Domain Administrator will usually be a known support contact in your organization.

If additional systems are available in other projects, your Domain Administrator will be able to re-allocate those resources to you. If there are no free systems available, and none that can be reassigned, the Domain Administrator will be able to request additional systems from CollabNet, and then assign them to your project.

What happens when I move a machine between projects?

When you move a machine from one project to another, the machine is not automatically rebuilt. It still runs the same profile and all the same software it was running in the original project.

There can be reasons for wanting to move a machine between projects without rebuilding it. For example, due to a reorganization, the same users and hosts may be providing the same service, but under a different department or project.

Moving a machine does not reboot the machine or stop any running processes.

If the old project was running processes on the machine that you do not want, you will have to stop them manually.

Moving a machine changes the access rights on the machine, but does not end existing login sessions.

Once the machine has changed projects, after a few minutes, the authentication will have changed so that users from the old project can no longer log in and users from the new project can now log in.

Existing user login sessions, however, are not terminated. Those users will not have `root` (UNIX) or Administrator (Windows) access any more.

Moving a machine does not reset the contents of the `localadm` group.

The `localadm` group is used to maintain a list of users on the system who are local administrators. This group is not emptied when a machine moves projects. It could contain users which you do not want as administrators on the host in the new project.

What is an audit log?

Every action performed by the user in the TeamForge Lab Management system is recorded in the Audit Log.

For example, when a host is rebuilt using a profile, these are some of the details captured in the Audit log:

- The old profile.
- The new profile.
- How long it took to complete the rebuild process.

What is the correct procedure for modifying a hosted Lab Manager profile?

All profile modifications must be done through the Lab Management UI, under Administration > Manage Profiles. Lab Manager profiles should not be directly modified and changes should not be committed to subversion.

To modify your profile, follow these steps:

1. In the browser, login to https://mgr.cubit.domain.com/
2. Click on Administration
3. In the left pane, click on Manage Profiles
4. Click on the profile (your_profile_name)
5. Click on the Packages tab and choose your options

What is port forwarding?

Use port forwarding to let TeamForge Lab Management hosts connect to machines on other networks.

While TeamForge Lab Management is a secure and isolated environment, occasionally there are valid reasons to let other traffic in and out of TeamForge Lab Management.

Port forwarding consists of routing traffic from one network port on one host to another (same or different) network port on another host.

Take great care when exposing network services running on TeamForge Lab Management hosts to the outside world. Acquaint yourself with your organization's security policies, or develop them if you don't already have them, and make sure all services that are exposed comply with these policies. Even better, work with CollabNet to establish access controls around the access to TeamForge Lab Management, so only authorized hosts from within your enterprise -- and trusted partners -- can access TeamForge Lab Management.

☞ **Important:** CollabNet strongly recommends that any service that can yield a shell account on a TeamForge Lab Management node -- such as `SSH` -- not ever be port forwarded to outside the TeamForge Lab Management environment using this interface.

For CollabNet Hosted customers: CollabNet performs regular scans of its network, and if we see a dangerous or vulnerable network service configured, we may take any steps necessary to protect the overall security of the TeamForge Lab Management customer environment, including disabling the offending network service and the associated port forwarding.

What is involved in administrating profiles?

If you are a project admin in TeamForge Lab Management with profiles assigned to your project, or if you are a TeamForge Lab Management domain admin, it is important that you have an understanding of how to administer profiles.

☞ **Important:** It is essential that you first have a solid understanding of how to retrieve and interpret profile details. Making changes to profiles without understanding the effects of what you are doing can cause disruption to your own project, and if the profile you are administering is a public profile, possibly many other projects that use this profile as well.

Profile definitions are versioned, which means the history of changes to profiles is preserved. This allows users to build with various versions of a profile, go back to earlier ones if a new one doesn't work for them, and upgrade to a newer version when available. As an administrator, you have the power to set descriptions on profiles and individual versions, block the use of certain versions, and so on.

Within the Profile Library, you can reach the Profile Admin page by selecting a profile in the Profile Library, and clicking on the **Admin** tab.

Properties can be version-independent or version-dependent.

- Version-independent properties are common across all profile versions -- they do not vary, regardless of the profile's version. These are basically the settings that are configured when the profile is first being added to Lab Management with profiles assigned to your project, or if you are a TeamForge Lab Management. They are displayed at the top of the Profile Admin screen.
- Version-dependent properties are associated with a particular version of a profile.

  ☞ **Note:** These properties, when set, will apply to the version in question, as well as to any subsequent version. Setting the property on a later version will overwrite the property on earlier versions.

  These properties are displayed below the version-independent properties on the Profile Admin screen. Most of the properties that you can edit are version-dependent properties.

Only one version's properties are displayed at any given time for editing. To view multiple versions at the same time, use the Profile Details page.

Version-independent properties

☞ **Note:** Some version-independent properties, set at profile creation time, cannot be changed.

There are three version-independent properties that you can change:

Summary	This is a brief summary of the profile. You can enter a more detailed description for each version in the properties section.
Project	The project that the profile belongs to. If you select "no project" using the "-" option, the profile will belong to the domain. The **Project** selection box is only available to domain admins.
Is Public?	This setting governs whether the profile is usable by all profiles, or just to members of this project.

Version-dependent properties

There are many version-dependent properties that you can change for each profile. Some of these properties include:

- Those that specify hardware requirements needed to build this profile. This will make sure a profile is assigned to a host that can actually successfully be rebuilt with the profile.
- An option to specify whether the profile version can be used or not. Useful for marking a profile version "bad" so it cannot be inadvertently selected by users.
- The logo to associate with the profile version. This logo will be displayed in many locations throughout the system to easily identify profiles. Use the link to the Profile Logo Gallery to add and maintain logos.

☞ **Note:** When working with these properties, be sure you are operating on the correct version!

Description	A description for this version of the profile. Try to put some informative text here, so potential users of this profile will have some guidance as to what this version contains, or how it is different from other versions.
Tag	A symbolic name for a profile version. Tags can be used to make versions easier to remember, and can be moved around between profile versions, similar to a "tag" in Subversion or CVS. Valid characters for Tags are: letters A-Z, numbers 0-9, and underscores ; although a profile cannot be all numeric, and must contain at least one non-numeric character. The Tag HEAD is reserved, and always refers to the latest version -- buildable or not -- of the profile.
	Tags are commonly used by project admins and other project leaders to instruct their users about the proper versions of profiles to use.
Can new systems be built with this version of the profile? (also known as "Buildable").	For any number of reasons, you may wish to restrict profiles so that one or more versions of that profile are not buildable. For example, you may wish to force your users to always use the latest version of your profile: this would be easily accomplished by making all the profile versions not buildable except the most recent. You can change the buildability of a version at any time.
Icon	You can choose from any of the available icons for your profile, although a profile icon is strictly optional. Icons are not private to your project, and are shared among the whole domain, so do not upload anything too secret (or naughty!).

Buildable CPU Types	The types of CPU that can be used to build the profile, for example, "Xeon" or "UltraSparc IV". Setting this property is strictly optional, even if the profile has CPU type restrictions.
Buildable CPU Archs	The CPU architectures that can be used to build the profile, for example, "x86_64" or "sun4v". Setting this property is strictly optional, even if the profile has CPU architecture restrictions.
Buildable Number of CPUs	The number of CPU's required to run the profile. Setting this property is strictly optional, even if the profile has restrictions around the number of CPU's it can use.
Buildable Hardware Models	Specific hardware models which are required to run the profile, for example, "PDP-11". Setting this property is strictly optional, even if the profile has hardware model restrictions.
Size (in GB)	The minimum hard disk size, in gigabytes (GB), required to install and run the profile.

Who controls which profiles can be used in a project?

As a project administrator, you can control which operating system profiles the users in your project can build hosts with.

TeamForge Lab Management's Profile Library gives ownership of profiles to individual projects, or to the entire domain. Profiles can either be public (available to all projects on the site) or private (available only to the project which owns the profile).

However, that fact that a profile is public, or belongs to the project, does not necessarily mean that it can be used to rebuild hosts. The profile must be specifically allowed for use in the project by a project administrator before it can be used to rebuild hosts.

Sometimes it is desirable not to restrict an entire profile, but only one or more versions of that profile. Individual versions of a profile can be prohibited by the project administrator of the project which owns the profile using the Profile Admin screen.

If the profile does not belong to your project, you cannot restrict the use of that profile at the individual version level: that decision is made at the discretion of the owners of the profile.

If you are the project administrator of the project which owns a profile, you can change the profile's public/private status at any time using the Profile Admin screen.

When you prohibit a profile from your project, this has no effect on hosts which are already built inside your project. They will continue to function normally running the profile they were already running. The owner of the system, however, will be unable to rebuild the system with its current profile, or any other profile which is not allowed in the project.

As a project administrator, you may wish to force your users to rebuild their systems once you have prohibited the use of a profile. On the Profile Summary page you can find a listing of all systems that are running each profile. This helps you track down systems running a profile that you have prohibited, and with your project administrator privileges, you can rebuild those systems with a profile of your choice.

How are my project systems being utilized?

Regardless of the number of hosts in a project, in practice it is common to find shortages of free hosts. At the same time, there are almost always hosts which are under-utilized or even completely idle, which could be re-allocated or consolidated.

Finding and reallocating these hosts allows more efficient use of your project infrastructure.

Project-Level Analytics

The **Analytics** tab in your project shows the following metrics. These metrics are the base available across all operating systems that TeamForge Lab Management supports.

- Load Average
- Processes
- CPU
- Logged-in Users

☞ **Note:** More metrics, including host-specific and operating-system-specific metrics, are available for individual hosts when you click on the host name in the results table.

To see a metric, click the name of the metric. You can see these time ranges:

Daily	Approximately the past 24 hours
Weekly	Approximately the past 7 days
Monthly	Approximately the last 30 days
Yearly	Approximately the last 365 days

If a given host says "no value" in any of its columns, this means that TeamForge Lab Management has been unable to collect this data over the requested time interval. If the machine is up, and not collecting data, contact a TeamForge Lab Management administrator to investigate why data collection is not working properly. If the machine is currently down, or was down during the requested time range, you will not be able to get any performance data for those times. There is no way to retroactively "catch up" data if collection is not working properly.

The data used to build each graph and chart presented to you can be exported in CSV (Comma-Separated Value) format, suitable for opening in any spreadsheet application. This allows you to build your own visualizations of the data to complement the ones TeamForge Lab Management creates.

Beating The System: Dealing With the Possibility of Users Generating "Fake" Load To Make Machines Seem Busier

Since we publish the metrics for determining how busy TeamForge Lab Management thinks machines are, it is possible for irresponsible users to generate automated jobs which simulate a busy machine, even if the machine is really not being used for anything.

We encourage all users and administrators of the project to make sure people in your projects understand that this type of behavior is not acceptable and may lead to loss of privileges or other actions against them. Presumably, if someone has a good reason for wanting to keep a machine, it is good for their project, and your organization if they do so.

As administrators, we strongly recommend you take the time to understand and listen to your users' concerns about how machines are allocated. You may just need more machines in your project, or you may be overzealous about de-allocating machines once they drop below a certain usage threshold. De-allocating machines that seem to be not busy may seem efficient, but if those machines took their users a long time to set up, that might be counterproductive. Perhaps you can reach a middle ground and use virtual machines, or a smaller virtual or physical machine, for that user.

Finally, we always recommend talking to your users and trying to understand how they are really using their machines. The statistics that TeamForge Lab Management gathers are a starting point for managing your software development and testing infrastructure, but not the final word.

What is host URL mapping?

Host URL Mapping in TeamForge Lab Management allows you to access web services running inside the TeamForge Lab Management environment from anywhere, using a simple and consistent URL, with optional SSL encryption services added on.

Host URL Mapping provides three major benefits:

- Maintains a consistent URL even if the service is moved to a different TeamForge Lab Management node or different base URL.
- Provides external access to resources that would otherwise be only accessible inside the TeamForge Lab Management environment. Because Host URL Mapping uses standard HTTP/HTTPS ports, you will not be blocked by firewalls that sometimes prevent port forwarding from working properly.
- Allows you to transparently add SSL encryption to web services running inside your TeamForge Lab Management environment.

If your application mixes absolute and relative URL's, host URL mapping dynamically rewrites the absolute portions of your URLs to help ensure that your applications display properly when they are mapped. While not perfect, this feature has been tested with a number of web applications and found to be effective.

> ⚠️ **Caution:** Take the utmost care when exposing web services running on TeamForge Lab Management hosts to the outside world. Acquaint yourself with your organization's security policies, or develop them if you don't already have one, and make sure that all services that are exposed comply with these policies. Even better, work with CollabNet to establish access controls around the access to TeamForge Lab Management, so only authorized hosts from within your enterprise -- and trusted partners -- can access TeamForge Lab Management.

CollabNet strongly recommends that any web service that you expose be password-protected, or otherwise require authentication to access.

CollabNet Hosted customers: CollabNet performs regular scans of its network, and if we see a dangerous or vulnerable web service configured, we may take any steps necessary to protect the overall security of the TeamForge Lab Management customer environment, including disabling the offending service and the associated URL mapping.

> 👉 **Note:** While you can convert non-SSL URL's into SSL using Host URL Mapping, you cannot map SSL URL's. You can use *Port Forwarding* to expose SSL URL's outside of the TeamForge Lab Management environment, however.

Why doesn't URL mapping work for me?

My page doesn't look right. Images and graphics are wrong, or functionality doesn't work. The same page looks and works fine when viewed through Port Forwarding.

Host URL Mapper attempts to "clean up" HTML pages which pass through it in order to clean up absolute links. But it is not hard to construct an application that will slip through TeamForge Lab Management's filters and still not properly render all of its referenced objects inside of its pages. We make all reasonable attempts to clean up HTML, but not all applications can be properly rendered using Host URL Mapping. This is especially true for applications which make heavy use of Javascript, ActiveX, Java applets, and other types of rich client-side web programming.

To test this, use either a direct connection to the host (if available) or a port forwarded connection to the host to see if this behavior is present on the original version of the page. If the original page does not have this behavior, the first step is to verify the **Dynamic Rewriting** level is set to **More Aggressive**. If that does not work, please file a support request with CollabNet to evaluate the page.

How does host URL mapping compare with port forwarding?

URL mapping is good if you don't want your connection blocked, while port forwarding is good if you need non-HTTP services.

- Port forwarding is a facility for making any TCP or UDP network service available outside the TeamForge Lab Management environment. Host URL mapping only allows you to expose HTTP-based services.
- A major limitation of port forwarding is that many organizations' firewall security policies prohibit outgoing connections to arbitrary high ports. Because URL mappings all use standard HTTP/HTTPS ports 80 and 443, they will never be blocked. If you can access TeamForge Lab Management itelf, you will be able to access any application configured with host URL mapping.
- Host URL mapping can automatically add SSL encryption to your web services that are not running SSL encryption. With port forwarding, if you want SSL encryption, you must set it up on each host.
- Using host URL mapping, you can expose only a part of a server's URL space. With port forwarding, the entire URL space is visible.

- Port forwarding does not need to rewrite links inside the HTML, so more web applications will work under port forwarding.

How are virtual guests different from physical machines?

Virtual guests work like physical hosts, with some important differences.

Creating a new virtual guest

The process of creating a new virtual guest is very similar to the process of creating a new physical host, with the following differences:

- You must select a virtual host -- a physical machine -- to run the virtual guest on. You cannot run a virtual guest inside of another virtual guest.
- The virtual guest's project cannot be set independently of the virtual host, and the virtual guest will always be in the same project as its virtual host.
- You are constrained, with hard limits, by the RAM and hard disk available on the virtual host. TeamForge Lab Management requires that each host have a minimum of 512MB of free RAM and 10GB of disk free.
- Even if you are within the hard limits for RAM and hard disk space usage, you are still sharing other resources -- notably disk I/O bandwidth, network bandwidth, and CPU cycles -- with the host machine and any other guests already on the virtual host. Before creating a new virtual guest on a virtual host, we recommend carefully examining the virtual host's system performance to make sure it can handle the additional load. Of course, if you do find out later on that performance of your virtual guest is not as good as you would like, you can always migrate the virtual guest to another virtual host.
- You are not as constrained on your selection of MAC address with virtual guests ; you can choose any available address in the allowed range. Valid values for MAC addresses for virtual guests in TeamForge Lab Management are 00:50:56:01:00:01 to 00:50:56:3F:FF:FF. Using anything outside of this range will either result in the host not being reachable on the network, or the host coming in conflict with another MAC address on the network.
- You cannot specify the architecture or chip type for the virtual guest, since those properties are inherited from the parent.
- You do not need to specify a Lights Out Management IP address for the virtual guest, since the IP address used to manage the guest is always the IP address of the virtual host.

Disk size

While disk space is allocated to virtual guests at the time of virtual guest creation, it is not actually occupied on the host until it is needed by the guest. At the same time, TeamForge Lab Management does not keep an up-to-date count of exactly how much disk space is in use on the virtual host.

In practice, this means:

- You will likely have more disk space on your host than your virtual guests would indicate. But it is also possible to have *less* space than your virtual guests would indicate. For example, let's say you have a 100GB disk on your host, with two virtual guests, each with a 40GB disk allocated. But if you're only using 5GB of that 40GB in each guest, the remaining 70GB in unallocated. But on the flip side, let's say you allocated those two virtual guests at 10GB each, but they were using a total of 90GB on your local disk to store files. TeamForge Lab Management would let you make this allocation, but your virtual machines would crash when you tried to put more than a combined 10GB on them both.
- This translates into freedom in your management of disk space on the virtual hosts: you are free to temporarily "borrow" disk space from virtual guests that is not being currently used and use it for your virtual host.
- Along with this freedom comes the responsibility of being vigilant about maintaining sufficient space on your virtual hosts to always have enough disk space for any guests on the system. Monitor your usage carefully using Analytics page for the host.

Modifying hardware parameters of existing virtual guests

Some virtual guest hardware parameters can be modified after the virtual guest has been created.

Changing disk size	While changing the size of the disk is supported in TeamForge Lab Management, the change will not be reflected until you re-image your guest. And, in certain cases, it is possible for a change in the disk size of your guest to cause the guest to become unreachable, especially if you reduce the disk size. Therefore, we recommend that you change disk size on virtual guest only if you are going to immediately rebuild the guest.
Changing RAM size	Changing RAM size is a very low-risk operation. In order for the change to take effect, the virtual guest must be rebooted after the change is made in TeamForge Lab Management. Reducing the RAM available to your virtual guest can make the system run much slower.
	☞ **Note:** In between changing RAM size of the virtual guest and rebooting the virtual guest, you must wait approximately 5 minutes to insure your change is propagated.
Changing the number of CPUs	Virtual guests can support either one or two processors. Two CPUs for virtual guests are supported only if the virtual host has at least two CPUs. Like changing a virtual guest's RAM allocation, changing the number of CPUs in a virtual guest is a low-risk operation, and the virtual guest must be rebooted after the change has been made in TeamForge Lab Management.
	☞ **Note:** In between changing number of CPU's for the virtual guest and rebooting the virtual guest, you must wait approximately 5 minutes to insure your change is propagated.

What is involved in migrating a virtual guest?

When you migrate a virtual guest to a different host, there are a few things to keep in mind.

- The user performing the migration must be a TeamForge Lab Management Domain Admin.
- The current virtual host and guest, and the desired new virtual host, must all be in the same project. In other words, you cannot change both the project allocation and the parentage of a host in a single step.
- The target virtual host must have enough RAM, disk, and CPU to accommodate the virtual guest you wish to migrate to it. The rules governing the RAM, disk and CPU needed are the same as if you were creating a new virtual guest on the host.

  TeamForge Lab Management calculates these values for you and presents you with a drop-down list of hosts that fulfill all these criteria. But if you are wondering why a host is not a valid potential new virtual host, see the host's current hardware parameters and allocations on the **Host|Admin** page for the virtual host.
- If your TeamForge Lab Management installation is composed of more than one subnet, both the virtual guest being moved and the target virtual host must be on the same subnet. If you are not sure if your installation has more than one subnet, check with your local TeamForge Lab Management support contact.
- Only one virtual guest at a time can be migrated to a given virtual host, although there is no limit on the number of virtual guests that can be moved from a given virtual host. The reason for this restriction is that moving a virtual guest to a host is a very I/O-intensive operation that will consume all available disk I/O on the destination host. Allowing more than one simultaneous move will not be any faster, and increases the risk of over system instability for the virtual host.
- If more than one virtual guest migration to a given virtual host is started at the same time, they will complete serially. The exact order of the migrations may not correspond exactly to the order in which they were initiated and cannot be accurately predicted.

• Occasionally, after the migration has completed successfully, the virtual guest will require an extra reboot to start up properly.

What's a good way to read log information?

To read log entries, download the log as a CSV file.

You can also filter the logs using the **Filter** option at the top of all audit log screens.

On any log page, you can click **Download all log entries in csv format** to download the corresponding CSV file.

👉 **Note:** The CSV file is formatted in Microsoft Excel format. If you import it into OpenOffice, set **Text Delimiter** to " (double quotes).

Email

Questions about email settings in TeamForge.

Concept definition.

Can I customize my site's email notifications?

The default text for CollabNet TeamForge email notifications can be customized.

Email templates are located in the `templates/mail` directory in the branding repository.

Some email templates that are commonly customized are:

• `user_welcome.vm`
• `account_request_rejection.vm`
• `project_approve_pending.vm`
• `user_forgot_password.vm`

Email template formatting follows the standard Velocity conventions:

```
user_welcome.vm
##subject
Welcome to CollabNet TeamForge  5.0!
##subject
##body
The Welcome message goes here...
##body
Email subject line
Email message text
```

Can I specify an alternate email address?

Yes, of course you can specify one or more alternate email address. CollabNet TeamForge supports user profiles with one primary email address and up to three secondary email addresses.

The email address specified while creation of a user account is considered the primary email address. The alternate email addresses are optional and can be specified while updating the user profile.

👉 **Tip:** To add your secondary email addresses, use the **Admin** > **Users** page to update your profile.

How can I check if port 25 is open?

If you know the mail server is up and running, check whether you can talk over port 25 to your mail server. This can be done using a one-line command: telnet <appserver name> 25 Substitute the <appserver name> with your own server.

Once you type this into your DOS window and hit return, you should see some sort of response from your mail server, as shown below:

Trying 208.75.196.84... Connected to cu190.cubit.sp.collab.net (208.75.196.84). Escape character is '^]'. 220 cu190.cubit.sp.collab.net SMTP Server (JAMES SMTP Server 2.2.0) ready Mon, 27 Jul 2009 06:38:20 -0700 (PDT)

When a discussion forum is set up, do all members receive a notification mail?

Yes. A mail informing users about the creation of discussion forum is sent to all the members of the project who have the **Discussions** (check box) selected as a monitored application.

Users can enable **Discussions** as a monitored application. To do that:

1. Go to **My Workspace > My Page > Monitoring** and select a project from the **Edit Monitoring Subscriptions and Preferences** pane.
2. Select the **Monitored Applications** tab.
3. Select the **Discussions** check box and click **Save**.

For more information about discussion forums, see *Administer forums and mailing lists* on page 68.

Does the project owner receive a mail when a discussion forum is started in his project?

Yes. Whenever a discussion forum is started the project owner gets an email.

I am unable to edit a specific artifact via email, but I can via the web UI. Why is this?

There may be work flow rules applied to the tracker that require specific fields to be set. As you can only define the artifact title and description via email, the artifact creation fails. If you wish to create artifacts in this tracker via email, the tracker admin will need to disable these work flow rules.

In addition, you may not be able to edit artifacts belonging to a tracker via email, if the tracker has two mandatory flex fields with parent-child relationship, but no default value relationship between the default values of the parent and child flex fields. In such cases, it is recommended to always establish the default value relationship between flex fields that have a parent-child relationship.

How do I set up a local alias via James?

In situations where you need to obtain a SSL certificate for your domain, and your SSL certificate provider only permits you to use addresses related to your TeamForge domain, it may be necessary to generate an email alias from within TeamForge. Since there is currently no way to do this through the UI, you'll have to do it from the James administrative interface.

First, you'll need to connect to the James administrative interface on your system. If you've followed our best practices guide in our knowledgebase, you'll know that you should have port 4555 firewalled to everyone but localhost. SSH to your TeamForge server, and then issue the following command:

```
telnet localhost 4555
```

This will bring up the Remote Administration Tool:

```
[root@app1 root]# telnet localhost 4555
Trying 127.0.0.1... Connected to localhost (127.0.0.1).
Escape character is '^]'. JAMES Remote Administration Tool 2.2.0
Please enter your login and password
Login id: admin Password: (text is echoed locally)
Welcome admin. HELP for a list of commands
```

First, we'll need to add a new user:

```
adduser <username> <password>
```

Then, we'll need to set the forwarding address of that user

```
setforwarding <username> <email address where you want email to go>
```

finally, we'll exit the James administrative interface.

```
quit
```

Your changes should be in place.

Does TeamForge support using /etc/aliases for local mail delivery?

No, TeamForge uses the James SMTP server, which does not use the /etc/aliases file.

To enable local mail aliases, you will need to configure user mapping in the XMLVirtualUserTable in the `/opt/collabnet/teamforge/runtime/james/apps/james/SAR-INF/config.xml` file.

> ☞ **Note:** Please note that while the James SMTP server is used as part of TeamForge, customizations such as these cannot be supported by CollabNet.

How can I stay informed about events on my CollabNet TeamForge site?

To keep up with changes, you can be notified by an automatic email when an item you are interested in is updated.

Monitoring items lets you stay up to date automatically on all changes without having to log into CollabNet TeamForge and check the status of each item.

You can monitor individual items, folders, and entire applications in each project for which you are a member. You can configure the frequency of email notifications to suit your personal preferences, and suspend monitoring messages entirely if you are out of the office or for any reason do not want to receive messages.

Items that can be monitored include:

- Entire applications, such as all tasks or all documents.
- Folders, such as task folders or document folders.
- All items in a tracker, forum, or package.
- Individual items, such as a task, a document, a tracker artifact, or a release.

You can tell by looking at each item whether you are currently monitoring it and how to begin or end monitoring it.

- For each item, list of items, or folder, you see a **Monitor** button or menu option.
- If you are currently monitoring an item, the **Monitor** option is replaced by a **Stop Monitoring** option, and the monitoring icon is displayed.

Whenever you create an item or edit an item, you automatically begin monitoring that item.

You do not receive monitoring notifications for your own changes to a monitored item.

If you have the appropriate permissions, you can see who is monitoring an item.

You can add other users to the monitored item. After a user is added to a monitored item, the user can continue monitoring the item, configure monitoring preferences for the item, or stop monitoring the item.

How can I reduce the number of emails from monitoring a folder?

Monitoring a folder might spam the mailboxes with emails that are generated for every small change.

Email notification preferences under the **Monitoring** sub-tab under My Workspace can be modified to Send Daily Digest Email, which provides a summary of the email alerts.

1. Go to **Projects > My Workspace**.
2. Select **Monitoring** from the **My Page** menu.
3. In **Edit Monitoring Subscriptions and Preferences**, click All Projects and click on **Email notification preference**.

4. For each project, the monitoring folders under various components (Tracker/File Releases/Tasks/Discussions/ Wiki) can be customized to Daily Digest Email.

Can TeamForge accept email for more than one domain?

You can configure James to accept email for more than one domain by adding the additional domains to the <servernames> section in the config.xml file.

Add the domains to the <servernames> section of this file: `/opt/collabnet/teamforge/james/ james-2.2.0/apps/james/SAR-INF/config.xml`.

Around line 53, you should see the following:

```
<servernames autodetect=""true"" autodetectIP=""true"">
    <servername>localhost</servername>
</servernames>
```

You can add other host names for James to accept mail for by adding more <servername> blocks. The comments in the config.xml file explain this further. Please keep in mind that these changes may be overwritten by future TeamForge upgrades.

How do I configure TeamForge to send mail on a specific network adapter in a multi-NIC configuration?

When a host has multiple NICs, James will try to do the right thing when sending mail. In some network setups, this is not correct, and manual configuration is needed.

James requires multiple changes to fully configure how it interacts with the network. Open the config.xml file, located in `$SF_HOME/apps/james/james-2.2.0/apps/james/SAR-INF/` for version 5.1.

Locate the '`<mailet match="All" class="RemoteDelivery">`' section. add a subnode '`<bind>$addr</bind>`' where $addr is the ip address that James should be sending mail from.

Near that area, there is a <servernames...></servernames> section. Confirm/change the two autodetect options (autodetect, autodetectIP) to false. Next, add the fully qualified host name, and the ip address that will be used, to their own <servername> entry.

After the changes are complete, save the config.xml and restart the application.

Reports

Questions about reports in TeamForge.

Concept definition.

What kind of objects can I create reports on?

Reports can report on data from trackers, tasks, planning folders, and repositories.

Task reports can report on data in a single project or across multiple projects. Tracker reports can report on data in a single tracker or across multiple trackers in a project or multiple projects. Task and Tracker reports are in tabular format and grouped under Table reports. You can also create reports on data in planning folders and on data in source code repositories.

How do I connect to the Datamart?

You can use the *psql-reporting-wrapper* script to connect to the datamart.

Usage

Run the following command:

```
sudo [RUNTIME_DIR]/scripts/psql-reporting-wrapper
```

How do I change the time to run the ETL jobs?

The *ETL_JOB_TRIGGER_TIME* can be modified to specify a different time.

By default, the ETL job runs at 2:30 AM (local time) everyday. It is recommended to run this once a day to avoid any performance degradation of the Teamforge site. See *ETL_JOB_TRIGGER_TIME* on page 788 for more information.

How can I check the status of ETL?

The `[RUNTIME_DIR]/scripts/collabnet status etl` displays the status of the ETL process.

You can get additional information about the various ETL jobs that are configured using the command `[RUNTIME_DIR]/scripts/etl-client.py -a`

What can I learn from a burndown chart?

Burndown is the estimated amount of work that remains to be done in an iteration or a release, compared with the work originally estimated.

At any time, you can click on a planning folder (release, iteration or standard) and view the burndown chart which shows the work to be completed in terms of story points within the given time frame. This helps you see how the team is progressing towards "done". In the burndown chart of a standard planning folder ('Folder') you can see how the sum of the estimated and remaining effort and points (story points) for all contained and descendant features is changing within a given time frame by choosing the options available in the View Mode.

In both 'Release' and 'Folder' burndown charts, you can use the trend line shown in your burndown chart to do some useful things:

• Project the time when all the work for this iteration will be completed.
• Predict the amount of work your team can expect to complete during any given iteration.

Release burndown chart

A release burndown chart shows an iteration-wise breakup of the work to be completed, that is, it shows how much work (in terms of points) is remaining at the start of each iteration.

The information you derive from your burndown chart will help revise the product scope, make accurate planning decisions, and refine implementation details.

In general, a burndown chart trends downward until it reaches zero. In practice, some events can reverse the downward trend of your burndown chart. For example, development work frequently uncovers a greater scope for a user story than was initially estimated. As a result, you'll revise your estimate of remaining work on that story, and your burndown figure may be pushed upwards.

The burndown chart uses whatever effort units you are using in the planning folder.

☞ **Note:** The story told in your burndown chart is only as reliable as the underlying data. The owners of individual task artifacts can help keep the burndown chart accurate by regularly updating their remaining effort and points (story points).

What can I learn from a capacity chart?

The capacity chart shows the project team's judgment of how much work can be done with the resources available and the time period represented in the planning folder.

- When you compare average capacity with individual users' assignments, you can see which team members are relatively over- or under-assigned. You can also see how much of the estimated and remaining effort is unassigned.
- Comparing average capacity with the relative priorities of artifacts is one way to gauge your team's ability to deliver the work that the product owner has defined.
- When you view average capacity against the count of open and closed artifacts, you have another way to assess your team's likelihood of meeting its sprint commitment.

The data behind the capacity chart is updated in real time, so that your team can respond quickly to changes in effort estimates or relative priorities.

Capacity is expressed as a numeric value, using the same scale as you are using for estimated, remaining, actual effort accounts.

You can see the capacity of a planning folder in the planning folder summary view.

☞ **Tip:** A planning folder's capacity must be equal to or greater than the total estimated effort (or total remaining effort) for the artifacts in the planning folder. To avoid confusion, you may want to wait for estimates to emerge before setting the planning folder's capacity.

How do I export reports?

You can export data of a Table report.

Clicking the **Export** button that is placed at the top right corner of the **View Report** page exports tracker or task report data to a .csv file.

What can I learn from an Open by Priority chart?

Dynamic planning is most effective when you know the relative priorities of work items you are tracking. The **Open by Priority** chart helps you check that your team is working on the optimal mix of artifacts.

Use the **Open by Priority** chart for a quick overview of the type of work your project is working with. A glance at this chart can suggest some broad generalizations, which you'll then want to test by examining your planning folder contents more closely.

- When the darker bars on the left side of the chart are high, you are probably working on the high-impact issues that come up earlier in a work cycle. Don't be surprised if the trend line in your **Burndown** chart is not yet sloping steeply downward to the right.
- When the lighter bars on the right side of the chart are high, you are likely to be working on cleanup issues and refinements. You may be approaching the end of the work cycle you defined in this planning folder. At this stage, the Closed slice of the **Open vs. Closed** chart is likely to be equal to or larger than the Open proportion.

☞ **Note:** The **Open by Priority** chart won't tell you how much work is involved in the artifacts that it measures. For that, check the **Burndown** chart.

What can I learn from an Open vs. Closed chart?

Sometimes you just want a raw count of the number of work items you are looking at.

Use the **Open vs. Closed** chart for a quick overview of your team's progress in purely numerical terms. A glance at this chart can suggest some broad generalizations, which you'll then want to test by examining your other charts and your planning folder contents more closely.

The **Open vs. Closed** chart is good for some basic summary information. You may want to share this chart with executive sponsors who need to know how things are going in quantitative terms, but don't need a lot of detail about the types of work being addressed.

How does TeamForge deliver activity reports?

The data in your reports comes from a special database that extracts live site data from the production database at intervals you specify.

You can specify the time at which the reporting data is refreshed from the production database. By default, the extraction takes place daily at 2:30 a.m. in the TeamForge application server's time zone. See *Schedule data extraction for reporting* on page 577.

The reporting database can be deployed on a separate machine to help channel load away from the application server. Historical data is available even if the application server no longer stores it.

Where does the reporting data come from?

An ETL application extracts data from the live production PostgreSQL or Oracle database where the TeamForge site stores most of its critical data. (Information about reporting configurations is also stored in the production database.) Some data is also gathered from the file system.

How is the production data converted into reporting data?

TeamForge extracts a snapshot of the production data, transforms it into a format that supports reporting requirements, and loads it into the datamart, which is optimized for fast retrieval. The Extract-Transform-Load (ETL) application is a Tomcat JVM running as a TeamForge service under the TeamForge integration server architecture.

Where is the reporting data kept?

After the ETL app collects and processes the live site data, it is stored in a separate database called the datamart. If the TeamForge site uses a PostgreSQL database, then the datamart is also a PostgreSQL database; likewise for Oracle. The datamart uses a Star Schema-based design for tables.

How are the reports shown in the TeamForge user interface?

The reports are rendered in the TeamForge UI using Adobe Flex.

☞ **Note:** When a site is upgraded, there will be a delay before reporting data is available to users, until the scheduled ETL run has occurred. Performing a manual ETL run immediately after an upgrade is not advisable, since it could consume a lot of system resources leading to performance problems.

Why do I get duplicate records in a tracker report?

Yes, this is a known limitation.

For a tracker report, when you select more than one value from a multi-select, user-defined field as filter and if an artifact is associated with all of the selected values, then that artifact's record is duplicated for each of the selected values.

For example, assume that you have a 'Select User' multi-select, user-defined field with values 'User 1', 'User 2' and 'User 3' in a tracker report. All these three values are associated with 'artifact 1001'. Select all three values as filter and generate the tracker report. You will see 'artifact 1001' record being duplicated, that is, you will see three individual 'artifact 1001' records created for each of the three users.

Subversion and CVS

Questions about Subversion and CVS.

Concept definition.

What's the difference between a 'managed' and 'unmanaged' CVS server?

You can use TeamForge to control a CVS server, or you can run the CVS server stand-alone, or 'unmanaged'.

☞ **Note:** Only CVS servers can run unmanaged. Subversion servers are managed by TeamForge.

Managed server

A *managed server* gets its instructions from TeamForge to which it is integrated with. Use TeamForge for SCM tasks such as:

- Creating repositories
- Adding and removing users
- Enabling or disabling user access to the repositories

Both new SCM servers and those with existing repositories can be managed by TeamForge.

Unmanaged server

On an *unmanaged CVS server*, you use the CVS's own utilities to handle repository management, user management, and access permissions.

Add an unmanaged server if you know that all repository and user creation and management must be done manually. For example, set up an unmanaged CVS server when:

- A corporate IT policy prohibits applications such as TeamForge from making any changes to the users or repositories on the corporation's CVS servers.
- A CVS server has existing repositories that cannot be changed to adhere to TeamForge standards.

Subversion replication in TeamForge

Here's some information on how replication could be useful in your TeamForge site and what to consider when you plan to set up a replica.

Why replicate?

Typically, you would deploy a replica for one these reasons:

Projects in remote locations with lower bandwidth or higher latency want the performance of a "local" server.	Your company has a number of developers clustered together at a remote location. When you install a replica inside the LAN of these developers, they can greatly improve their Subversion performance and keep a lot of their traffic off the WAN. In this scenario, you would probably want a replica in each such location. Keep in mind that a replica can only be a proxy for one master -- so if your company has more than one Subversion master server, you may need more than one replica at each location.
You want to reduce the load on the master server.	For example, continuous integration tools can place a lot of load on the server and moving that load to a separate server can increase the response time for other users. In this scenario you probably only need to add one replica; you'd add it as close to the master as possible so that synchronization is quick. Of course the previous point can be a factor here. If the continuous integration server is at remote location, then you would want to put a replica near the continuous integration server.

Rules for using a replica

To create a replica in TeamForge, you start with Subversion Edge. A Subversion Edge wizard lets you convert the server into a replica of a Subversion server in TeamForge.

When you configure the replica server, you provide the TeamForge username and password to use for the replica. These are the credentials the replica uses when it replicates Subversion content. This user must be added to projects and given permissions to the repositories being replicated. Those permissions also control what parts of the repository will be replicated. So if you have folders that should not ever live on remote servers, you can set up path-based permissions and that content will never be replicated to the server.

If you forget to set up permissions, the replication will fail. However, there's no real harm done, and once you fix the permissions, you can do it again.

The replica user can be a normal user account -- it does not have to be an Admin account. If the replica is set up and maintained by an Operations team, they might want to just use an Admin account so that project teams do not have to worry about adding the user to the project or setting up permissions.

Permissions for end users accessing those repositories will follow the normal TeamForge rules.

How do I move an existing CVS repository into TeamForge?

You can import and manage an existing CVS repository with TeamForge.

Do the following to move an existing CVS repository:

1. Stop CVS access to the old repo.

   ```
   chmod -x /usr/bin/cvs
   ```

2. Tar the old repo.

   ```
   cd /cvsroot/old_repo
   tar zcvf /tmp/old_repo.tar.gz
   cd..
   mv old_repo /tmp
   ```

3. Restore CVS access.

```
chmod +x /usr/bin/cvs
```

4. Transfer the repo to the TeamForge CVS server.
5. Create the new repo from within TeamForge.

```
Browse to your project
Click the Source Code button
Create your new repo
```

6. Untar the old repo.

```
cd /cvsroot/new_repo
tar zxvf /tmp/old_repo.tar.gz
```

7. To synchronize permissions, perform the following steps.

   • Login as a TeamForge site admin
   • Click the Admin link
   • Click the Integrations button
   • Check the CVS integration you want
   • Click the Synchronize Permissions button

8. Verify the new repo.
9. Remove the old repo.

```
/bin/rm -r /tmp/old_repo
```

How do I move an existing SVN repository into TeamForge?

If you have an existing SVN repo, you can manage it with TeamForge.

To move an existing the SVN repo, perform the following steps:

1. Stop SVN access to the old repo.
2. Dump the old repo.

```
svnadmin dump /svnroot/old_repo > /tmp/old_repo.dmp o mv /svnroot/
old_repo /tmp
```

3. Restore SVN access.
4. Transfer the repo to the TeamForge SVN server.
5. Create the new repo from within TeamForge.
6. Browse to your project and click the **Source Code** button, then create your new repo.
7. Load the old repo.

```
cat /tmp/old_repo.dmp|svnadmin load /svnroot new_repo
```

8. To synchronize permissions, perform the following steps:

   • Login as an TeamForge site admin.
   • Click **Admin**.
   • Click **Integrations**.
   • Select the SVN integration you want.
   • Click the **Synchronize Permissions** button.

9. Verify the new repo
10. Remove the old repo

```
/bin/rm -r /tmp/old_rep
```

How do I enable Neon debugging in my Subversion client?

This is easliy done in a Linux environment by editing the existing servers text file. On Windows it is not easily done and not recommended.

On a Linux system, edit the `~/.subversion/servers` file by adding the line "`neon-debug-mask = 130`" (without quotes) to the [global] section of the file, making sure that you also un-comment the [global] line as well. Once Neon debugging is enabled, you should see much more output from each svn command.

Although Neon debugging is possible on Windows, it involves steps that are too complex for most end users to undertake, including compiling Windows binaries for the specific platform and manually handling any errors that arise during that process.

How do I create a local backup of a remote SVN repository?

You can back up your remote Subversion repository to your local system by keeping a mirror copy of the repository locally using `svnsync`.

Trackers

Questions about Trackers in TeamForge.

Concept definition.

Can I assign an artifact or task to a group of users?

Unfortunately this is not possible with the current implementation of groups in TeamForge.

As a workaround, you can create a new user account and set its email address to that of a mailing list that all group members are subscribed to. Then the submitting user can assign the artifact to this user account, and the mailing list will be notified.

How do I create an artifact through email?

To create a tracker artifact using email, send an email message to <tracker id>@<TeamForge server>.

You can find the tracker id on the **Artifact List View** page. Fields are mapped as follows:

- To: Tracker email address
- Subject: Artifact title
- Body: Artifact description
- Attachments: Attachments

I don't need the priority and assigned-to fields, how do I get rid of them?

These fields are used internally by the system and cannot be deleted or disabled.

Why do the open tracker counts differ from when I filter on 'Open'?

The 'Open' category on the summary screen is based on the meta status 'Open'. It includes multiple statuses; Open, Fixed, and any other statuses that are defined as equivalent to Open by the tracker admin for your project.

For example, suppose the Tracker Summary block shows 21 Open artifacts. However, when filtering on the 'Open' status, only five artifacts are displayed.

If you filter by 'All Open' you will see all 21 issues with statuses defined as Open. If you filter specifically by 'Open' you will see only the artifacts that specifically have the status 'Open'.

How do I edit the predefined data in the various fields in the issue tracker?

Use **Configuration Options** to edit the required data in the fields.

For more information about customizing your installation see: *Customizing your TeamForge installation*

Can I limit the number of characters for fields in a tracker?

No, it is not possible in the current version of TeamForge.

Currently, while creating the tracker custom flex field of input type 'Text Entry', you cannot control the maximum number of characters to be allowed. There is a 'Field Width' that takes values to control the size of the text field, but not to limit number of characters entered.

Is it possible to move an artifact from one tracker to another?

Yes. It is possible.

Simply use the cut button to remove the artifact, then paste it to another tracker. You can also change the tracker type while editing the artifact using the **Edit Artifact** page. The destination tracker need not be in the same project, but if the tracker definitions differ data could be lost.

How can I change the display order of the fields in my tracker?

Simply drag and drop the fields within your browser.

Horizontal rules may also be added in this manner.

How do I create a parent or a child artifact?

A tracker artifact can be a parent or a child to another artifact.

- A child artifact can have only one parent.
- A parent artifact can have any number of children.
- A parent artifact cannot be closed if a child is open.

To create a new parent or child artifact of an existing artifact:

1. Click on the **Dependencies** tab of the artifact.
2. Click on **Create New Parent** or **Create New Child**. Fill in the form as you would for submitting an unrelated artifact.
3. Click **Next**.
4. Enter a comment that describes the relationship, if appropriate.
5. Click **Finish**.

☞ **Note:** If **Choose Parent** or **Create New Parent** is not visible, the artifact already has a parent artifact.

What does the status of a planning folder mean?

The status of a planning folder communicates where it currently stands in the development process. You can set the status of a planning folder to values you define yourself.

The status of a planning folder can help project members make sense of what can be a complex development process. For example, consider these scenarios:

Drop everything!

Just when you have prepared a planning folder and are ready to kick off a sprint, the product owner gets word of a more pressing set of features and asks you to postpone this work and get right to the newly revealed priority. You are worried that your project's momentum might lead project members to keep picking up tasks in this planning folder instead of switching to the new work.

Create a status called 'On hold', and specify it as an 'Inactive' status. When you give the planning folder your new 'On hold' status, project members whose view shows only active folders would not see this one in their planning folder list.

It's a wrap

Your sprint has just concluded. You don't want any further development work to happen in this planning folder, but you do want team members and others to be able to peruse it for insight into what went right and what went wrong.

Create a status called 'Retrospective', and specify it as an 'Active' status. When you assign this status to your planning folder, participants will be able to confirm that the sprint in question is finished and their comments are invited.

Status tips

- It is a good idea to choose a default status, such as 'Under construction', so that project members are not confused when new planning folders appear.
- You can organize your planning folder statuses in a way that makes sense for your team, such as alphabetically or chronologically.
- Delete statuses that are no longer being used, so you do not clutter up the selection.
- When you delete a status that is in use by one or more planning folders, the status of those planning folders is changed to 'None'. Planning folders that were created in an earlier TeamForge version also get a 'None' status at first.

How is a project template structured?

When you create a template that includes project content, each tool brings in its own kind of structure, depending on the type of content it manages.

Folders

Some tools, such as Documents, Tasks, and File Releases, can be thought of as folders that contain individual items, subfolders, or both.

For example, the Documents tool has a Root Folder, inside which you can create a tree of subfolders to organize your documents according to your project's needs. Every document lives inside one of these folders.

☞ **Note:** This works the same way regardless of the name of the root folder. For example, in the Tasks tool the root folder is called the Tasks Summary, and in the Discussions tool the root folder is called the Forum Summary.

In a project that has been in use for any significant time, users have probably created some number of documents, which they have shared by adding them to a folder in the Documents tool. When you create a project template from that project, you can choose to include those documents in the project template or not.

- If you choose to include the documents in the template, those documents will appear in any new project that is created using that template.
- If you do not include the documents in the template, new projects based on that template will include the Documents tool, including the Root Folder and its sub-folders, but none of the documents that existed in the original project.

Not folders

The Reports and Wiki tools are not organized in folders. When you include the content from these tools in a template, new projects created from that template include a flat collection of all the Wiki pages or reports in the original project, with all their text and data.

How do I create new values in the 'Reported in Release' field?

SFTF will populate that field automatically from the packages that are defined in the file release tool.

What fields can I use in a tracker?

You can use fields provided by TeamForge, and you can create your own custom fields. Most trackers use a mix of TeamForge-provided fields and user-defined fields.

TeamForge-provided fields

Some fields are provided by TeamForge for all trackers.

System-defined fields: Some TeamForge-provided fields are always present in every tracker. These are identified as "system-defined" fields in the field list.

- The **Artifact ID**, **Title**, and **Description** fields are system-defined fields. They are always present and always come first when you view an artifact.
- Some system-defined fields can be reordered. For example, the **Assigned To** field is a system-defined field, so it is always present, but you can move it around in your tracker's display.

Configurable Fields: Some fields are available for any tracker, but can be modified in different ways. These are called "configurable" fields.

- Any configurable field can be a required field. When a field is required, a user cannot submit an artifact without providing a value for that field.
- Some configurable fields can be disabled. When a field is disabled, TeamForge does not store data for it or include it in calculations. For example, the **Reported in Release** field is meant for tracking bugs, so you may want to disable it for a tracker that's for user stories. But the **Status** field is important for any tracker, so it can't be disabled.
- For some configurable fields, you can specify whether users see the field when they submit an artifact.
- Some configurable fields need values for users to choose from. You have to provide those choices.
- You can set the width and height of configurable text fields.

User-defined fields

When none of the TeamForge-provided fields captures exactly the information you need to track, you should create a field tailored to your own needs. Such fields are identified in the field list as "user-defined" fields. See *Create your own tracker fields* on page 38 for details.

What happens when a changed value makes a dependent field invalid?

When a user changes a field value on which the values of other fields depend, some dependent values may need to be corrected.

This can happen if:

- A parent value has been changed in such a way that its child values are inconsistent.
- A text value has been changed in such a way that it no longer matches the regular expression that describes the acceptable values.

☞ **Note:** Linking fields in dependent relationships does not modify existing data. However, when users later modify fields that are linked, they do have to adhere to the relationships among the fields.

Changing values

When a user changes the value of a field, and the value in another field becomes invalid as a result, the latter field (we call it the 'dependent' field) is reset to its default value the next time someone updates that artifact.

For example, picture a tracker called 'Interior decoration' in which, if the **Color scheme** field is set to Warm, the **Carpet color** field can be set only to Red, Orange or Yellow. Red is the default value.

User 1 sets **Color scheme** to Warm and **Carpet color** to Yellow.

User 2, after consulting the client, comes in and changes the color scheme to Cool. Now the allowed values of **Carpet color** are Blue, Gray and White, with Blue as the default. But the actual value is still Yellow, which is now invalid. It's up to the user who is updating the artifact to choose a new color from the new set of valid colors. If the user doesn't make a choice, TeamForge automatically changes the value of **Carpet color** to Blue, because that's the default value when **Color scheme** is set to Cool.

Changing dependency rules

When you restructure a hierarchy of dependent values in a way that leaves invalid child fields in multiple artifacts, each of those artifacts must be fixed the first time a user edits it.

For example, in the artifact `room1`, User 1 has set the **Color scheme** field to Warm and the **Carpet color** field to Red. Suddenly User 2, the project manager, realizes that red is a terrible color for carpeting. User 2 changes the tracker settings so that **Carpet color** can be set only to Orange, Yellow, or Maroon.

The next time User 1 comes back to the `room1` artifact, TeamForge warns that the field value is in an inconsistent state. If the user tries to edit the specific field that is inconsistent, changes to the artifact cannot be saved until the user selects one of the new allowed values for that field.

Changing text validation rules

If you change the regular expression that governs what content is allowed in a text field, each affected field must be fixed the first time a user edits it.

For example, imagine that your 'Interior decoration' tracker includes a text field in which users can record the telephone number of a carpet installer. You set up a validation rule like `[(]\d\d\d[)]\s\d\d\d[-]\d\d\d\d` to ensure that only standard U.S.-format phone numbers can be entered.

The tracker has been in use for a week when you realize that email would be a better way to communicate with carpet installers. You change the validation rule to something like `\b[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,4}\b`, to ensure that users will enter an email address.

The next time a user tries to update an artifact with a telephone number value in that field, TeamForge warns that the field value is in an inconsistent state. If the user tries to edit the specific field that is inconsistent, changes to the artifact cannot be saved until the user enters a value for that field that matches the new validation rule.

Deleting a parent field

If you delete a field that contains values that another field's values depend on, the dependent field becomes a standard single-select field on its own. No correction is needed.

Changing 'requiredness'

*   When a field has a parent field that is required, the child field's default value is set to None. If that required parent field is deselected, the child field no longer has to be required, but Required remains the default.
*   If you require a specific field value before an artifact can be placed in a given status, that field's children are subject to the same requirements. See *Create a tracker work flow* on page 40 for more about controlling what status an artifact can be in.

Discussions

Questions about Discussions in TeamForge.

Concept definition.

Is the discussion forum creator subscribed by default?

Yes. The user creating the discussion forum is subscribed by default.

For more information about discussion forums, see: *Communicate with project members*

How can I monitor a forum or mailing list?

There are two methods for monitoring a forum or mailing list in TeamForge: at the Discussion Forum level and at the My Workspace level.

At the Discussion Forum level:

- Check the box beside the forum in question.
- Select the **Monitor** button.
- Select **Monitor Selected**.

At the My Workspace level:

- Select **Monitoring**.
- Select the project name.
- Select the **Monitored Applications** tab.
- Select **Discussions**.
- Click **Save**.

☞ **Note:** You must monitor a forum to receive email from the mailing list associated with that forum.

How do I create a new forum topic?

A forum can have any number of topics. A topic starts a message thread to which other users can reply. Create a new forum topic to begin discussion of a new subject.

To create a forum topic:

1. Click **Discussions** from the **Project Home** menu.
2. On the **Forum Summary** page, click the title of the forum in which you want to create a topic. The **Topic Summary** page is displayed.
3. Click **Create**. The **Create a Topic** page is displayed.
4. Enter a description of the topic in the **Subject** field.
5. Enter the message text in the **Message** field. After the topic is created, other users can reply to this message.
6. (Optional) If you want to add an attachment to the topic, click **Browse** to select the file.
7. Click **Save**. The forum topic is created.

How do I find the email address for a forum?

The email address for the forum is displayed on the **Topic Summary** page. If a mailing list is not enabled, you will not see an email address on the **Topic Summary** page.

To find the email address for a forum:

1. Click **Discussions** from the Project Home menu.
2. On the **Forum Summary** page, click the title of the forum in which you want to create a forum topic. The **Topic Summary** page is displayed. The email address is listed in the Mailing List field.

How do I remove a user from a discussion?

Unfortunately there is not currently a way for a TeamForge admin to remove a user from monitoring.

You will need to have the user login and remove themselves from monitoring via the **Monitoring** button on the **My Workspace** page.

How do I post a reply to a forum topic?

To post a reply to a forum message, follow the steps below.

1. Click **Discussions** from the **Project Home** menu.
2. On the **Forum Summary** page, navigate to the forum topic in which you want to post a message.
3. In the section containing the message you want to respond to, do one of the following:

    - Click **Quote** to quote the original message in your response message. The **Post a Reply** page is displayed with the original message quoted.
    - Click **Reply** to omit the original message from your response message. The **Post a Reply** page is displayed without the original message.

4. Enter the message text.

5. (Optional) If you want to add an attachment to the message, click **Browse** to select the file.
6. Click **Save**.

The forum message is now posted. Other project members can reply to it using the same process.

I can see the message I posted to a discussion in the web UI, but I didn't receive any of the responses through email. Why is this?

To receive form posts through email, you must monitor the forum.

You can do this by placing a check next to the forum you wish to monitor and choosing 'monitor selected' from the **Monitor** drop-down list at the bottom of the Forum list. For more information on this, see *Monitor many items*.

Why are some of the discussions threaded?

Posts that are sent the mailing list address of the discussion will create new topics, and hence will not be threaded.

Posts that are made in response to another post will be threaded beneath the original post.

What happens when I post to a moderated discussion forum?

A message to a moderated discussion forum is held until a moderator acts on it. (Except if you are a trusted user. These messages do not require moderation.)

- The moderators of the discussion receive an email notification that you have posted a message. The email notification contains the URL path to moderate the post.
- A moderator can either approve or reject your message.
- If the moderator accepts the message, the message status changes to "Accepted" and the message is posted to all the users monitoring the forum.
- If the moderator accepts the message and trusts the sender, the message status changes to "Approved and Trusted" and the message is posted to all the users monitoring the forum. All subsequent posts from you are automatically approved.
- If the moderator rejects the message, he/she can include his/her comments or reasons in the moderation rejection email, and the message is removed from the archive.
- In the **All Topics** tab, an hourglass icon indicates which topics contain posts that await approval.

Who can be a moderator?

A discussion forum moderator is selected by the forum administrator.

- You can be a moderator by being a project member with forum post permissions.
- A moderator can moderate a discussion forum using emails or the web UI or one of the CollabNet desktops.

  If the mailing list isn't enabled for a moderated discussion forum, moderation can be done only through the web UI.
- A moderator can trust the sender and approve a message or reject the message.

> ☞ **Note:** To enable or disable moderation or add or remove moderators, you must be either the forum administrator or project administrator.

Can I move a discussion to another project?

No. Discussions cannot be moved between projects.

For more information see: *Discussions and mailing lists*

Do project owners get automatically subscribed to the discussion forum started by another member?

A project owner does not get subscribed to the discussion forum started by another user.

To make sure the project owner is included, add the user as a member when creating the forum.

Can I subscribe to a TeamForge discussion forum's mailing list through an email?

Sure, you can subscribe to a discussion forum's mailing list to keep up with changes, through an email. When you subscribe or monitor a discussion forum, you are notified by an automatic email whenever there is any update to the discussion.

To subscribe to a mailing list: You can subscribe to the discussion forum's mailing list by sending an email to `<mailing list name>-project name-subscribe@<domain>`.

You must have the 'Discussion: view' permission to be made a subscriber. Check with your forum administrator, if you do not have the permission.

As with other items that you may be monitoring, the discussion forum to which you subscribe through an email is also added to the **My Workspace** > **Monitoring** > **Monitored Objects** list. Your user name is added to the 'Users Monitoring' list of the discussion forum. Both these entries are removed when you unsubscribe from the mailing list.

To subscribe in digest type, send an email to `<mailing list name>-project name-digest-subscribe@<domain>`

To unsubscribe from a mailing list: You can unsubscribe to the discussion forum's mailing list by sending an email to `<mailing list name>-project name-unsubscribe@<domain>`.

Why would I want to make a discussion forum moderated?

Automated processes can only do so much to protect forum quality. Moderated posting can be thought of as a temporary transfer of control from automated processes to a human decision maker.

Software can instantly analyze an email to determine whether the sender is allowed to post messages directly to the list or if the message should be sent for moderation. However, the software doesn't recognize the sender as a person who participates in the organization, nor does it have the sophistication needed to determine whether the message contents are on-topic for the mailing list.

Advantages of moderated posting:

- Moderators help keep content on-topic by rejecting off-topic messages and providing helpful suggestions to users whose posts are rejected.
- Moderators help maintain a positive environment for list users by rejecting messages containing harsh or abusive language.
- Moderators can allow deserving nonmembers to post to lists that are ordinarily closed to nonmembers.
- Moderators prevent spam.

Example

Ashish makes the development forum in his project a moderated one as he wants to make sure that all the messages posted to the discussion come to him for approval before they're included in the forum. When a message arrives, he reads the message, and if it's appropriate for the discussion, he accepts it; if not, he rejects it.

Over time, Ashish finds that the traffic in his discussion has increased and he is no longer able to moderate all the posts by himself. So he adds a couple of other senior developers in his project as moderators, who can share the responsibility of moderating the forum.

After a while, Ashish realises that he doesn't have to reject any messages posted to the forum as everyone seems to understand the purpose of the forum and uses appropriate language in emails. So he removes the restriction and make the forum an unmoderated one. Now Ashish and the other moderators no longer receive emails for approval when a user posts a message to the discussion. Messages are directly included in the forum and delivered to the forum subscribers.

Documents

Questions about Documents in TeamForge.

Concept definition.

How does TeamForge support documents?

A key element in a successful product development project is ensuring that stakeholders from all functional groups are involved in an iterative document review and approval process, particularly for critical documents, such as Product Requirements Documents.

It is also critical that any changes to key documents are recorded and the new information quickly communicated to everyone concerned.

The TeamForge Document Manager helps you manage your documents throughout their life cycle and ensure the appropriate level of involvement of other project members. In addition to storing documents for reference purposes, the Document Manager provides a document review work flow process to allow you to actively engage other users in the review and approval of a document.

Simple notification process

If you would like to keep other TeamForge users informed when the status of a document, such as when a document is posted or updated, or there is a change in the contents or status of an existing document, but do not require a formal review and approval process, the following steps provide a best practice for this level of document management:

- Post the document to the Document Manager in the desired location. If you would like to prevent other users from editing, moving, or deleting the document, use the 'Lock Document" feature. (If you choose not to lock the document, any user with the document edit permission can edit and update the document.)
- Notify other users that the document is available and request that they begin monitoring it. All users monitoring an item receive email notification whenever the item is updated.

Document work flow review and approval

In cases where more formal document review is desired, the TeamForge Document Manager provides an easy-to-use work flow process for managing a document's review and approval.

After submitting a document, you can start a document review. You are prompted to provide the following information:

- The names of all required and optional reviewers. (All reviewers must have permission to access the document.)
- The date by which the review must be completed.
- Email message text.

You will be notified each time a reviewer submits review comments.

When selected to review a document, users receive an email notification with the relevant details and a link to the document.

You can also choose to attach the document to the email notification. An item also appears in the **Documents Awaiting Review** section of each reviewer's **My Page**, including the due date.

A **Submit a Response** section is provided within the Document Manager where reviewers can enter their review comments. Reviewers should be instructed to include "I approve this document" or similar text in the **Submit a Response** section to indicate that they have approved the document.

All reviewers and the document submitter will be able to read the reviews of all other reviewers once they have been submitted.

Why can't I edit a document when it opens in my browser?

There are chances that your browser swallowed the running application.

If you click on a document in TeamForge (for example, a Word doc), and it opens in your browser instead of launching a separate MS Word window, chances are you will not be able to edit this document. This is because your browser has either called one of the MS document viewers that do not have native edit capabilities, or the browser has 'swallowed' the running MS Word process and the document has written to your %TEMP% directory as a read-only file.

You can either choose to download the file to your desktop and then click on it to edit, or configure your browser to not swallow the application (this is described in various MS KB articles).

Why are some uploaded documents missing icons when displayed in TeamForge?

TeamForge has a small internal mapping of which icon is associated with which mimetype(s). This mapping is necessarily small, as a complete mapping would be exceedingly large, and almost always out of date.

You can examine the mimetype that TeamForge is receiving by examining the document's attributes using our SOAP API. You can also override this mimetype info via the API by uploading a new version of the document (which can be the exact same document contents). There is sometimes a generic icon for documents from certain applications. TeamForge accepts and stores the mimetype that the user's browser sends on the HTTP upload of the original document. As an example, let's examine the MS Word document. Some browsers will send strings like `application/vnd.ms-word`; others will send `application/msword` and others will send something completely different.

Can I link to documents outside of TeamForge?

TeamForge has the concept of a 'url doc' to support this very usage.

When creating a document in TeamForge, simply choose this type from the **Create** screen and then enter the URL that references the document in the existing external system. This will create a 'placeholder' document in TeamForge that can be associated to, reviewed, and so on as if it were a normal document, while maintaining the document's actual contents in the external application.

Can I lock a document in TeamForge?

Absolutely, yes. You can specify a document as locked at any point in time (document create, document edit, etc).

You can do this through the UI or through the SOAP API. Simply check relevant boxes available in the **LOCK DOCUMENT** section on the **Document details** page.

By default, edit and download locks are disabled for a document. To lock and unlock your document, you must have the Document Edit permission in addition to the Document Create permission. To prevent others from editing or downloading a document, you can choose the edit lock or the download lock as required. Selecting the edit lock automatically enables the Document Administrator to edit the document and allows you to select the download lock option.

To edit or download a locked document, the user must have the Document Edit permission. The user who has locked the document can edit or download it at any point in time. A Document Administrator that has the option, **Allow Document Admin to edit** enabled in the **LOCK DOCUMENT** section or a Site Admin can also edit or download a locked document.

Does TeamForge automatically resolve conflicts in documents made by multiple concurrent editors?

If you store your documents in CVS, Subversion, or some other SCM tool, then that tool will handle conflict resolution per its normal means.

However, if you are using the Document Manager component of TeamForge, then there is no automated means within the product to prevent another user from uploading a newer version of a given document that does not contain the changes you just uploaded. To prevent this, we recommend you use the edit lock and download lock features of TeamForge to prevent others from editing and downloading a document you are currently editing.

What document types are supported in TeamForge?

TeamForge supports all known and unknown document types.

In fact, TeamForge makes absolutely no distinction on a file type when uploading a document. This means that you can even upload binary files (like a Zip archive) into the document manager. Note that while the TeamForge accepts any incoming data stream as a document, it does use the mimetype sent by the browser to determine if it has the proper icon to display for the document.

Why doesn't an open review automatically close when a new version of the document is uploaded?

TeamForge does not automatically close a document review under any circumstances (new document version, review due date passes, and so on).

This was a conscious decision on our part in which it was decided that TeamForge cannot always be aware of the business rules or personnel availability at a customer. For example, TeamForge cannot know that the one person whose document review input is most needed is on vacation for the four days the document was under review. If TeamForge were to close the review, then it would disappear from the user's **My Page**. Additionally, TeamForge cannot know that a new version of a document supersedes the prior versions (or that it does not supersede it). You may be uploading a 'draft' or 'work-in-progress' of the new version as document review feedback is received and if TeamForge were to close the review at this point, you might have received feedback from less than one percent of the reviewers.

Who can work with documents?

Any project member with the Document Edit permission can edit a document.

Can I set permissions so that users can move documents but not delete them?

You cannot configure document management permissions so that a user can move documents but not delete them.

It is not possible to separate move and delete permissions, because a move is actually a copy/delete action. The document is not really moved, it is copied to the new location and then deleted from the original location.

Why am I not able to access a folder in TeamForge documents?

You can access a folder in TeamForge documents only if you have permission for the folder or any of its parent folders.

For TeamForge documents, users having permissions to the parent folder can access all the subfolders. Users having permissions to a subfolder can only view the hierarchy of the parent folders; they cannot access the documents in those folders.

Are role-based permissions allowed for subfolders in the TeamForge Documents?

Yes. It is allowed in TeamForge Documents.

The Project Administrator can give permission to access sub folders in the TeamForge Documents based on the user roles, using the Roles option.

Can I set permissions for documents alone?

Yes, as a project administrator, you can set individual permissions only for documents.

The Document Admin permission allows a project member to create, edit and administer both documents and document folders. Though this does not include the delete permission, individual permissions to create, edit, and delete can be set separately for document folders and documents on the **Documents Permissions** page (**Project Admin > Permissions > Edit Role > Documents**). This is to restrict a project member from handling document folders or subfolders. For example, if you do not want a project member to create a document folder, but only create a document within a specific folder, you can set the 'Create/View' permission for documents alone. The available permissions are:

- Create/View
- Edit/View
- Delete/View
- View Only

Wiki

Questions about Wiki in TeamForge.

Concept definition.

How do I generate a wiki table of contents?

You can create a table of contents from any heading text that you have in your wiki page.

For versions 5.2 and earlier, generating a wiki table of contents requires the Wiki TOC plugin, available through CollabNet Professional Services.

To enable TOC for a Wiki page, place the following in your Wiki page at the spot where you want the Table of Contents to appear.

```
%%insert-toc
%%
```

The Table of Contents is generated automatically based on the heading markers in the wiki page, e.g. !!!Heading

How do I delete an attachement from a wiki page?

You can delete an attachment from the wiki page.

Use the following steps to delete an attachment:

- Browse to the wiki page that contains the attachment and click **Edit**.
- Update the wiki page, even though you have not made any changes to it.
- Click **Show Details**.
- Click the **Attachments** tab.
- Select the attachment you want removed and click **Remove**.

How can I detect orphan wiki pages?

No. You cannot detect orphan wiki pages.

Unfortunately at this time, TeamForge does not have the ability to locate or display orphaned wiki pages within the UI. This functionality is slated to be included in a future release. If you have an immediate need, please contact Technical Support and a suitable SQL query will be devised.

How do I edit the wiki home page?

Navigate to the project home page and click the **Edit** button to edit the wiki home page.

Once you have enabled the wiki as the project home page option in Project Admin, you must return to the project home page by clicking the **Project Home** button. There will be an **Edit** button on the bottom right corner of the page under graphs. You may need to scroll your browser window to the right to see this button.

How do I make the version comment required for wiki updates?

You can use Velocity to customize the pages of TeamForge and make the version comment required.

Create /opt/collabnet/teamforge/sourceforge_home/templates/body_footer.vm with the following contents:

```
<script>
var updateButton = document.getElementById('edit_wiki_page.update');
if ( updateButton ) {
updateButton.href = 'javascript:submitWikiPageUpdate();';
}
function submitWikiPageUpdate () {
if ( document.editPage.versionComment.value ) {
submitForm(document.editPage, 'submit');
```

```
return;
}
alert("Please include a detailed Version Comment for this change.");
}
</script>
```

Ensure that the file is owned by the sf-admin user:

```
chown -R sf-admin.sf-admin /opt/collabnet/teamforge/sourceforge_home/
templates
```

Ensure proper permissions:

```
chmod 0644 /opt/collabnet/teamforge/sourceforge_home/templates/
body_footer.vm
```

TeamForge picks up the change immediately.

I set my project to 'use wiki homepage' so why isn't my wiki showing up?

TeamForge currently uses two distinctly different wikis.

- If there is the wiki you have already edited, which is available by clicking on the **Wiki** button at the top of any project page, and there is the 'project home' wiki, which is what you enabled in the **Project Admin** screen.
- If you visit the project home page after setting this option, and if you have the proper RBAC permissions, there should be an **Edit** button in the bottom-right corner of the home page under the graphs. Use this button to edit the project home page wiki.

Why would I attach things to a wiki?

There are specific scenarios that would require attachments to a wiki.

The most common example of when you would attach something to a wiki would be when you need an image in the wiki page and you are concerned that if the image is hosted remotely (a corporate web server, for example), it might be moved or removed. Additionally, you might wish to attach a file to a wiki page if the attachment is only truly important in the context of the wiki page and therefore is not important enough to be uploaded to the Document Manager of TeamForge.

CollabNet TeamForge 8.2 release notes

Released on: 08 February, 2016

New features in CollabNet TeamForge 8.2

TeamForge 8.2 has a lot of new features. Here's a list of a few release-defining new features in TeamForge 8.2.

Important announcements

One-hop upgrade compatibility for TeamForge 8.2

👉 **Note:**

The TeamForge 8.2 installer supports one-hop upgrade from TeamForge 7.1 or later versions only.

During an upgrade, the TeamForge 8.2 migration script detects the TeamForge version you run, checks if it's TeamForge 7.1 or later, and if 'true', proceeds with the upgrade.

The migration script aborts with an error message if it detects TeamForge 7.0 or earlier versions, in which case, you must upgrade your site to TeamForge 7.1 or later first and then upgrade to TeamForge 8.2.

👉 **Note:**

TeamForge 8.2 upgrade instructions are for upgrading from TeamForge 8.1 (including patch releases, if any) to TeamForge 8.2. Contact *CollabNet Support* if you want to upgrade from TeamForge 8.0 or earlier versions to TeamForge 8.2.

No support for Perforce integration

Perforce integration is not supported from TeamForge 8.2 and later. All Perforce related `site-options.conf` tokens are obsoleted and documentation removed.

No support for context-sensitive Help for sites without internet access

The HELP_AVAILABILITY and REMOTE_HELP_URL `site-options.conf` tokens have been removed from TeamForge 8.2 (and later). Remove these tokens from your `site-options.conf` file if you are upgrading to TeamForge 8.2.

No support for SUSE Linux Enterprise Server (SLES)

SLES platform is no longer supported. See *CollabNet product end-of-life policy* for more information.

Redesigned TeamForge header

The TeamForge header has been redesigned with an eye for a simplified user experience. With the dynamically changing (context-sensitive) menu header, what you now have in place of a toolbar is a consolidated set of menu
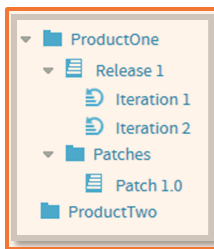
options. Based on your user permissions, you can easily select/access the three main categories: your workspace, system administrator menu and your project.

Main Menu selection: My Workspace

Select Projects to access the Create New Project button

Use this grid to select the function or option you wish to access

The grid displays on a black background because these are site-wide options

Main Menu selection: Admin

Use this grid to select the function or option you wish to access

Planning folder type

The planning folder creation has now been enhanced to make your project planning more organized and more agile-specific. You can now create planning folders of three different types: **Release**, **Iteration** or **Folder**. This enables you to compartmentalize artifacts within an even more structured planning folder hierarchy,making project planning and tracking much easier. Evidently, you can generate three different burndown charts depending upon the planning folder type. For more information about the planning folder types and the modified burndown charts, see *Create a planning folder* on page 17, *What can I learn from a burndown chart?* on page 1009 and *Velocity and average velocity* on page 965.

- Folder
  - Can contain sub-Folders
  - Often used for Product or Deliverable identification
- Release
  - Can contain sub-Releases
- Iteration
  - Mandatory Start and End dates

Select the Planning Folder type using the type button

Folder types are identified by icon in the Planning Folder list

Planning Folder Creation Dialog

👉 **Note:** Though having multiple planning folder types help you logically organize your planning folders, there's no hierarchical restrictions on these different planning folder types. For example, as a user, you have the liberty to create a "Release" planning folder as a child of an "Iteration" planning folder, which may not be the ideal use case in an Agile project.

Project toolbar and integration administration

- Project toolbar administration has now been renamed **Tools** in the **Project Admin Menu**.
- All tools are now presented in a consolidated list: core tools like Tracker and Source Code along with integrations such as IAF applications, linked applications, and Orchestrate sources.
- Use the Tools menu to administer project integrations. The new "Tools" concept consolidates integration administration into a single interface for better context (e.g., ReviewBoard IAF and Orchestrate settings are consolidated on a single page).
- Use the "Add Tool" function to add new integrations to your TeamForge project.

Boards

With the requisite permission, you can now do the following on the Kanban Board:
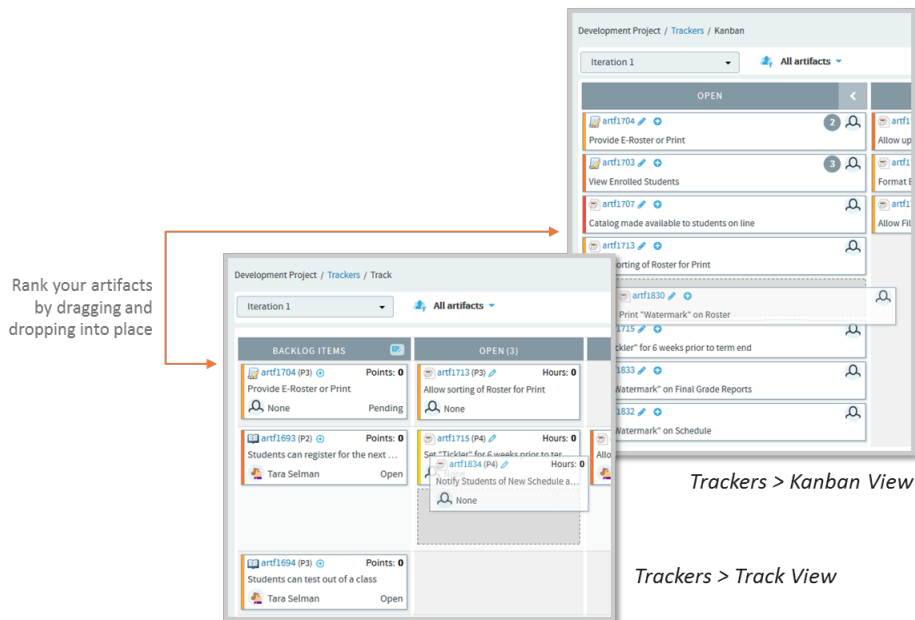
- You can quickly create an artifact, including child artifacts.
- While viewing closed items, use the new calendar tool to toggle between the following options:
    - view artifacts that were closed in the past 60 days

    - or view all of them.

All the related icons (Configure/Refresh/Show all closed artifacts/Manage Boards or Select Board) are now grouped to form a small toolbar at the top right corner of a kanban board for easy navigation.

Rank cards on Task and Kanban boards:similar to planning board, you can now drag and drop a card above or below the other cards within a swimlane to position it in the order you want the artifact addressed on the task and kanban board as well.
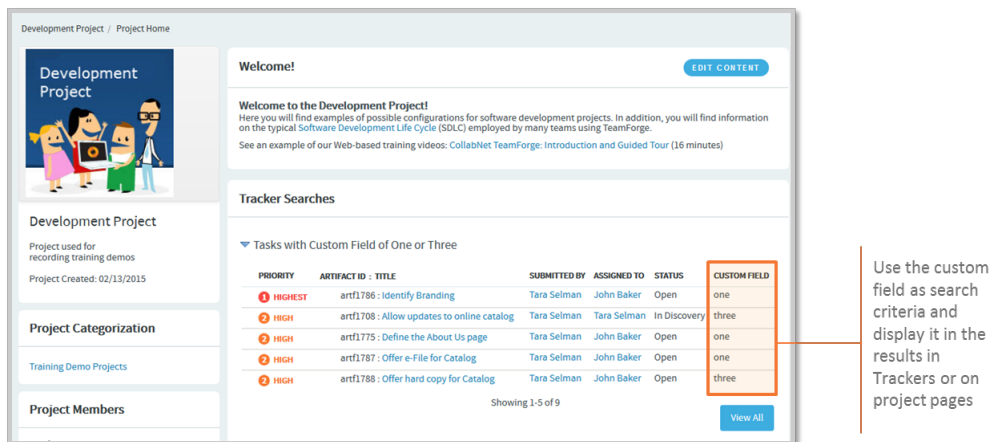
Rank your artifacts
by dragging and
dropping into place

Trackers > Kanban View

Trackers > Track View

Retain card ranking across views:the positioning of cards is retained when you switch between the list and board views (Planning, Task and Kanban).

Tracker

'Priority', a configurable single select tracker field: the single select tracker field, **'Priority'**, is now a configurable field. When you create a new tracker, the default priority is set as '4 - Low'. However, you can set a different default value for this configurable single select tracker field, **'Priority'**, at a later point in time. See *Configure tracker "Selec"t field values* on page 39 for more information.

☞　　**Note:**　You cannot delete or disable the **'Priority'** field.

Tracker search results display on Project Home page: the saved search results list (Tracker Search Results component) displayed on the **Project Home** page has been enhanced to reflect the column configuration changes you make in the tracker list view; that is, the column configuration of tracker search results on the **Project Home** page is an exact replica of the columns configured in the tracker list view. Depending on your tracker column configuration, you can now use custom fields as search criteria on your tracker or Project Home page.

Use the custom
field as search
criteria and
display it in the
results in
Trackers or on
project pages

Project Home Page - Tracker Search Results

Export selected list of artifacts: you can now export artifacts of your choice from the Trackers and Planning Board list views as well as the tracker search results page.

Association Viewer

The Association Viewer in code browser (introduced in 8.1) which shows associations graphically on a timeline now replaces the Associations listing in all the relevant pages (except discussion posts) of TeamForge.

Wherever an Associations tab is available, you'll now be able to toggle between a list and a trace association view. The associations displayed include both the TeamForge native objects as well as associations with objects from third party integrated tools. The behaviour is consistent across TeamForge with the following features:

- You can add/remove associations manually.
- The Trace view, which displays the traceability chain of associations in a graphical format includes an artifact's parent/child dependencies as well.
- The list of associations is paginated to better handle large lists of associations.

SCM

Unlike in the past, repositories created (or deleted) by site administrators and users with site wide role (with Integrations - SCM INTEGRATIONS permission) need no approval even on SCM servers that send such requests for approval to an administrator. Read more about the 'Requires Approval' feature *here*.
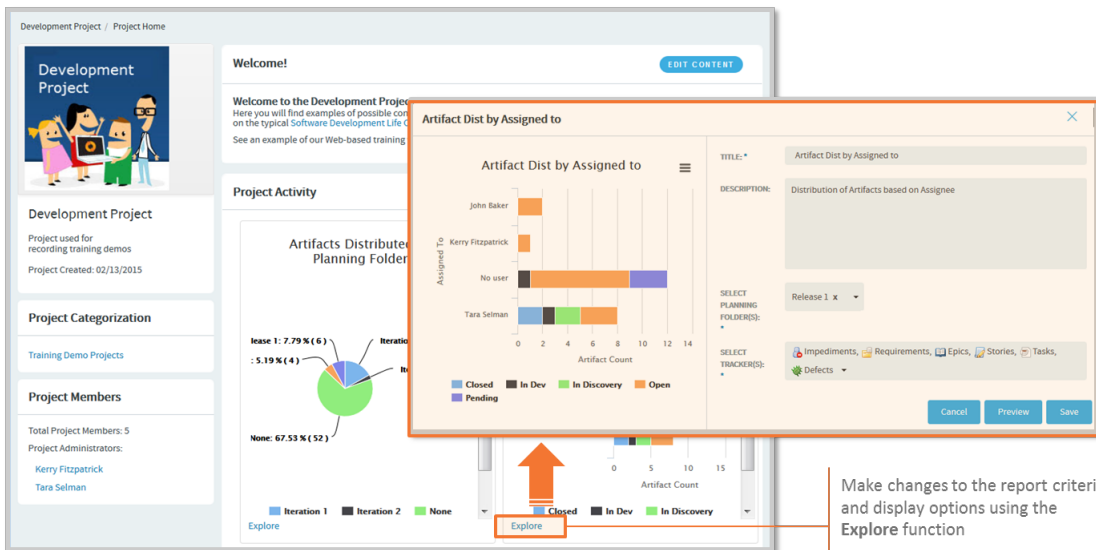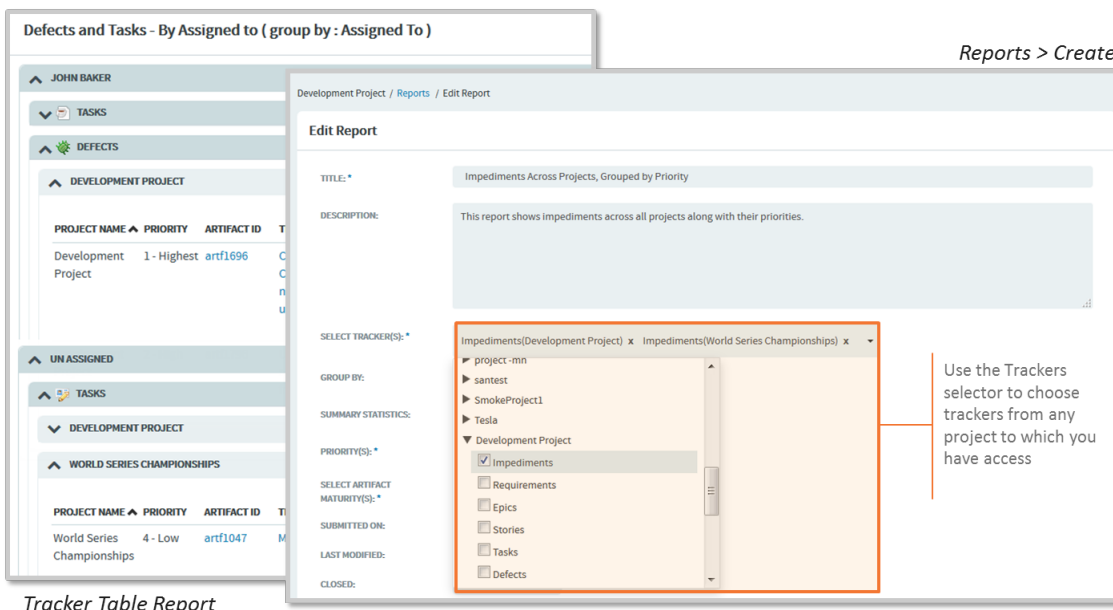
Internal code browser

Git specific changes:

- A new **Graph** tab has been added which allows you to visually see the branching and merging history within your repository.
- Deleted and rewritten branches that were protected by the History Protection feature can be seen on the Branches tab with options to resurrect or permanently delete the references.
- Support for Git submodules has been added. When submodule points to a repository in TeamForge, it can be directly browsed.
- When viewing a commit, the number of lines added/deleted for each file is now shown in the Summary view.
- The branch comparison feature has been enhanced to show the consolidated list of changed files along with the number of lines added/deleted in each of them.

Reports

Edit reports from Project Home page: users with the appropriate permission can now edit a report from the **Project Home** page itself using the new **Explore** link available at the bottom left corner of all the reports.

Make changes to the report criteria and display options using the **Explore** function

Cross project tracker reports: Tracker reports can now report on data not just across multiple trackers from a single project but even across multiple trackers from multiple projects. For more information, see *Table reports: Task and Tracker reports*.

Reports > Create

Use the Trackers selector to choose trackers from any project to which you have access

Tracker Table Report

Orchestrate

- Unlike earlier releases, with TeamForge Orchestrate 8.2 and later, you do not have to download and install the commit and tracker adapters manually. The TeamForge installer, by default, installs (or upgrades, in case you are upgrading to TeamForge 8.2) the commit and tracker adapters. In addition, TeamForge commit sources are created automatically for each project repository (Git and Subversion).
- Orchestrate installation process provides helpful diagnosis and solutions for common failure modes.
- Orchestrate installation provides resume functionality in case of failure.
- Orchestrate commit-to-work item association style now matches TeamForge. For example, [artf1234, JIRA-1234].
- Reporting API now accepts the "PIPELINE READ" permission for access.

site-options.conf

Obsolete `site-options.conf` tokens	The following site option tokens are obsoleted in line with CollabNet's product optimization policies. Also see: *CollabNet product end-of-life policy*.

- PERFORCE_CLIENT_DIR
- PERFORCE_GROUP
- PERFORCE_LICENSE_FILE
- PERFORCE_LOG_DIR
- PERFORCE_PORT
- PERFORCE_REPOSITORY_BASE
- PERFORCE_SERVICE_CMD
- PERFORCE_USER
- HELP_AVAILABILITY
- REMOTE_HELP_URL
- EXTERNAL_TOMCAT_INSTALL_DIR
- INTEGRATION_BUILTIN_TOMCAT
- ETL_BUILTIN_TOMCAT
- CEE_COMPATIBLE

/dev/urandom special file in JAVA_OPTS

The following JAVA_OPTS `site-options.conf` tokens use the `/dev/urandom` special file for better performance. Make sure these tokens in your `site-options.conf` file have -Djava.security.egd=file:/dev/urandom added to them.

- *ETL_JAVA_OPTS*
- *INTEGRATION_JAVA_OPTS*
- *PHOENIX_JAVA_OPTS*

☞ **Note:**

- Refer to the `/opt/collabnet/teamforge-installer/8.2.0.0/conf/site-options-dedicated.conf` file for the default values of these tokens.
- The JBOSS_JAVA_OPTS site-options token has all JVM parameters but -Xms1024m and -Xmx2048m hard-coded in the TeamForge core application. You need not manually configure any other parameter (such as -XX:MaxMetaspaceSize=512m -XX:ReservedCodeCacheSize=128M -server -XX:+HeapDumpOnOutOfMemoryError -Djsse.enableSNIExtension=false -Dsun.rmi.dgc.client.gcInterval=600000 -Dsun.rmi.dgc.server.gcInterval=600000) for this token in the `site-options.conf` file.

Scripts

The `indexupgrade.py` script is no longer available.

Nexus integration

Version 2.0 of the TeamForge-Nexus integration plugin is available. Upgrade your Nexus integration plugin. See *Upgrade the TeamForge-Nexus integration plugin* on page 739 for more information.

Bug fixes: When users log on to Nexus using CTF user credentials, the Nexus UI showed poor performance. This is fixed. You need to upgrade to "*CTF-Nexus-Integration-Plugin-2.0*" to get this issue fixed.

Caching with Memcached

Memcached caches Subversion (SVN) authentication and authorization information and serves the `mod_authnz_ctf` module's authentication and authorization requests thereby reducing the number of SOAP calls, which in turn results in less load on the TeamForge Application Server. See this *wiki page* for more information about Memcached.

See *Install Memcached* on page 571 for more information about how to enable caching with Memcached.

Issues resolved in CollabNet TeamForge 8.2

Here's a list of few noteworthy issues fixed in TeamForge 8.2.

Logs (vamessages.log and vasession.log)

The user login (both via Web user interface and SOAP) and SVN operations are now logged in the `vamessages.log` file as long as you have the log level set to DEBUG mode. See *Change the logging level on your site* on page 584 for more information about DEBUG and INFO modes.

☞ **Important:** The `vamessages.log` file subsumes the `vasession.log` file. The `vasession.log` file has been removed.

SOAP

- The createRepository method of the *ScmApp* service has been updated. Now, repository creation requests from all users but site administrators and SCM administrators go for approval.
- The moveDocument, deleteDocument and deleteDocumentByVersions methods of the *DocumentApp* service have been updated. Now, any locked document (including its versions) can be moved or deleted by none but the user that locked the document.
- The *PlanningApp* API is updated to generate a unique name for new planning folders even if a user tries to create multiple planning folders with the same name.

Platform

- Sites with huge user base and with specific site-options enabled witnessed relatively long JBoss startup time during upgrades, which is now resolved.
- JBoss startup issue due to serial incoming requests has been resolved.
- An issue that prevented relaying of GIT emails (GIT on a separate server) to the TeamForge application server has been resolved.

Reports

- You can now generate tracker reports filtered by any status of your choice.
- An issue that prevented Task Report from being exported has been resolved.
- The **Assigned To** field in TeamForge reports showed the user's login username. It now shows the user's full name instead.

SCM

- An encoding issue with the SCM Viewer that prevented `.docx` files from being downloaded has been resolved.
- An issue that prevented reuse of names of previously deleted repositories has been resolved.

Trackers

- A thread lock issue (with artifact count) has been resolved.
- Disabling and enabling flex fields on a tracker with a large number of artifacts resulted in a proxy timeout, which is now resolved.

- The 'Calculate Points' field of a tracker, if disabled, left the 'Points' field non-updatable. This is now resolved.

Planning folder (SOAP API)

In SOAP API, the validation to ensure if the start and end dates of a child planning folder are within that of its parent folder was so far set for planning folder creation alone. This validation has now been set for planning folder move as well.

Project Home Page

Resolved an issue that resulted in an 'Invalid URL' error while trying to access the project pages in the www folder.

IAF

- You now have the ability to exclude CCF integrated application when you create project templates from an existing project.
- Serial attempts to export an integrated application resulted in OOME (out of memory error), which is now resolved.

Security

- A Cross Site Request Forgery (CSRF) vulnerability has been fixed.
- The *LOGIN_ATTEMPT_LOCK* token is used to set the permissible number of unsuccessful login attempts after which the user account is automatically locked. Unlike in the past, you can now selectively disable this feature even if you have set the *REQUIRE_PASSWORD_SECURITY* site options token to true. Set a negative value (such as '-1') if you want to disable this feature. For more information, see *LOGIN_ATTEMPT_LOCK* on page 797.

  ☞ **Note:** With TeamForge 8.1 and earlier versions, setting *LOGIN_ATTEMPT_LOCK=-1* means the user account would be locked at the very first unsuccessful login attempt. This behavior has been removed from TeamForge 8.2 (and later).

UI/UEX

- An issue with branding customizations not showing up in a few pages has been resolved.
- Tracker statuses with an apostrophe such as "Won't fix" resulted in an issue that blocked Popup dialogs in the **View Artifact** page, which is now resolved.

Wiki

- An issue with using double byte characters in Wiki page names (during wiki page creation) has been resolved.
- Resolved an issue with Document and Wiki search that returned an 'Invalid search syntax' error when double quotes are used in the search text.

Orchestrate

- Switching Orchestrate integration off and on (exactly in that order) with TeamForge projects resulted in some issues with tracker and repository publishing activities, which is now resolved.
- Fixed issues with menus cutting off on Orchestrate web pages.
- Fixed a defect due to which Orchestrate-based associations to work items would be lost upon tracker/issue type change. Orchestrate now properly aggregates associations to a work item regardless of tracker/issue type changes.
- Fixed a defect due to which Reporting API output did not interpolate TeamForge links properly.
- Several installer and upgrade enhancements and bug fixes.

Documents

- An issue that prevented all but trusted users and moderators from posting to 'Moderated' discussion forums via email has been resolved.

- Now, any locked document (including its versions) can be moved or deleted by none but the user that locked the document.

Discussion forum

Unlike in the past, if you are not logged on to TeamForge, trying to download a document with its URL takes you to the TeamForge Login page instead of throwing the 'Invalid URL' error.

Known issues in CollabNet TeamForge 8.2

The following issues, including any workarounds we may have, are known to exist in the TeamForge 8.2 release.

File Releases

Exporting the list of **Reported Tracker Artifacts**, **Fixed Tracker Artifacts** or **Planned Tracker Artifacts** in File Releases with the 'Team' data is not possible. Trying to do so results in a blank page.

To work around this known issue, remove the 'Team' field from the 'Selected Columns' list and the export should succeed.

Team creation

When a user that is not a project's member but belongs to a user group which has Project Admin permissions,attempts to create a team, the user, by default, becomes the team owner. However, the team creation fails because the user is not a direct member of the project in this case.

To work around this issue, remove the user from the **Add Team Members** list on the **Create Team** page and designate a direct project member as the team's owner.

CLI reports on Project Home page

Reports are not properly rendered (reports go blank) in the **Project Home** page if you rearrange the CLI report components up or down. To work around this issue, refresh your browser every time you move a CLI report component on the **Project Home** page.

Discussion Posts from anonymous guest users

Discussion posts via the web UI from anonymous guest users are blocked with a 'session timed out' message.

Proxy timeout error while deleting a flex field on a tracker with a large number of artifacts

You might end up with a proxy timeout error while deleting one or more flex fields on a tracker with a large number of artifacts. Adjust (increase) your Apache timeout settings should you face such a situation.

Compatibility between JDBC drivers and database versions

CollabNet recommends using compatible JDBC drivers meant for specific database versions. For more information, see *this troubleshooting topic*.

Default value in user defined field

For existing artifacts, if you add a new user defined drop-down field, irrespective of the default value set, the field displays 'None' on the tracker artifact page.

CLI branding support for RHEL 6 with SELinux

There is an issue with adding or modifying CLI reports via the branding repository on RHEL 6 (32 and 64-bit environments) with the SELinux setup.

About box

Some of the 'Add-on' entries do not show up in the About box.

Simbel

User avatar fails to get imported via Simbel.

ViewVC exception

An exception occurs in the ViewVC mode when you replicate Subversion Edge or integrate it with TeamForge. To work around, use the TeamForge code browser.

TeamForge Avatar display issue on RHEL/CentOS 6.7

TeamForge Avatar image is not displayed properly post Review Board installation on RHEL/CentOS 6.7. Run the following commands to work around this issue:

```
yum erase python-imaging
yum install teamforge
/etc/init.d/httpd restart
```

Not able to restart specific services using the "collabnet" script

Restarting specific services such as "pgsql" using "collabnet" script is not possible. As a workaround, stop the service and start it again. For example:

Use these commands...	...instead of this command
`etc/init.d/teamforge_cli_server stop` `etc/init.d/teamforge_cli_server start`	`etc/init.d/teamforge_cli_server restart`
`/etc/init.d/collabnet stop pgsql` `/etc/init.d/collabnet start pgsql`	`/etc/init.d/collabnet restart pgsql`
`/etc/init.d/collabnet stop tomcatcs` `/etc/init.d/collabnet start tomcatcs`	`/etc/init.d/collabnet restart tomcatcs`

Reports

Jump to ID search (or in other words "Quick Search") throws an error when you search for a report with its ID.

☞ **Note:** Report ID is of the form "reportxxxx", where the value for xxxx (four digit number) can be found in the URL of the report in question.

Help you can download and print

You can grab the online help from this page if you'd rather view it in a reader or print it out.

Click here to open the TeamForge 8.2 documentation PDF file.