

# Current Integration Build (RELEASE CANDIDATE)<#>

See details of new features since 4.0.1 in the [What Is New In IDE 4.5](#) page. Installation instructions available [here](#).

## Integration build 200810162330 [Download#](#)

Update site is available for this build at <https://www.qnx.com/account/updates/foundry27/ide/tau/>

Over 100 bugs is fixed, some highlights:

- **Documentation:** Over 50 corrections in IDE users guide
- **Application Profiler:** Fixed export of time columns for csv, added user preference to pick format
- **Application Profiler:** Removed condition that profiling only starts after main is entered
- **Application Profiler:** Fixed hanging when entering table mode for sampling with many threads
- **Code Coverage:** Fixed code coverage sample project and cheat sheet instructions
- **Code Coverage:** Improve diagnostics when data is not valid or not in sync
- **Build:** Build templates are now properly filtered by language
- **Build:** Makefile projects comes with pre-defined qnx includes that allow proper auto-complete and navigation
- **Build:** Makefile projects - fixed environment variables for makefiles (CC,CXX, LD, etc)
- **Build:** Fixed convector to C/C++ Project
- **Launch & Debug:** Fixed launch shortcut for QNX Application
- **General:** Updated branding graphics
- **Target:** Restored Launch telnet and Launch remote photon commands
- **Memory Analysis:** Fixed leaks generation on exit
- **Memory Analysis:** Fixed incorrect handling of lowering realloc in lircheck and libmalloc\_g

Notes for Installing on top of 6.3.2 or earlier

- Application Profiler
    - Note: option use pipe would not work, because it requires qconn update ([available here](#)).
    - Note: to use Function Instrumentation you need library libprofilingS.a, which is available as a patch for 6.3.2 ([here](#)).
  - Memory Analysis
    - If you running unpatched 6.3.2 it won't find library in default configuration. If you want to install the patch, see [installation instructions](#). If you don't want to install the patch, you can go to Window->Preferences->QNX->Memory Analysis and reset Advanced preferences to use old library: enter libmalloc\_g.so as library and /dev/dbgm as device.
  - Doc for 6.3.2 OS
    - To see 6.3.2 OS doc in new IDE you need to install extra patch, see [installation instructions](#).
- 

## New & Noteworthy for Older Builds<#>

### Integration build 200809151800 [Download#](#)

Over 200 bugs is fixed, some highlights:

- **Documentation:** Updated IDE documentation for the new features
- **Application Profiler:** Added feature to pause-resume profiling on process running with function instrumentation
- **Application Profiler:** Fixed corrupted session during profiling on sh

- **Application Profiler:** Fixed data updating in realtime on Windows host
- **Code Coverage:** Fixed importing shared library data
- **Code Coverage:** Fixed importing code coverage using wizard
- **Code Coverage:** Fixed building with code coverage for standard make
- **Code Coverage:** Fixed data collection for gcc 3.3.5 on Momentics 6.3.2
- **Build:** Fixed building for Managed projects
- **Build:** Added feature to build in parallel in QNX Projects
- **Debug & Launch:** Improved Upload tab of launch configuration dialog (before was Download)
- **Debug & Launch:** Fixed bugs in source lookup
- **Target:** Improved drag & drop from and to Target Navigator view
- **Memory Analysis:** Minor usability improvements - renamed several labels on chart and column names
- **Memory Analysis:** Added scroll lock action to prevent view from refreshing
- **Memory Analysis:** Fixed zooming/filtering problem on Allocation chart
- **System Builder:** Fixed permission problem when Starting TFTP server on Linux & Neutrino
- **System Builder:** Fixed build file export
- **System Builder:** Fixed lock-up when importing BSP
- **System Information:** Target Navigator and Target File System Navigator - usability improvements

### Integration build 200807171037 [Download#](#)

- **Memory Analysis:** memory analysis tool now support new better library that replaces libmalloc, called librcheck. See instruction on how to install this library [here](#).
- **Memory Analysis:** Backtrace support now includes non x86 target and c++ code
- **Memory Analysis:** Optimized code of runtime library and events loading code in IDE
- **Memory Analysis:** Added support for new/delete operator as memory events
- **Application Profiler:** fixed bugs
- **Code Coverage:** fixed bugs - better integration with 4.2.x toolchain and better report generation
- **System Builder:** added feature to import directory, fixed bugs
- **Debug & Launch :** Download tab renamed to Upload since it is uploading file to target, fixed several UI issues and bugs with this tab
- **Debug & Launch :** Attach to a process with a tool now have general interface, you can attach to a process with debbuger, profiler or memory analysis tool.
- **Debug & Launch :** Launch dialog now allows to specify priority of a process
- **Debug:** Debug now will properly show exit code of the application
- **Target:** File system navigator now supports rename operation

### Integration build 200805161215 [Download#](#)

- **JTAG Debug:** New debug launch configuration to support "GDB Hardware Debug" through JTAG interface. Built-in supports for Abatron BDI2000 and Macraigor USB2Demon JTAG devices, with other devices support through self-defined hardware specific command sets.
- **Application Profiler:** Implement Export/Import from xml. Export to csv (excel).
- **Application Profiler:** Implemented dynamic data upload for instrumented binaries (previously only post-mortem worked).
- **Application Profiler:** Implemented Profile->Attach for instrumented binaries. In attach launch config same profiler trace file name should be used as by application that we attaching to.
- **Debugger:** Added support for gdb catchpoints

#### Knows Issues and Notes

- Application Profiler
  - Note: Internal db format for app profiler was changed, old session created by previous version would not load anymore

- Note: option use pipe would not work, because it requires qconn update ([available here](#)).
- Note: new dynamic upload mode does not work for short lived sessions (less than 2 seconds)

## Integration build 200804131800 [Download#](#)

- **Platform:** Neutrino hosted IDE. Note: Update Manager and SVN are not working due to missing SSL support in the Java VM.
- **Target Management:** New Terminal view with support for Telnet, SSH, various file transfer protocols, and full customization (fonts, colors, sizes)
- **Code Coverage:** Fixed some major bugs in Code Coverage, connected to specifics of code generated by contemporary C++ compilers.
- **OS documentations:** [Download](#) the "doc-plugin-080413.zip" file, extract the contents to your \$QNX\_TARGET folder. It will create a "docs" sub-folder under your \$QNX\_TARGET. Restart the IDE for the change to take effect.
- **Known Issue:** The Eclipse help may display empty navigation buttons and Table of Contents page. This is due to a conflict between the Eclipse help engine and the new Code Coverage plugin. While fixes are being worked on, a temporary work-around is to move the plugin "com.qnx.tools.ide.coverage.ui\_4.0.1.v200804101800" out of the "eclipse/plugins" folder, restart the IDE and you should see the help working again. Move the "com.qnx.tools.ide.coverage.ui\_4.0.1.v200804101800" back to the "eclipse/plugins" folder; next time the IDE starts the help and Code Coverage should both work.

## Integration build 200803131004 [Download#](#)

- **Subversion client:** SVN Team Provider and [Subversive](#) SVN connectors.
- **Code Coverage:** Support for gcc 4.2.1
- **Code Coverage:** Simplified coverage calculation: use line coverage instead of basic block coverage (basic block coverage numbers are still available in properties view for each particular component)
- **Code Coverage:** Stats for functions defined in header would be excluded from source file coverage
- **Code Coverage:** Added option for selective data collection in order to improve scalability
- **Code Coverage:** Added import of coverage data from remote target
- **Debug:** fixed problem with showing long long variables
- **Debug:** added feature to move line number of the existing breakpoint
- **System Profiler:** fixed label provider for event timestamps

## Integration build 200802271920 [Download#](#)

- **CDT:** CDT 4.0.3. [Bug Fixes](#)
- **Debugger:** New feature "Got To Address..." in disassembly view. Now you can view arbitrary address in disassembly view.
- **Debugger:** Source locator - added Absolute Path Mapping entry. For large project which is compiled on the same host as where IDE runs it is preferred way for source location, instead of default (Launch Configuration->Source tab)
- **Debugger:** Launch Configuration -> Debug tab: added option to allow set breakpoints using full file name. This is preferred way of setting breakpoints if you have many files with the same basename in the project. Only works when 3.3.5 or higher gcc is used or gdb 6.7 update 4 or higher (available to download from community site)
- **Debugger:** fixed navigation to breakpoint source code when breakpoint is set from gdb console (using address)
- **Debugger:** fixed post-mortem debugging: passing user defined shared libraries path to debugger (there are also related gdb fixes - gdb 6.7 update 5 or higher)

- **C/C++ Development:** fixed out of memory problem when loading huge binaries for navigation in C/C++ View
- **C/C++ Development:** fixed problem with process monitoring when process on target side deliberately closing stdout/stderr

### Integration build 200802041024 [Download#](#)

- **Debugger:** Updated Debug tab in launch configuration for Qnx Qconn and Qnx Pdebug launches to include debugger selection box and verbose mode option
- **C/C++ Development:** fixed stripping on non-x86 platforms
- **C/C++ Development:** fixed some bugs in container projects (all of them were in UI presentation of build configuration for Container Project, build target specification in particular.)
- **C/C++ Development:** more tuning of container project import process and the IDE responsiveness after import completion. Both issues are very important for big workspaces.

### Integration build 200801171800 [Download#](#)

- **Debugger:** Fixed (one of them) problem with UI deadlock and debugger UI unresponsiveness during launch and some other cases
- **Debugger:** Fixed hardcoded timeouts for suspend operation, default MI command timeout will be used (user configurable)
- **Debugger:** Fixed problems with closing dead debug sessions when error happens during the launch
- **Debugger:** Fixed problem with debugger hitting disabled breakpoints and not hitting enabled (in some cases)

### Integration build 200801131800 [Download#](#)

- **Debugger:** QDE Debugger now has an option "Verbose console mode" that shows details of interaction with command line debugger in gdb process console. This option can be enabled from launch configuration or by right clicking on the process in debug view.
- **Debugger:** GDB breakpoints actions page called "GDB Command Action" is added to the "breakpoint actions" property dialog box. This allows you to enter GDB commands - one command per line. Now you can define GDB command action and attach it to one or more breakpoints. When the breakpoint is hit, the defined GDB command(s) will run and the result will be shown in the GDB console. There is also a new global preference "GDB Command Action" page accessible from "Windows -> Preferences -> C/C++ -> Debug -> Breakpoint Actions". This preference page allows you to define one of more reusable GDB command actions that can be attached to a breakpoint later.
- **C/C++ Development:** Several major fixes connected to QNX container projects have been made:
  - speed of container project import was increased
  - resolved problem of workspace consistency after re-import of container project
  - fixed the workspace refresh problem after container project import (sometimes IDE did not refresh project view without restarting or even showed root container only (without components)) .
  - fixed the problem when the order of components in container project (and the build order) after import was wrongly changed.
  - fixed deadlock during the import operation for container project.

### Integration build 200712061800 [Download#](#)

- CDT 4.0.2 [Bug Fixed in 4.02](#)
- Update site available <https://www.qnx.com/account/updates/foundry27/ide/tau/>
- **Debugger:** Fixed base class members viewing in Variables view
- **Debugger:** Fixed shared library loading for gdb 6.7
- **Debugger:** Added View Memory action in Variable view

- **Application Profiler 2:** New Version of application profiler (Application Profiler 2) available for trials. See [Instructions](#).
- **Application Profiler 2:** Feature list: [What is New in Application Profiler](#)
- **Application Profiler 2:** Additional platform patch required: [libprofiling.so](#)
- **System Profiler:** The icons are back in the Trace Log view. A little bit of visual eye candy that we inadvertently lost a few releases ago.
- **System Profiler:** The event owner labels are now consistent throughout the IDE ... and can be configured via the preferences to show the Name, ID or both
- **System Profiler:** The summary page now shows you a bit more information about the log, such as how many dropped buffers occurred and the total number of events
- **System Profiler:** The Trace Log view now shows the 'real time' of an event. This is a new column and the start of the log file is based on the DATE property in the log file. Additional precision can be added by setting the start time value in the resource property.
- **System Profiler:** Address translation for interrupt IP events is now included. The log file must be matched with the binary files in your workspace for the address decoding to occur.
- **System Profiler:** Kernel log capturing is added to Application Launch configuration as an additional tool. Now you can start logging at the same time you start an application

## Integration build 20071116#

Download: [http://community.qnx.com/sf/frs/do/listReleases/projects.ide/frs.ide\\_integration\\_builds](http://community.qnx.com/sf/frs/do/listReleases/projects.ide/frs.ide_integration_builds)

- **Platform:** Eclipse platform 3.3.1.1 [Release Notes](#) [What is new in 3.3](#)
- **CDT:** [CDT 4.0.1](#)
- The new QNX launch and debug shortcut adopt to the new CDT launch and debug framework. It is enhanced to have the ability to detect and associate binaries to remote targets architecture. It is more user friendly since it does most of the works to fill in enough information in the launch configuration, and only requires user attention to the launch configuration dialog while multiple choices are available.
- Added converter from QNX Projects to Managed Make Projects
- **System Profiler:** A new Event Log view has been created that allows users to see the entire log file contents without the "segmented" display of the older Log view. This is accessible via the **Window > Show View > Other ... > QNX System Profiler > Full Trace Event Log**
- **System Profiler:** The export of log file data can be made into a CSV file. This is possible if you try and cut/paste a large selection from the **Full Trace Event Log** view or as a right click option on the \*.kev file **System Profiler > Export as CSV data**
- **System Profiler:** User event labels are now visible at any zoom level (they are no longer affected by 'pixelization') and are also configurable at both a per-logfile and global basis. This is an initial cut of the interface for selecting the event data. Feedback is welcome

See details in the [What Is New In IDE 4.5](#) page.

## IDE Tau Alpha I (milestone build)#

- **Platform:** [Eclipse 3.3 platform](#)
- **CDT:** [CDT 4.0](#)
- A managed build integration for the QNX tool chain
- **System Information:** The QNX System Information enhancements with malloc information view that include various charts to help you observe changes in memory usage (allocations and deallocations). Because it is important for you to know when and where memory is being consumed within an application, the Memory Analysis Tooling feature includes several views that use the trace information from the Memory Analysis session to help extract and visually display this information (to determine memory usage; allocation and deallocation metrics). The IDE includes these new views to help you observe changes in memory over time: Outstanding allocations, Allocation deltas, Deallocation deltas, and Outstanding allocation deltas. To access these views, select "Window"-->"Show View"-->"Other" from the menu in the IDE, and then select "QNX System Information"-->"Malloc Information", and click "OK".

- Note: For you to begin to view data on your graphs, you need to set logging for the target, and you need to select an initial process from the Target Navigator view.
- **Memory Analysis:** The Memory Analysis Tool lets you import and export trace data from a Memory Analysis session view. With the Memory Analysis Tool, you can:
  - Export session information in CSV or XML format. In the IDE, you can export your trace data session information from a Memory Analysis session view. When exporting memory analysis information, the IDE lets you export results in .csv format or in .xml format. Later, you can import the event-specific results into a spreadsheet, or you can choose to import session data back into a Memory Analysis tool to review the results. To export, click "File"-->"Export" from the menu, and then select "QNX"-->"Memory Analysis Data".
  - Import memory analysis sessions from trace file or XML format. In the IDE, you can import memory events and memory errors from a Memory Analysis session view. When importing memory analysis session information, the IDE lets you import from libmalloc trace file (generated on the target), or from previously exported session data in XML format. To import, click "File"-->"Import" from the main menu, select "QNX"-->"Memory Analysis Data", and then follow the wizard instructions.nl