History -- or What's a Partition and why it is Adapative?#

Some customers were asking us to run virtual machines, or even for ARINC 653 compliance, so they could better manage their cpu time and memory resources. "Not real time" we thought. So over a few beers, we asked them what problems they *really* wanted to solve. It came down to three basic scenarios:

1. **The untrusted application**: the ability to put a possibly nasty application into some kind of constrained box so it cannot hog resources to the point it will limit the rest of the system
2. **Selling Throughput**: the ability to divide a system by capacity and performance into several pieces, sell each piece to a different customer, and then charge for the capabilities (speed, throughput) of that piece
3. **Emergency Reserve**: the ability to reserve a chunk of cpu time and memory so an emergency recovery shell still work when the system becomes overloaded or degraded.

Hmm. We thought. Virtualization-like solutions can do that but they are inflexible and not real time. We figured we could divide, or partition, system resources in a way we can reconfigure on-the-fly and make it realtime too. So there.

The Tech#

The basic ideas are to

1. Provide guarantees of cpu time and memory available to applications.
2. Provide limits, or controls, on time and space consumption.
3. Allow for dynamic reconfiguration.
4. Still provide priority-preemptive thread scheduling.

There are two components, a fair share scheduler, actually a partition scheduler, and a memory control system called, "memory partitioning". They provide time and space partitions, respectively. The two kinds of partitions can be used independently, or used together. A third piece, partitioning of filesystem resources is Coming Real Soon Now(TM).

- Adaptive Partitioning Scheduler
- Memory Partitioning