



PTP 2.0 Release Review

March, 2008



Agenda



- New Features
- Non-Code Aspects
- APIs
- Defects
- Schedule
- Process
- Community
- IP Issues
- Project Plan

New Features



- Resource Manager Support
 - PTP 2.0 introduces the concept of resource managers
 - A resource manager is anything that controls access to computational resources on a (possibly remote) computer system
- Monitoring and Control of Multiple Systems
 - Starting with 2.0, PTP will now be able to monitor and control more than one system simultaneously
 - Machines under the control of a resource manager are combined into a single view
 - Jobs from all resource managers are combined into a single view
 - Jobs can be launched or debugged using any active resource manager

New Features (cont...)



- Parallel Language Development Tools
 - Barrier analysis tool detects potential deadlocks in MPI programs
 - New project wizards and templates for MPI and OpenMP projects
 - User interface improvements
- Parallel Debugger
 - The debugger has been redesigned to use the new Flexible Debug Hierarchy and eliminate dependency on the CDT debug framework
 - Scalability improvements
 - User interface enhancements

New Features (cont...)



- Performance Tools Framework
 - Provides a framework to allow external performance tools to interact with Eclipse and PTP
 - Adds a performance-analysis launch configuration system, analogous to the launch and debug configurations
 - Allows specification of performance analysis operations via an external, user-created XML file, extensions to the basic performance framework or a combination of the
 - XML GUI generator, allows specification of a GUI for selecting CLI-based performance analysis tool options via an XML file

New Features (cont...)



- Remote Tools
 - Adds a remote abstraction layer that allows tools to access remote services without requiring knowledge of the underlying transport
 - Differs from RSE by providing these services specifically for Eclipse tools, rather than for a user interface
 - Both RSE and a light weight ssh-based implementation (provided as part of PTP) can be used as the transport



New Features (cont...)

- Support for IBM's Parallel Environment (PE) and LoadLeveler
 - Allows job submission and monitoring of interactive PE jobs
 - Allows job submission and monitoring of batch jobs
 - Support for AIX and Linux systems
- TAU Performance Analysis Tool Plug-in
 - First implementation of Performance Tools Framework
 - Extends the performance framework with capabilities specific to the TAU performance analysis system
 - Selective instrumentation
 - PAPI performance counter selection
 - Profile storage
 - Visualization using perfdmf and paraprof



Non-code Aspects

- User documentation:
 - <http://eclipse.org/ptp/doc.php>
- Release plan:
 - <http://wiki.eclipse.org/index.php/PTP/planning/2.0>
- Release notes:
 - http://wiki.eclipse.org/PTP/release_notes/2.0
- Design documentation:
 - <http://wiki.eclipse.org/PTP/designs/2.x>
- Test plan:
 - <http://wiki.eclipse.org/PTP/testing/2.x>
- FAQ
 - <http://eclipse.org/ptp/faq.php>

APIs



- Most 1.x interfaces have been deprecated and removed
 - Will remain in 1.x branch
- New 2.0 interfaces are Eclipse quality
- All plug-in version numbers incremented from 1.1.1 to 2.0
- Internal APIs marked as such

APIs (cont...)



- Resource manager extension points:
 - `org.eclipse.ptp.core.resourceManagers`
 - Defines a new resource manager type. This type is used to create instances of resource managers from a factory.
 - `org.eclipse.ptp.core.rmConfigurations`
 - Used to specify the resource manager configuration wizard for a resource manager
 - `org.eclipse.ptp.launch.rmLaunchConfigurations`
 - Allows plug-ins to contribute a wizard page that is used to configure a new resource manager

APIs (cont...)



- Debugger extension points:
 - `org.eclipse.ptp.debug.core.parallelDebuggers`
 - Allows plug-ins to contribute new parallel debuggers
 - `org.eclipse.ptp.debug.ui.debuggerConfigurations`
 - Used to specify debugger configuration user interface
- Remote tools extension points:
 - `org.eclipse.ptp.remote.remoteServices`
 - Allows plug-ins to contribute remote service providers
 - `org.eclipse.ptp.remotetools.environment.core`
`.remoteEnvironmentControlDelegate`
 - Allows plug-ins to contribute new remote tools environments

APIs (cont...)



- Performance framework extension points:
 - `org.eclipse.ptp.perf.dataManager`
 - Defines action to be taken by Eclipse after performance data is generated
 - `org.eclipse.ptp.perf.workflow`
 - Includes a tool workflow xml file within an extension
 - `org.eclipse.ptp.perf.configurationTab`
 - Defines a custom tab for performance tool configuration (supplementing the GUIs that can be defined by the xml gui generator)

Defect Statistics



- 70 bugs opened
- 54 bugs resolved/closed
- 16 new bugs
- 0 P1 and P2 bugs outstanding

Schedule



- Original tentative release date was November 2007
- This was moved to 1Q2008 due to resourcing changes
 - Milestone 1: First preview release Jan 2008
 - Milestone 2: Final pre-release version Jan 17, 2008
 - Milestone 3: GA release March 31, 2008
- Milestone 1 achieved on Jan 23
- Milestone 2 on track
- Milestone 3 on track

Process



- This release has been developed using open, transparent, and inclusive processes
- This release has followed its charter principles
- The PTP project makes appropriate use of
 - Bugzilla
 - Mailing lists (ptp-announce@eclipse.org, ptp-dev@eclipse.org, ptp-user@eclipse.org)
 - Newsgroups (eclipse.technology.ptp)
 - Regular, monthly, conference calls
 - Wiki (<http://wiki.eclipse.org/PTP>)
- There was one committer election during the 2.0 release cycle
 - Wyatt Spear, University of Oregon

Community



- Monthly developer meetings
 - ~10 participants
- Outreach activities:
 - Tutorials:
 - OSCON 2007
 - SC 2007
 - ORNL (in conjunction with PTP workshop)
 - Exhibition booth at SC 2007 conference in Reno, NV
 - Developer workshop at Oak Ridge National Laboratory, May 2007
 - Face-to-face meetings/BOFs at EclipseCon and SC 2007

IP Issues



- All plugins contain appropriate about and license files
- All contributions (code, documentation, images, etc) have been committed by individuals who are either Members of the Foundation, or have signed the appropriate Committer Agreement; in either case, these are individuals who have signed, and are abiding by, the Eclipse IP Policy
- Major contributions have followed due diligence process
- All non-committer code contributions, including third-party libraries, have been documented in the release and reviewed by the Foundation's legal staff
- All contribution questionnaires have been completed
- IP Log available at
 - http://www.eclipse.org/ptp/project-info/project_log.csv

Project Plan



- Draft 2.1 plan available at:
 - <http://wiki.eclipse.org/index.php/PTP/planning/2.1>