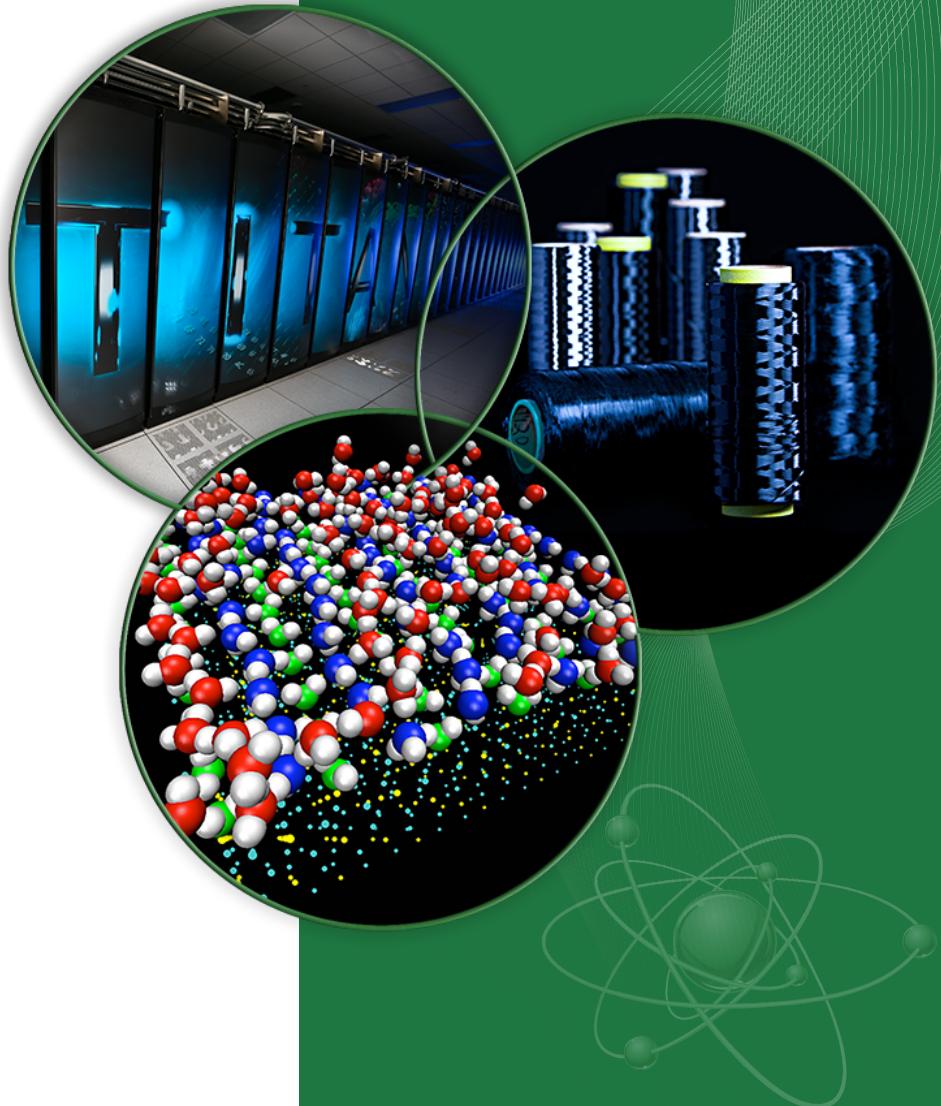


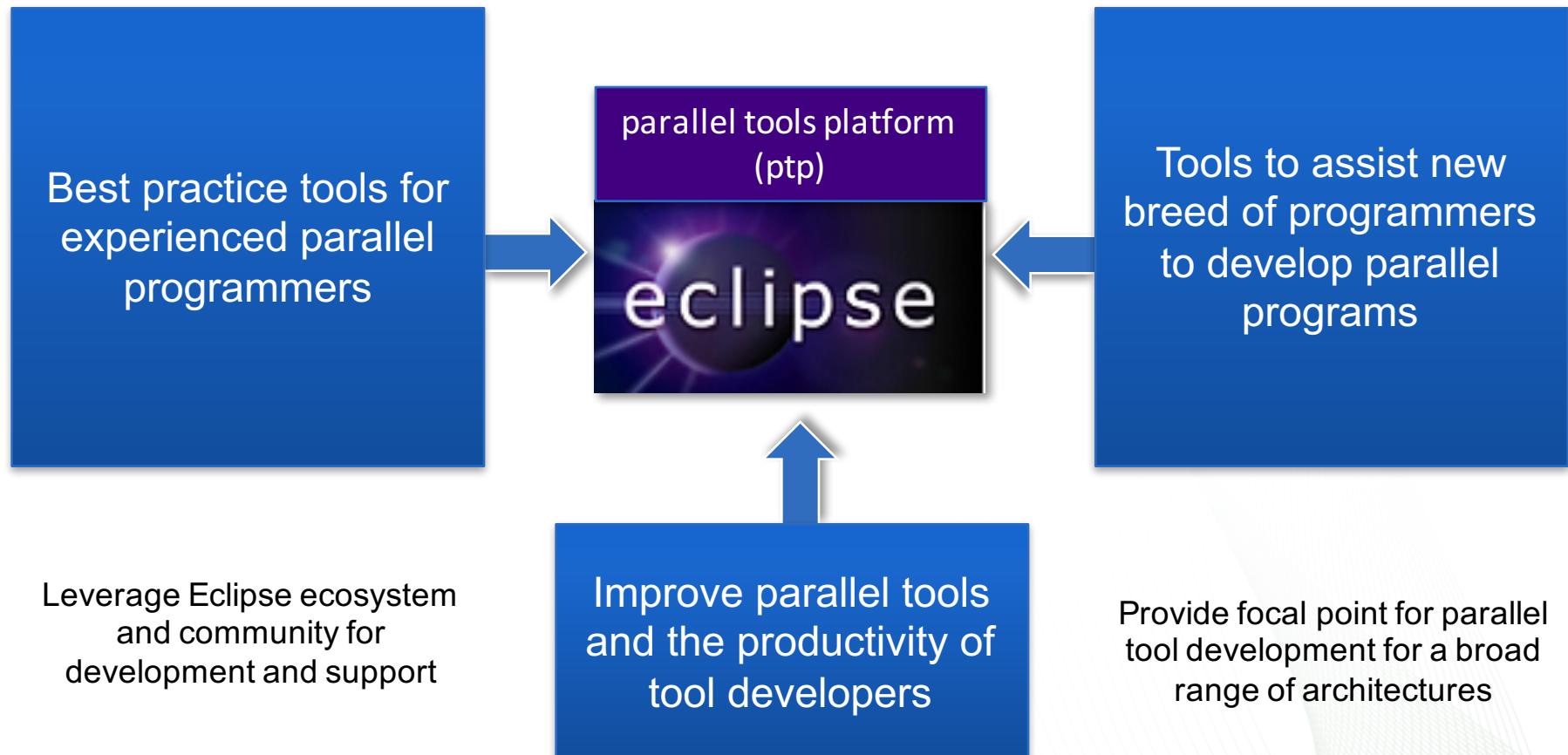
Eclipse for Science

How the Parallel Tools Platform can enhance the development of scientific applications



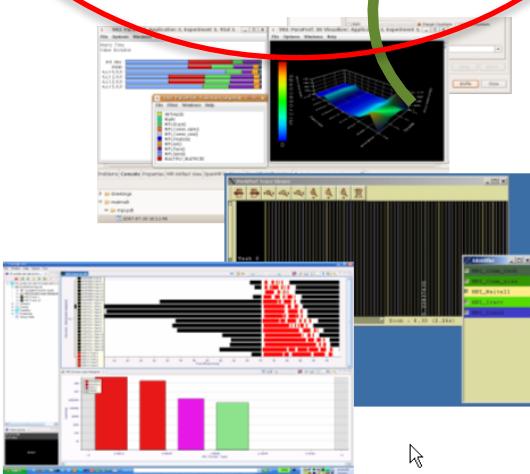
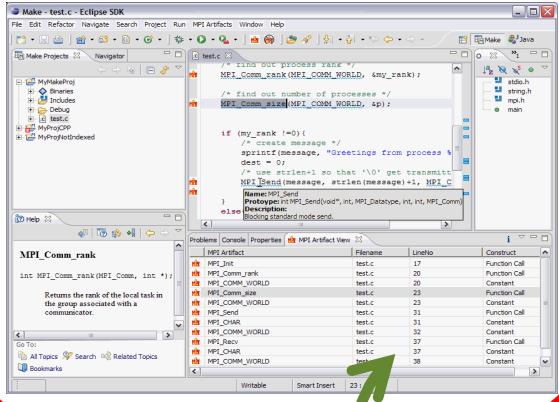
Parallel Tools Platform

Enabling Parallel Application Development



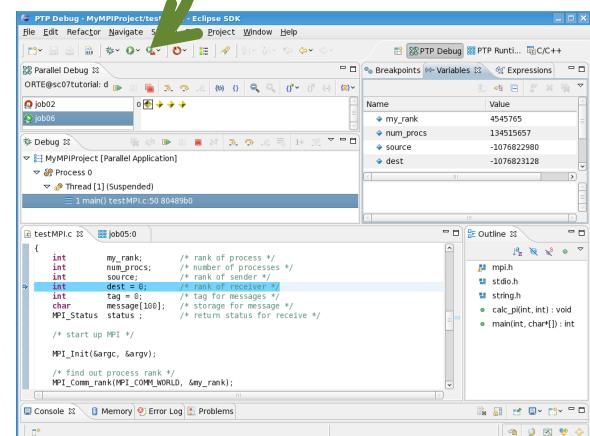
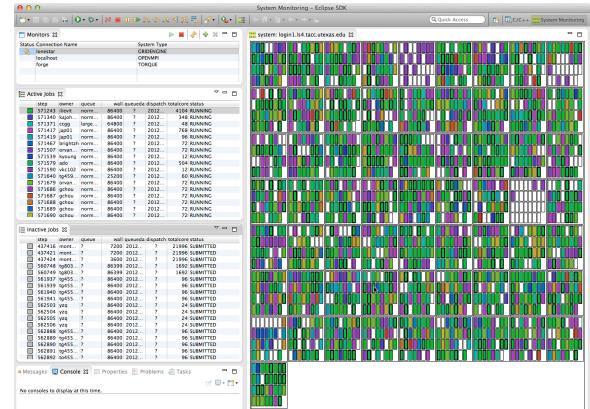
PTP Application Development Cycle

Coding & Static Analysis



Dynamic & Performance Analysis

Application Execution



Application Debugging

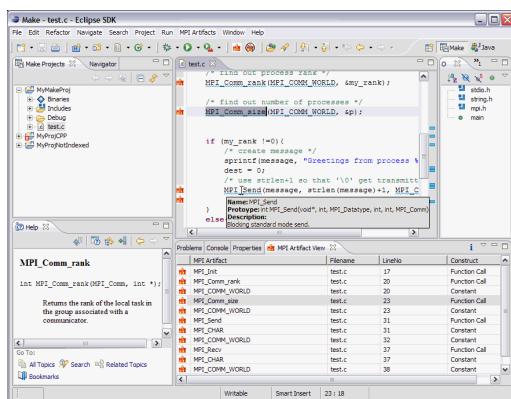
OAK RIDGE
National Laboratory

Coding & Static Analysis

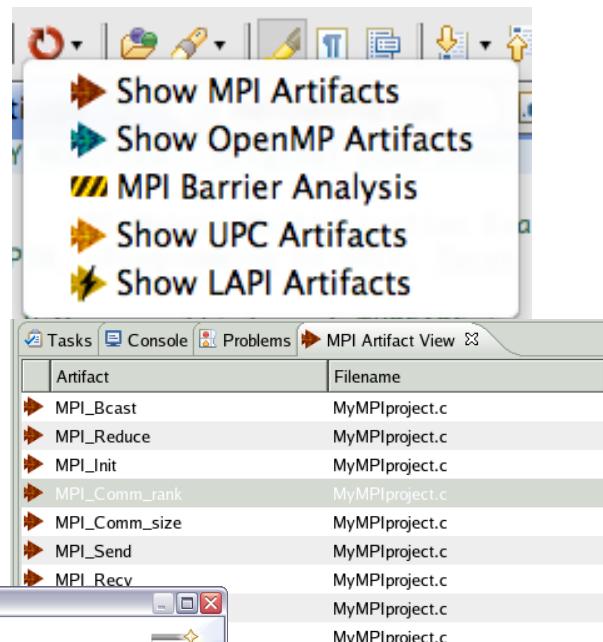
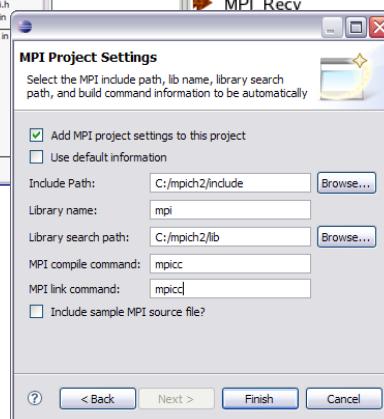
- Eclipse provides a wide variety of coding assistance tools
 - Project management, Editing and formatting, Navigation, Advanced searching, Refactoring, Version control
- C/C++ Development Tools (CDT)
 - Standard (Makefile) and managed builders, Support for arbitrary toolchains, Visual debugging using GDB, High level views (outline view, call hierarchy, type hierarchy, include browser), Refactorings
- Parallel Tools Platform (PTP)
 - Fortran, New project wizards (MPI, OpenMP) Content Assist, Hover help, Built-in API descriptions (MPI, OpenMP, LAPI, UPC), Location of parallel “artifacts” in code (MPI, OpenMP, PAMI, and UPC), Barrier analysis, Deadlock detection
- Python Development (PyDev)
 - Code completion, type hinting, refactoring, debugging, interactive console, unittest, code coverage, Django integration

Coding & Static Analysis

- Assistance tools to increase productivity of parallel programmers
 - New project wizards (MPI, OpenMP)
 - Content Assist (command/API completion), hover help, built-in API help descriptions in an html help “view” (MPI, OpenMP, LAPI, UPC)
 - Location of parallel “artifacts” in code: MPI, OpenMP, and UPC



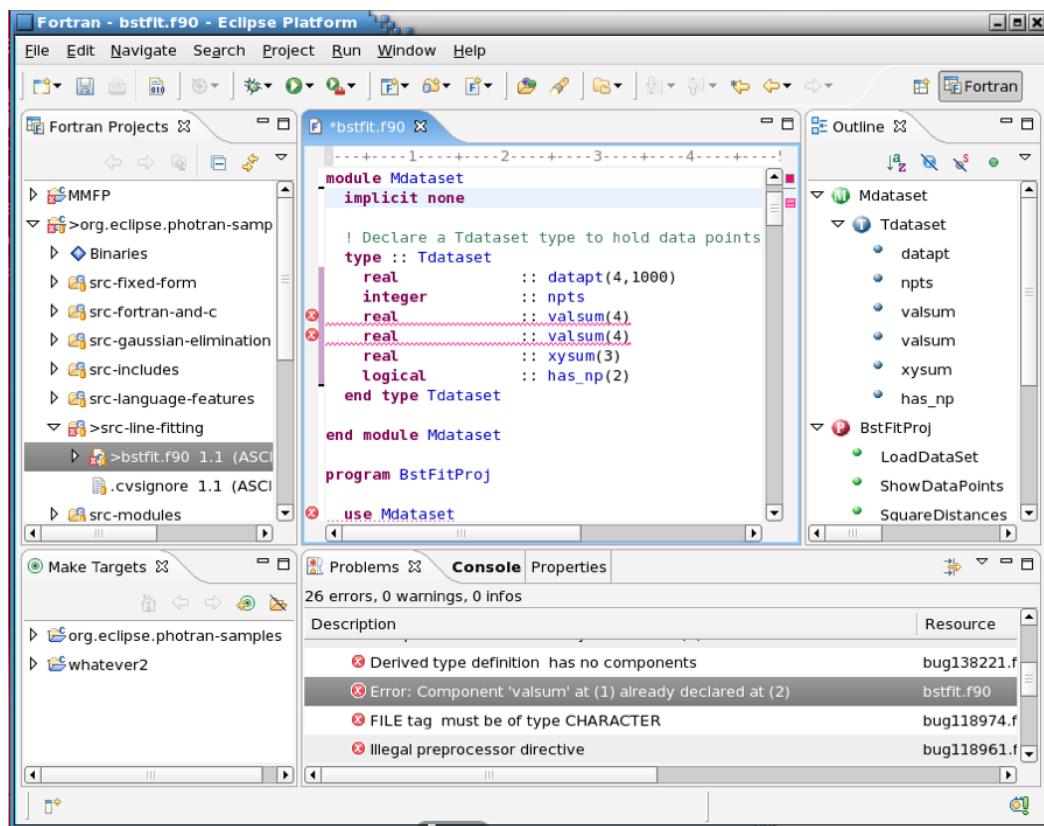
A screenshot of the Eclipse IDE interface showing the same MPI code as above, but with annotations. Red arrows point to the MPI_Comm_rank and MPI_Comm_size calls, with text boxes explaining their purpose: 'Returns the rank of the local task in the group associated with a communicator.' and 'Returns the size of the group associated with a communicator.'



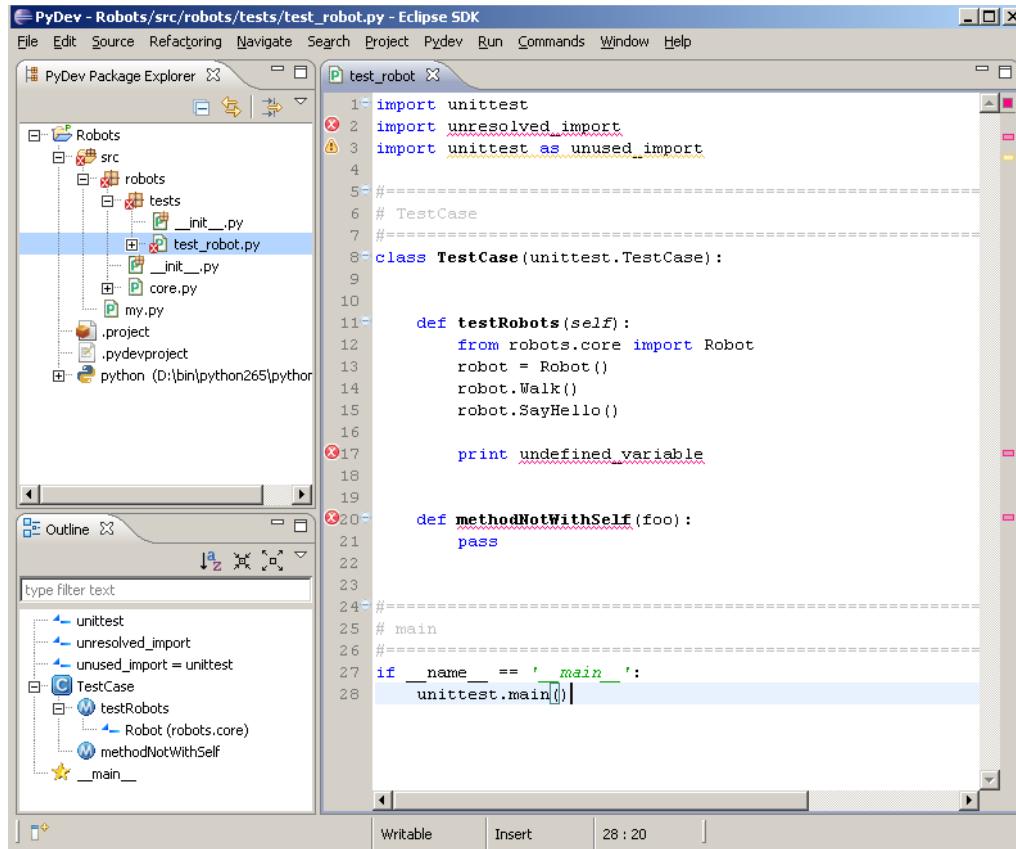
PHASE 3
HPCs
PERCS

Fortran Development Tools

- Photran features:
 - Supports Fortran 77-2008
 - Syntax-highlighting editor
 - GUI interface to *gdb*
 - Makefile-based compilation
 - Compiler error extraction
 - Outline view
 - Open declaration
 - Fortran refactorings
 - C preprocessor support



Python Development

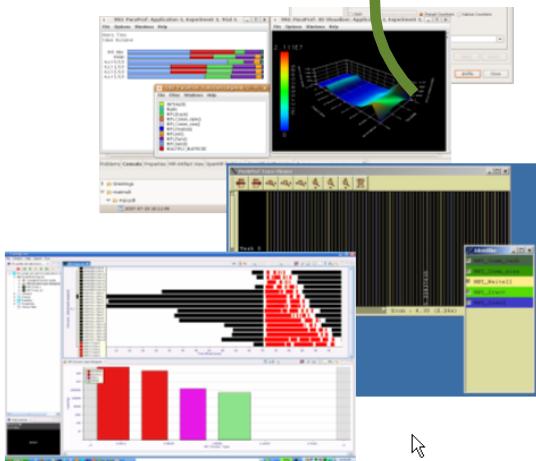


- PyDev is a Python IDE for Eclipse
- Create/manage Python modules
- Full array of Eclipse editing features for Python
- Python debugger
- Interactive console with Python interpreter
- Integration with Python unittest and code coverage modules

PTP Application Development Cycle

Coding & Static Analysis

A screenshot of the Eclipse IDE interface. The top window shows a C code editor for a file named 'test.c' with MPI library calls highlighted. Below it is a 'MPI Artifact View' tool showing function call details. A green arrow points from this interface down towards the bottom left.



Application Execution

A screenshot of the 'System Monitoring - Eclipse SCM' tool. It displays a grid of process status information and monitoring metrics. A red circle highlights the monitoring interface, and a green arrow points from the bottom right towards it.

A screenshot of the Eclipse IDE's 'PTP Debug' perspective. It shows a stack trace for a suspended thread, variable values, and a code editor for 'testMPI.c'. A green arrow points from the bottom left towards this interface.

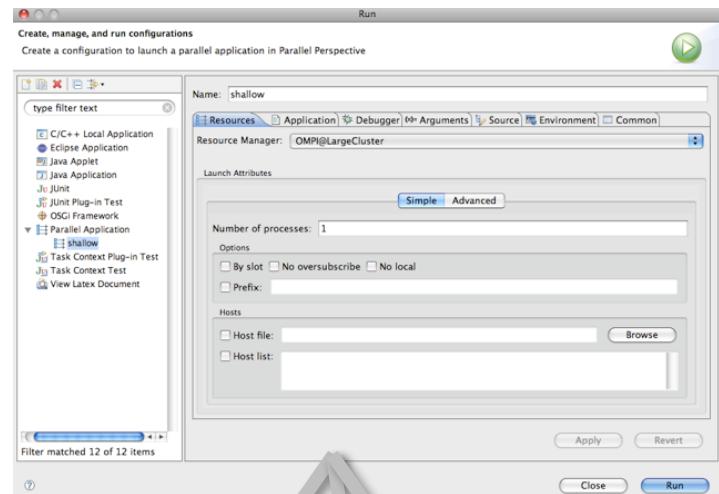
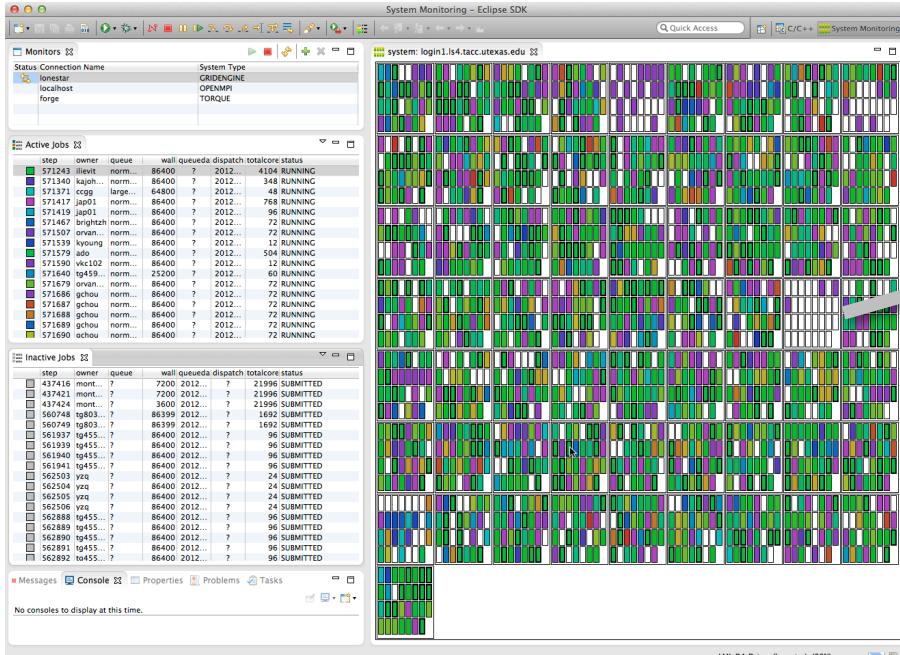
Dynamic & Performance Analysis

Application Debugging

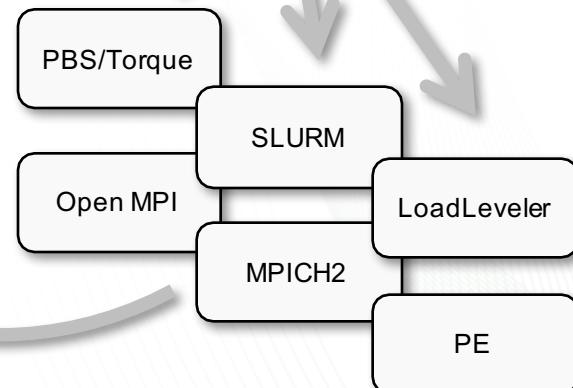
OAK RIDGE
National Laboratory

Application Execution

- Launching & Monitoring



- Improves visibility into target system
- Single point of interface for launching and control
- Manages interaction with different runtime systems and job schedulers



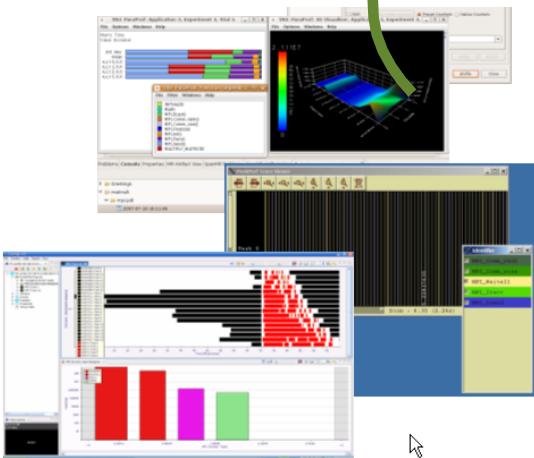
Application Execution

- Target Configuration Framework
 - Extensible framework for launching & monitoring
 - System and node status information
 - Job status (e.g. position in queue) & application status
 - Job submission & control
 - Debugger launch
 - Configuration files to support different resource managers
 - Job schedulers (LoadLeveler, PBS, Torque, SLURM, GridEngine)
 - Interactive runtimes (e.g. PE, Open MPI, MPICH2, MVAPICH)
 - Systems (AIX, Linux, Power, x86, BG/Q, Cray)
 - Local or remote system support
 - Command-line tools executed locally or via ssh connection

PTP Application Development Cycle

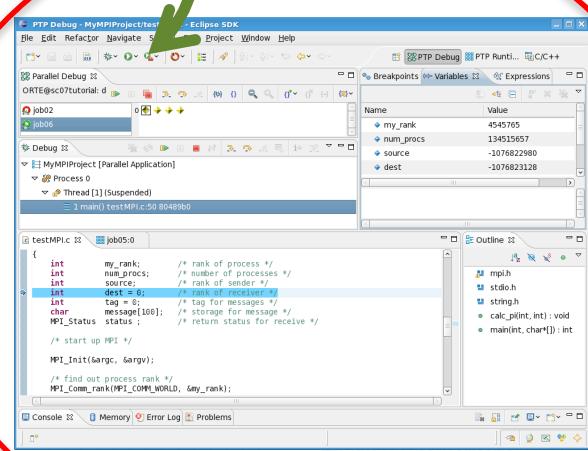
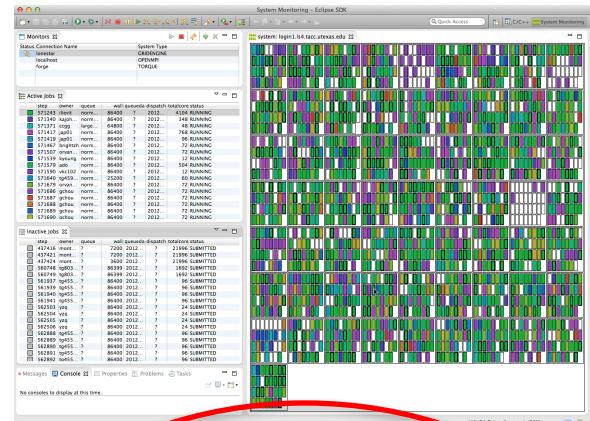
Coding & Static Analysis

A screenshot of the Eclipse IDE interface. The main window shows a C file named `test.c` with MPI code. The code includes functions like `MPI_Comm_rank`, `MPI_Comm_size`, and `MPI_Send`. Below the code editor is the `MPI Artifact View` tool, which lists various MPI operations and their details.



Dynamic & Performance Analysis

Application Execution

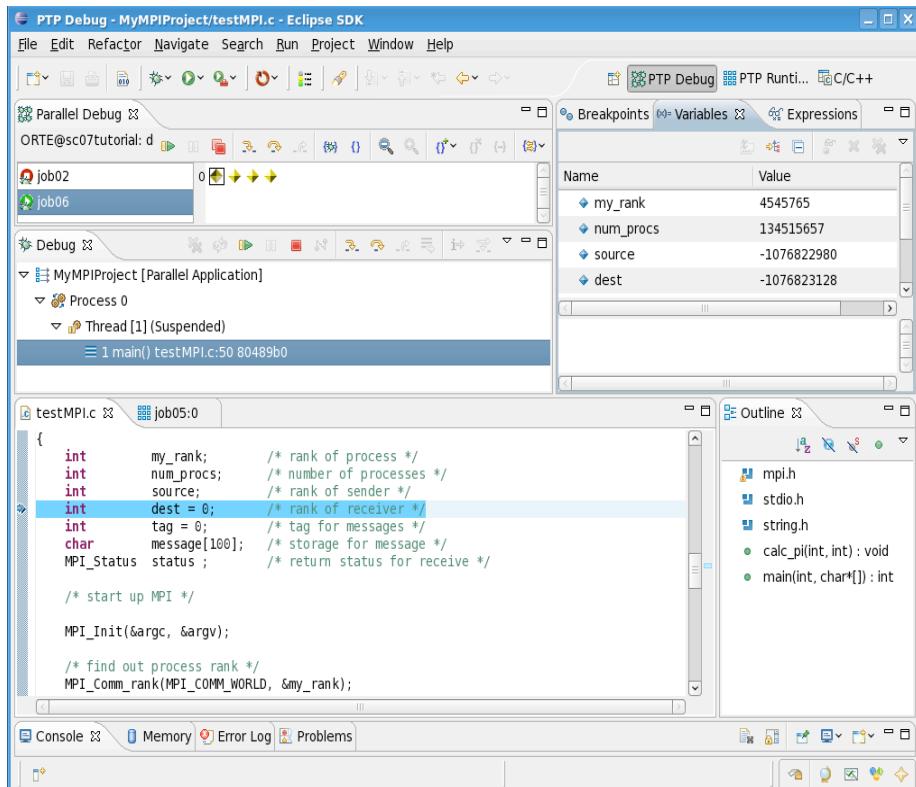


Application Debugging

OAK RIDGE
National Laboratory

Application Debugging

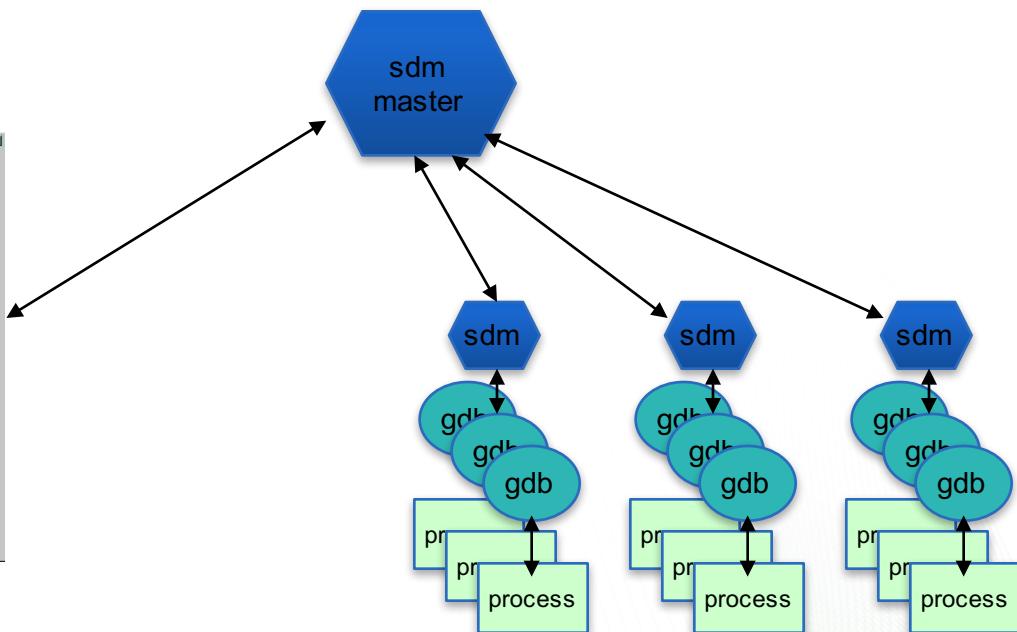
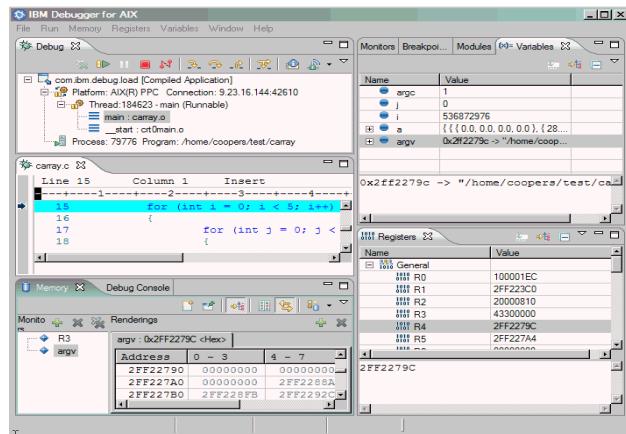
- PTP Parallel Debugger



- Mid-scale integrated debugger
- Tightly integrated with Eclipse
- Supports debugging multiple jobs simultaneously
- Utilizes backend debugger (e.g. gdb) for low level operations
- Targeted at SPMD programming models
- Supports mixed MPI & thread debugging
- Single process and group operations
- Platform for building new debugging paradigms

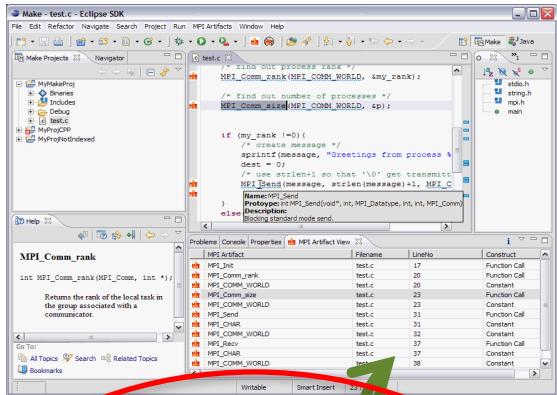
Application Debugging

- Scalable debugger using multicast reduction network
- Integrated with PTP and launched using target configurations
- Supports basic debug commands
- Uses gdb on backend

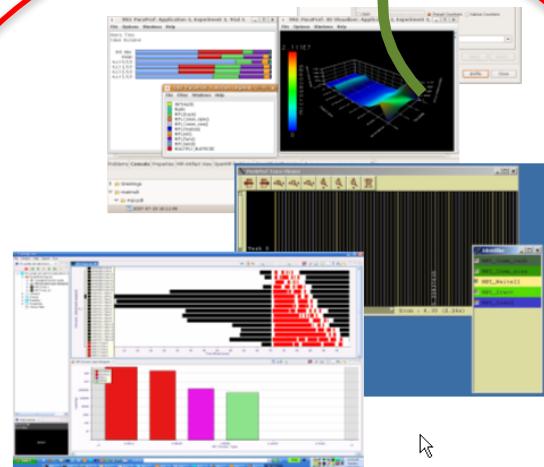
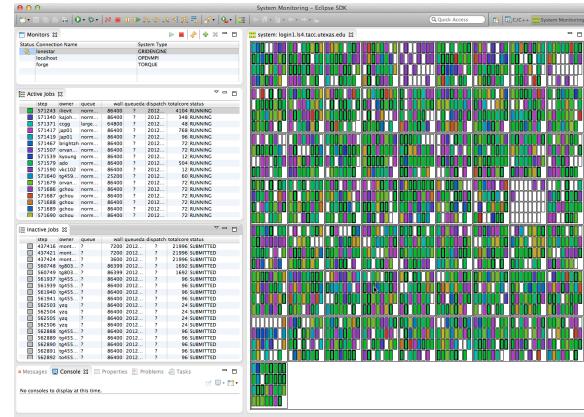


PTP Application Development Cycle

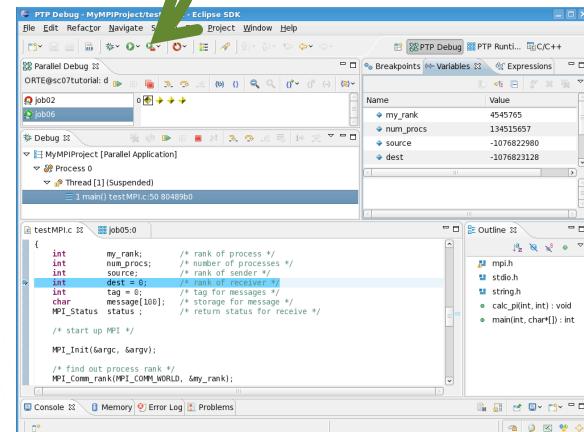
Coding & Static Analysis



Application Execution



Dynamic & Performance Analysis

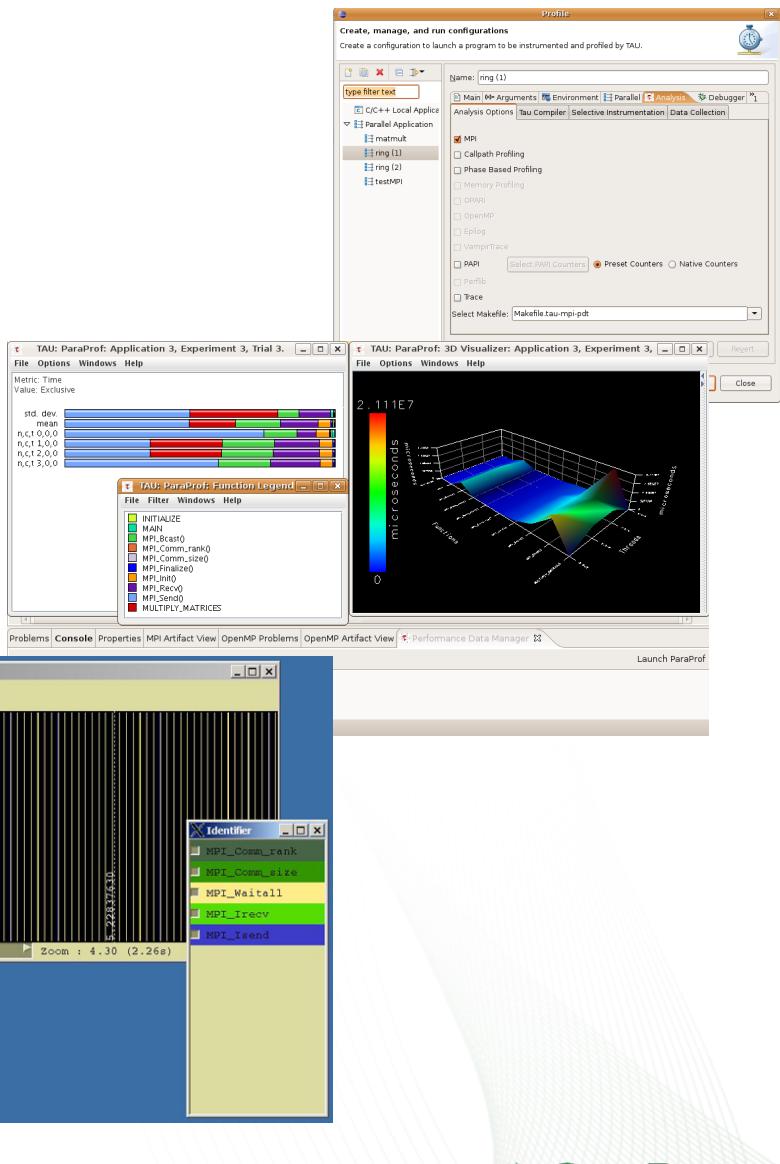


Application Debugging

OAK RIDGE
National Laboratory

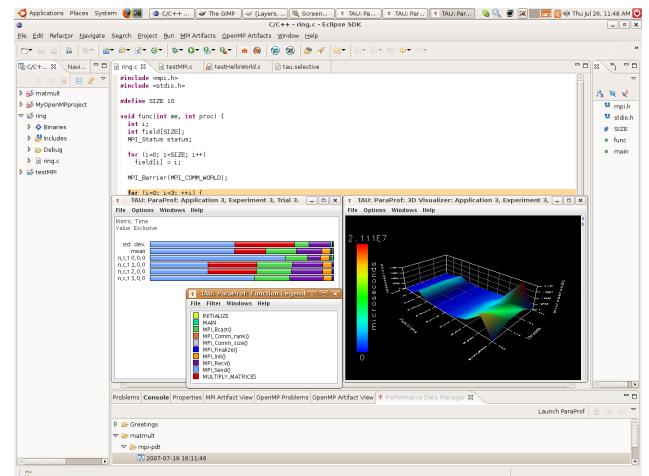
Dynamic & Performance Analysis

- Dynamic Analysis Tools
- Perform analysis on the running application using external tools
- Generate results that must be brought back into Eclipse as part of the development workflow
- May require external tool for visualization or other purposes

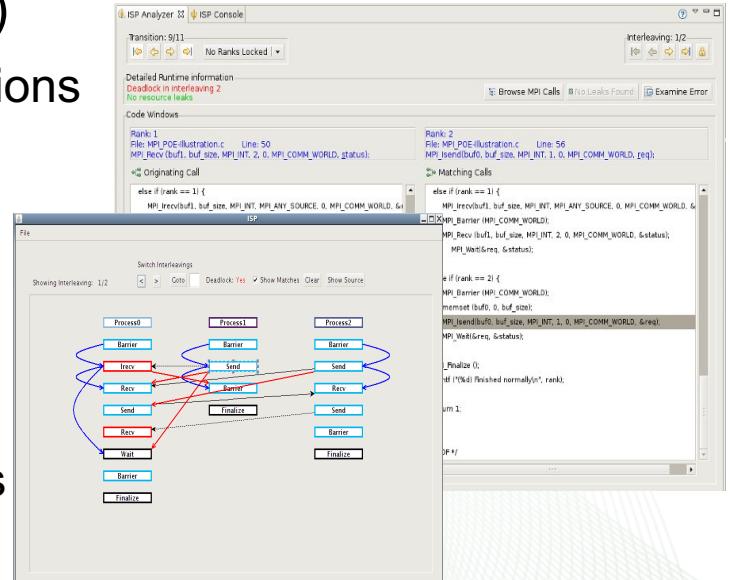


Dynamic & Performance Analysis

- Tuning and Analysis Utilities (TAU)
 - Instrumentation and transparent re-build of application executable
 - Execution of profiled application and collect performance data
 - Performance data visible in UI
 - Launches paraprof visualization client from Eclipse



- Graphical Explorer of MPI Programs (GEM)
 - Formal Dynamic Verification of MPI Applications
 - Detects all deadlocks, assert violations, MPI object leaks, and default safety properties
 - Matches sends and receives
 - Allows post-verification review of highlighted bugs
 - Works with a variety of MPI implementations



Online Information

- **Information about PTP**
 - Main web site for downloads, documentation, etc.
 - <http://eclipse.org/ptp>
 - Developers' wiki for designs, planning, meetings, etc.
 - <http://wiki.eclipse.org/PTP>
 - Articles and other documents
 - <http://wiki.eclipse.org/PTP/articles>

Community

- **PTP Mailing lists**
 - Major announcements (new releases, etc.) - low volume
 - <http://dev.eclipse.org/mailman/listinfo/ptp-announce>
 - User discussion and queries - medium volume
 - <http://dev.eclipse.org/mailman/listinfo/ptp-user>
 - Developer discussions - higher volume
 - <http://dev.eclipse.org/mailman/listinfo/ptp-dev>