



parallel tools platform

<http://eclipse.org/ptp>

# Improving the Eclipse Parallel Tools Platform in Support of Earth Sciences High Performance Computing

Jay Alameda

National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign

Wyatt Spear

Department of Computer and Information Science, University of Oregon

Brian F. Jewett

Department of Atmospheric Sciences, University of Illinois at Urbana-Champaign

Software Engineering Assembly Conference 2013

3 April 2013

# Acknowledgements

- ✦ Portions of this material are supported by or based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under its Agreement No. HR0011-07-9-0002, the United States Department of Energy under Contract No. DE-FG02-06ER25752, the Blue Waters sustained petascale computing project, which is supported by the National Science Foundation under award number OCI 07-25070, and the SI2-SSI Productive and Accessible Development Workbench for HPC Applications, which is supported by the National Science Foundation under award number OCI 1047956
- ✦ The SI2-SSI team is lead by Jay Alameda (NCSA), Greg Watson (IBM), Steven Brandt (LSU), and Allen Malony (U Oregon). Team members and senior personnel include Beth Tibbitts (IBM), Ralph Johnson (U Illinois), Chris Navarro (NCSA), Sameer Shende (U Oregon), Wyatt Spear (U Oregon), Brian Jewett (U Illinois), Galen Arnold (NCSA), and Rui Liu (NCSA)

# Outline

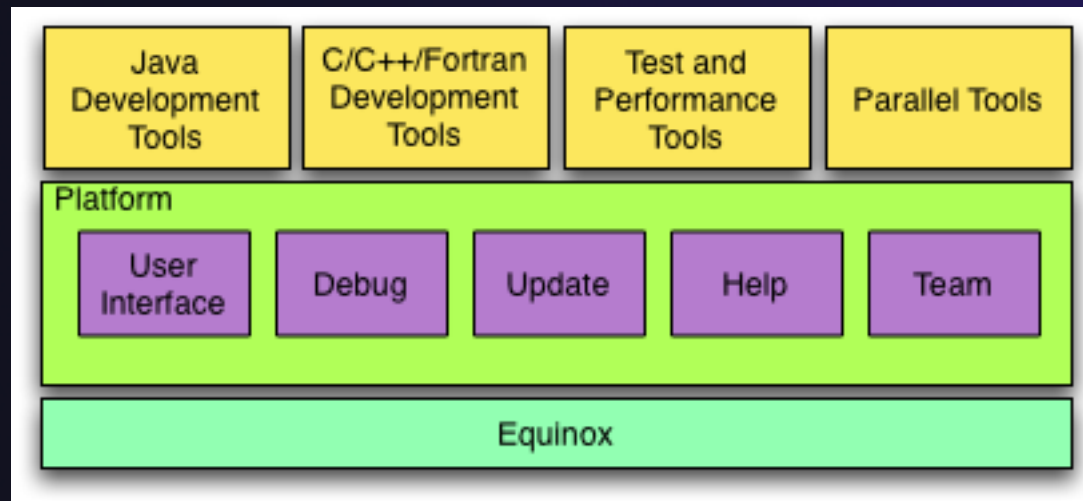
- ★ Overview of Eclipse and Eclipse Parallel Tools Platform (PTP)
- ★ Motivation for Workbench for High Performance Computing (WHPC)
  - ★ Improvements to Eclipse PTP
- ★ Software Engineering Practices Enabled by Eclipse PTP
  - ★ Code visibility
  - ★ Multi-system build management
  - ★ Performance tuning
  - ★ Source code control
  - ★ Issue Tracking
  - ★ Documentation
  - ★ Earth Science/Weather code example
- ★ Eclipse PTP Resources

# Outline

- ★ Overview of Eclipse and Eclipse Parallel Tools Platform (PTP)
- ★ Motivation for Workbench for High Performance Computing (WHPC)
  - ★ Improvements to Eclipse PTP
- ★ Software Engineering Practices Enabled by Eclipse PTP
  - ★ Code visibility
  - ★ Multi-system build management
  - ★ Performance tuning
  - ★ Source code control
  - ★ Issue Tracking
  - ★ Documentation
  - ★ Earth Science/Weather code example
- ★ Eclipse PTP Resources

# What is Eclipse?

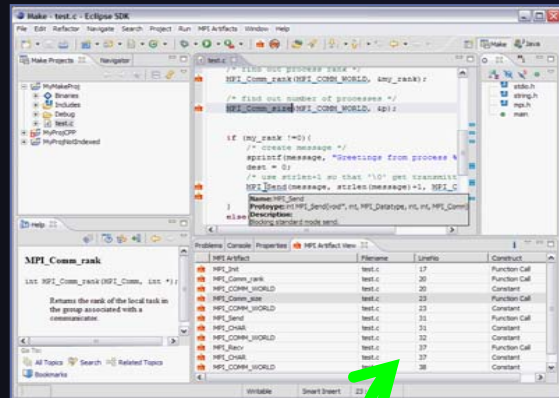
- ★ A vendor-neutral open-source workbench for multi-language development
- ★ A extensible platform for tool integration
- ★ Plug-in based framework to create, integrate and utilize software tools



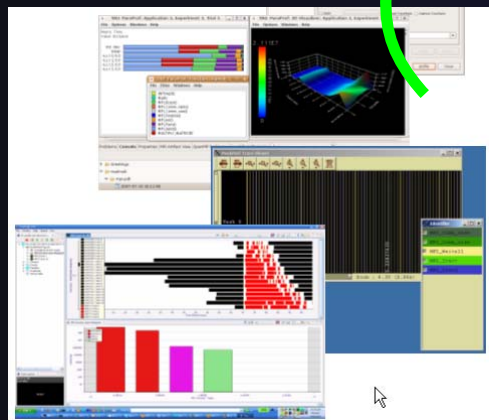
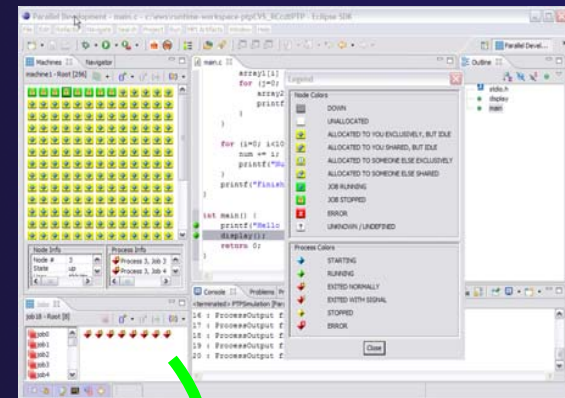


# Eclipse Parallel Tools Platform (PTP)

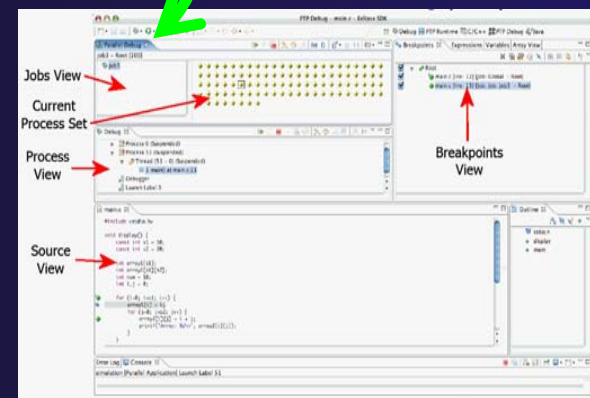
Coding & Analysis



Launching & Monitoring



Performance Tuning



Debugging



# Outline

- ★ Overview of Eclipse and Eclipse Parallel Tools Platform (PTP)
- ★ Motivation for Workbench for High Performance Computing (WHPC)
  - ★ Improvements to Eclipse PTP
- ★ Software Engineering Practices Enabled by Eclipse PTP
  - ★ Code visibility
  - ★ Multi-system build management
  - ★ Performance tuning
  - ★ Source code control
  - ★ Issue Tracking
  - ★ Documentation
  - ★ Earth Science/Weather code example
- ★ Eclipse PTP Resources

## Motivation for Workbench for High Performance Computing (WHPC)

- ★ Stable, portable platform for tool development
  - ★ Focus on tool functionality, manage rapid evolution of HPC platforms
  - ★ Encourage consistent tool look and feel
  - ★ Support for HPC application development practices
- ★ Why Parallel Tools Platform?
  - ★ High potential to meet needs of a WHPC.
  - ★ Target next generation of HPC developers growing up with IDEs (Eclipse, Visual Studio, ...)
  - ★ Need to cultivate community of users!

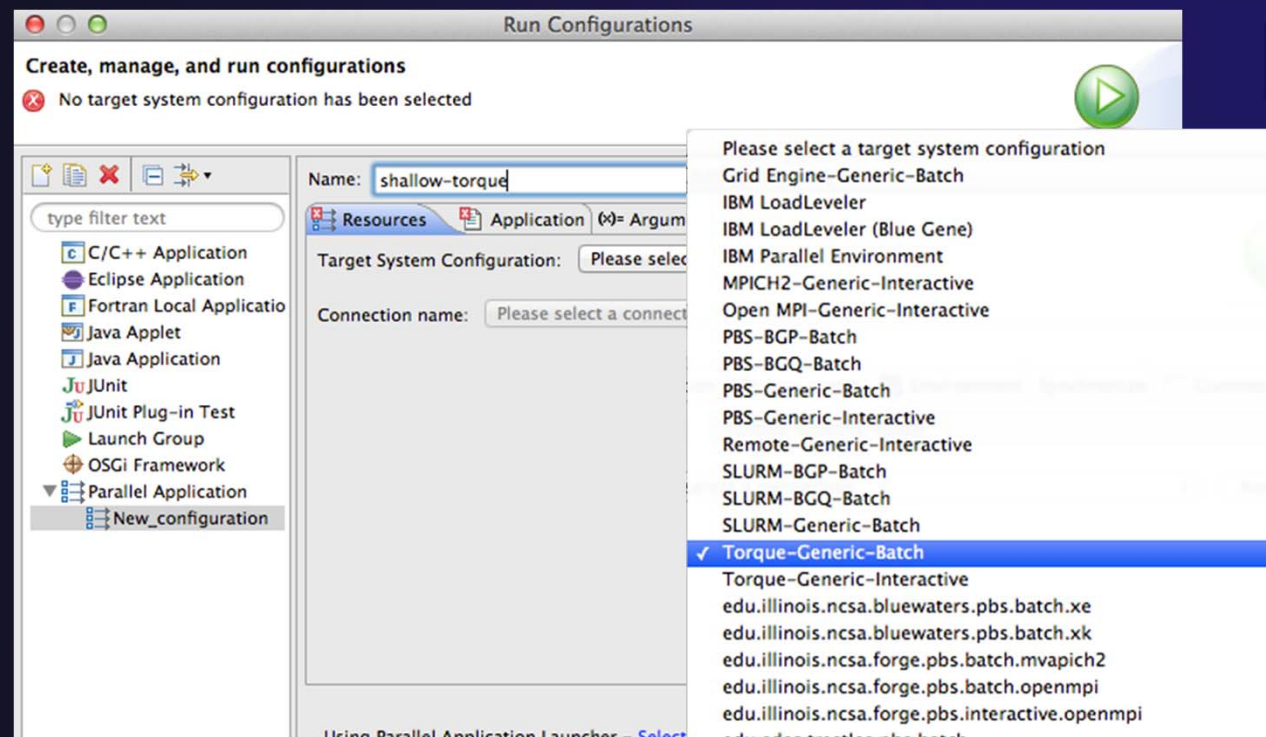


# Improvements

- ✦ Work within Eclipse release cycle
  - ✦ Major (API-breaking) improvements with coordinated June release
    - ✦ Last major release Eclipse 4.2 “Juno” released June 27, 2012
  - ✦ Minor enhancements and bug-fixes with two coordinated service releases in September and February
    - ✦ Eclipse 4.2 SR2 Released March 1, 2013
- ✦ Foci of improvements
  - ✦ Improve usability
  - ✦ Improve productivity

# Significant Recent Improvements

- ★ User-configurable machine configuration
  - ★ Wide variety of configurations now available:
  - ★ Documentation, tutorial at
    - ★ [http://wiki.eclipse.org/PTP/designs/Resource\\_Manager\\_Configuration](http://wiki.eclipse.org/PTP/designs/Resource_Manager_Configuration)



# Scalable System Monitoring

★ System view

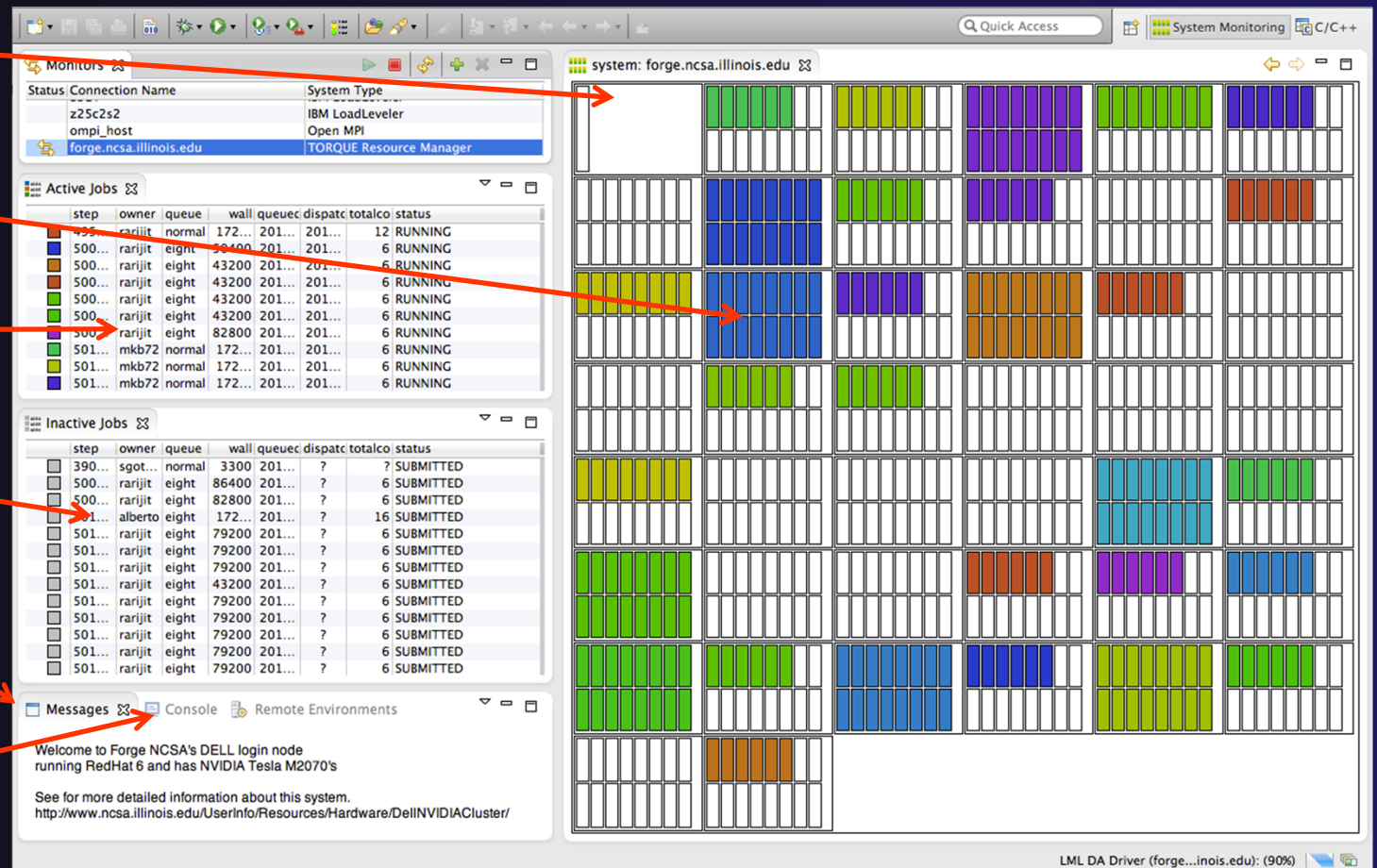
★ Jobs running on system

★ Active jobs

★ Inactive jobs

★ Messages

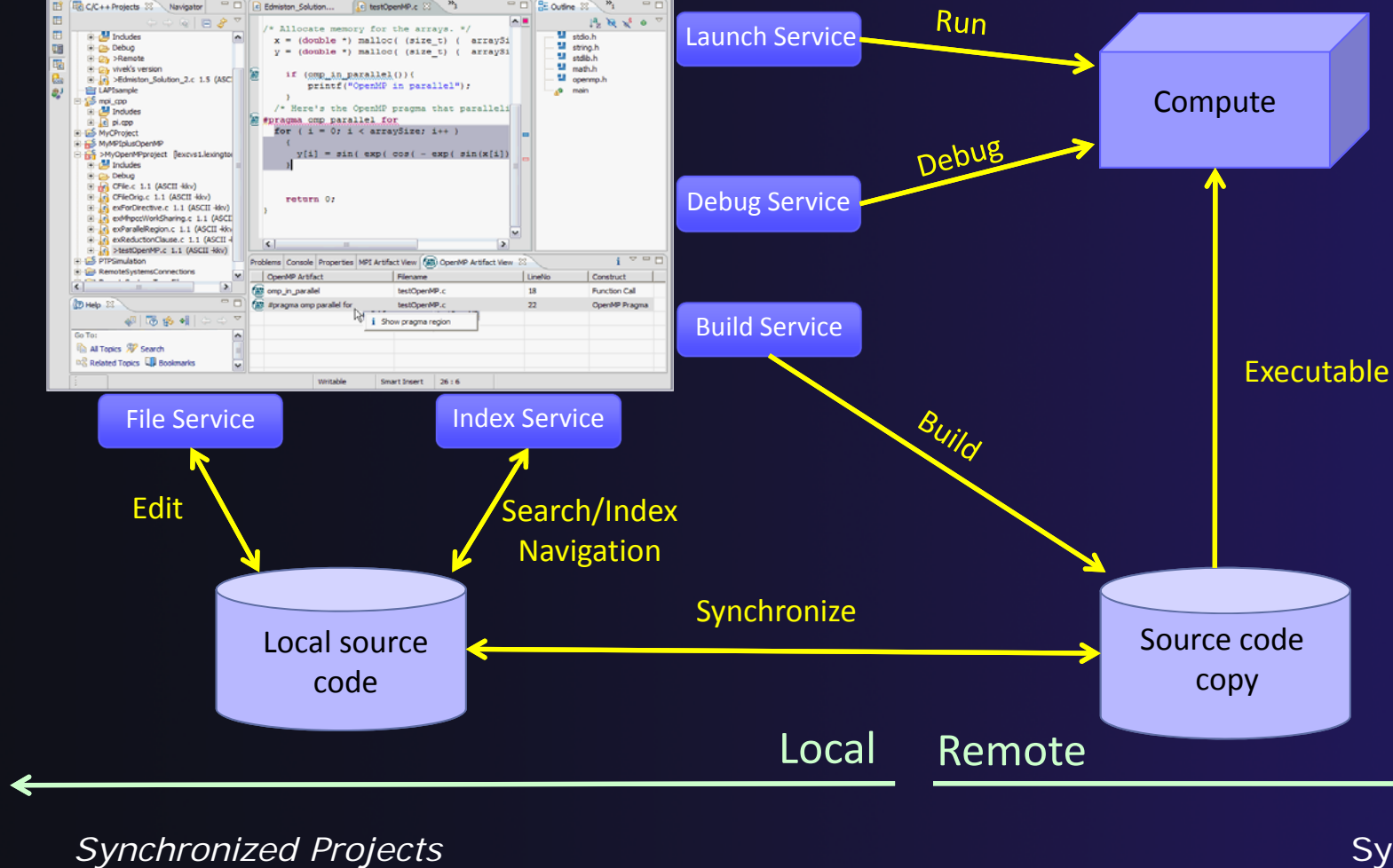
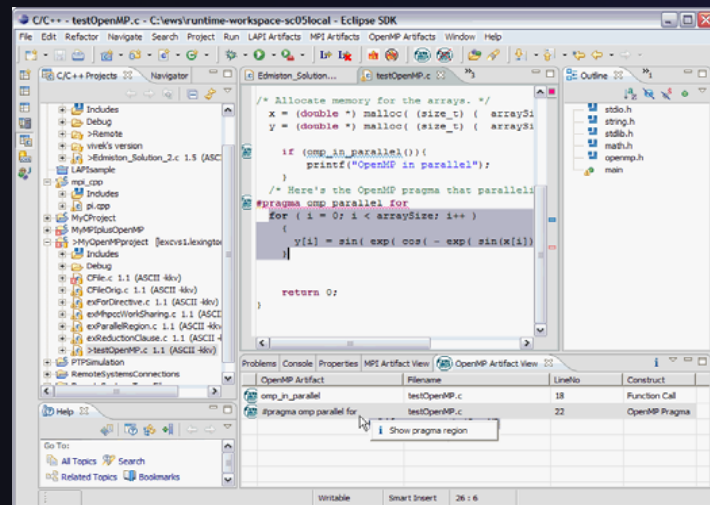
★ Console



*Running an Application*

Run-10

# Synchronized Projects

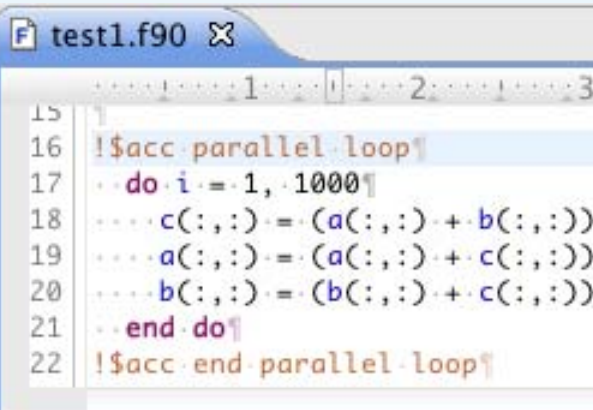




Integrated  
OpenACC  
documentation  
and PLDT  
support

(added for BW)

Documentation  
also available for  
MPI, OpenMP



The screenshot shows a code editor window titled 'test1.f90'. The code contains several OpenACC directives: `!$acc parallel loop`, `do i = 1, 1000`, `c(:, :) = (a(:, :) + b(:, :))`, `a(:, :) = (a(:, :) + c(:, :))`, `b(:, :) = (b(:, :) + c(:, :))`, `end do`, and `!$acc end parallel loop`. A code completion popup is visible, listing various OpenACC directives such as `!$acc cache`, `!$acc data`, `!$acc end data`, `!$acc declare`, `!$acc host_data`, `!$acc end host_data`, `!$acc kernels`, `!$acc end kernels`, `!$acc kernels loop`, `!$acc end kernels loop`, and `!$acc loop`.

**Code completion**

- `!$acc cache` – OpenACC cache directive
- `!$acc data` – OpenACC data directive
- `!$acc end data` – OpenACC end data directive
- `!$acc declare` – OpenACC declare directive
- `!$acc host_data` – OpenACC host\_data directive
- `!$acc end host_data` – OpenACC end host\_data directive
- `!$acc kernels` – OpenACC kernels directive
- `!$acc end kernels` – OpenACC end kernels directive
- `!$acc kernels loop` – OpenACC kernels loop directive
- `!$acc end kernels loop` – OpenACC end kernels loop directive
- `!$acc loop` – OpenACC loop directive

**OpenACC™ parallel directive**

Delineates a block of code that will be executed on an accelerator device.

Fortran Syntax	C Syntax
<code>!\$acc parallel [clause [, clause ...]]     block !\$acc end parallel</code>	<code>#pragma acc parallel [clause [, clause ...]]     block</code>

Supported clauses are `if`, `async`, `num_gangs`, `num_workers`, `vector_length`, `reduction`, `copy`, `copyin`, `copyout`, `create`, `present`, `present_or_copy`, `present_or_copyin`, `present_or_copyout`, `present_or_create`, `deviceptr`, `private`, `firstprivate`.

```

5  ..double precision :: a(SIZE, SIZE), b(SIZE, SIZE), c(SIZE, SIZE)
6
7  ..a(:) = 0..! This will raise a warning
8
9  ..do i = 1, SIZE
10 ..do j = 1, SIZE
11 ..a(i,j) = i*10.d0+j
12 ..b(i,j) = j*10.d0+i
13 ..end do
14 ..end do
15
16 !$acc-parallel-loop
17 ..do i = 1, 1000
18 ..c(:, :) = (a(:, :) + b(:, :)) / 2.d0
19 ..a(:, :) = (a(:, :) + c(:, :)) / 2.d0
20 ..b(:, :) = (b(:, :) + c(:, :)) / 2.d0
21 ..end do
22 !$acc-end-parallel-loop
23
24 ..!print *, "Averages:"
25 ..!print *, sum(a(:, :)) / (SIZE * SIZE)
26 ..!print *, sum(b(:, :)) / (SIZE * SIZE)
27 ..!print *, sum(c(:, :)) / (SIZE * SIZE)
28 ..print *, "Minimums:", minval(a(:, :)), minval(b(:, :)), minval(c(:, :))

```

After the build, compiler errors, warnings, and loopmark information are shown in the Problems view and source code editor

Problems	
1 error, 1 warning, 17 others	
Description	Resource
▼ ⚠ Warnings (1 item)	
⚠ The number of subscripts is smaller than the number of declared dimensions.	test1.f90
▼ ⓘ Infos (17 items)	
ⓘ A divide was turned into a multiply by a reciprocal	test1.f90
ⓘ A divide was turned into a multiply by a reciprocal	test1.f90
ⓘ A divide was turned into a multiply by a reciprocal	test1.f90
ⓘ A floating point expression involving an induction variable was strength reduced b...	test1.f90
ⓘ A loop nest at line 18 collapsed to a single loop.	test1.f90
ⓘ A loop starting at line 10 was not vectorized because a better candidate was found...	test1.f90
ⓘ A loop starting at line 17 was blocked with block size 512	test1.f90

(Cray, PGI support added for BW)



Resource Manager: ESS - Batch (XE)

Basic PBS Settings

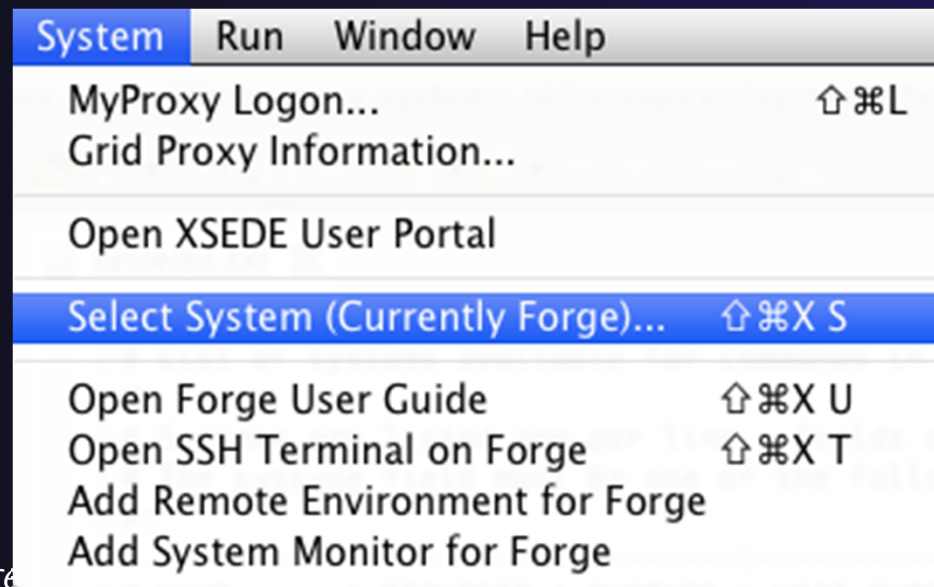
Import PBS Script

Name	Value	Description
Total MPI Tasks:	<input type="text" value="32"/>	Each XE6 node has two AMD Interlagos CPUs for a total of 32 integer cores and 16 floating point units per node. Therefore, the product of the number of MPI tasks per node and the number of OpenMP threads per task must be less than or equal to 32 (or 16 if running in single-stream mode). The number of MPI tasks per node must not exceed the total number of MPI tasks.
MPI Tasks per Node:	<input type="text" value="32"/>	
OpenMP Threads per Process:	<input type="text"/>	
Run in Dual-Stream Mode:	<input checked="" type="checkbox"/>	
Job Name:	<input type="text" value="ptp_job"/>	The name assigned to the job by the qsub or qalter command.
Account:	<input type="text"/>	Account to which to charge this job.
Queue:	<input type="text"/>	Designation of the queue to which to submit the job.
Total Memory Needed:	<input type="text"/>	Maximum amount of memory used by all concurrent processes in the job.

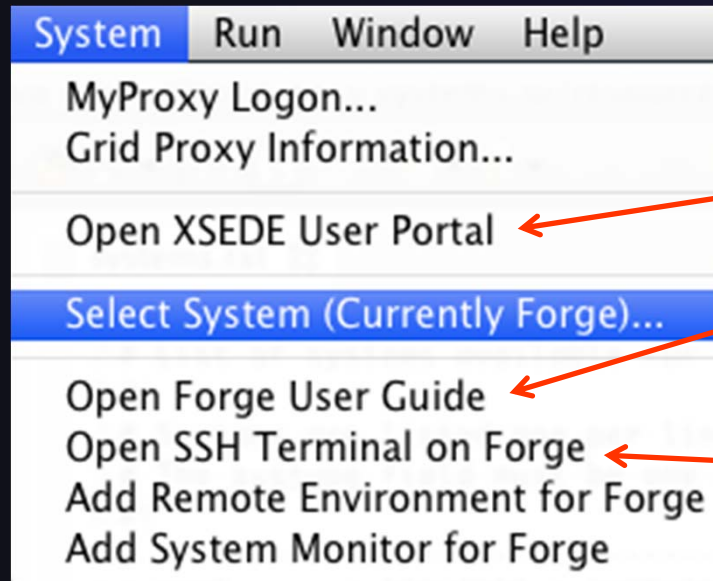
Graphical interface for launching a job (customized for BW)

# Additional Plug-ins from NCSA

- ★ NCSA publishes additional plug-ins can be added onto an existing PTP installation
  - ★ <http://forecaster.ncsa.uiuc.edu/help/index.jsp>
- ★ Contribute a **System** menu to the menu bar with XSEDE- and NCSA-specific commands



# System Menu



✦ Open Web content in Eclipse:

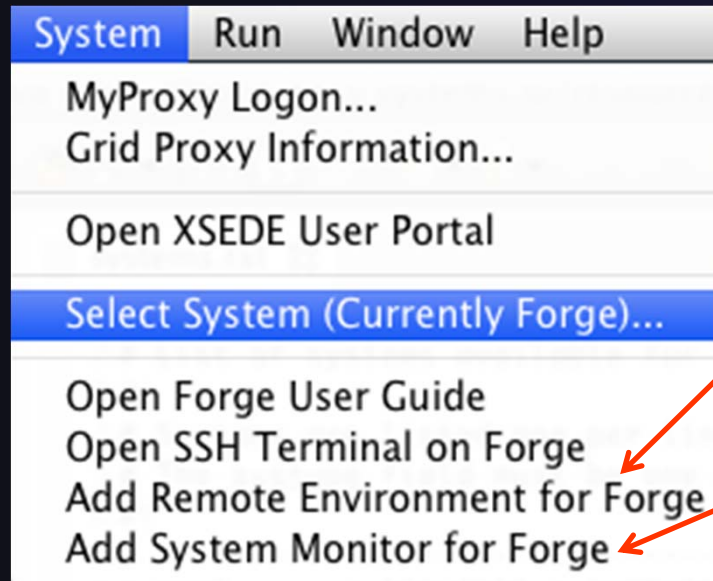
✦ **Open XSEDE User Portal**

✦ **Open User Guide** for a machine

✦ **Open an SSH terminal**  
(as an Eclipse view)

Eclipse-integrated SSH terminals are provided by the Remote System Explorer (RSE), one of the features that is included in the Eclipse for Parallel Application Developers package.

# System Menu

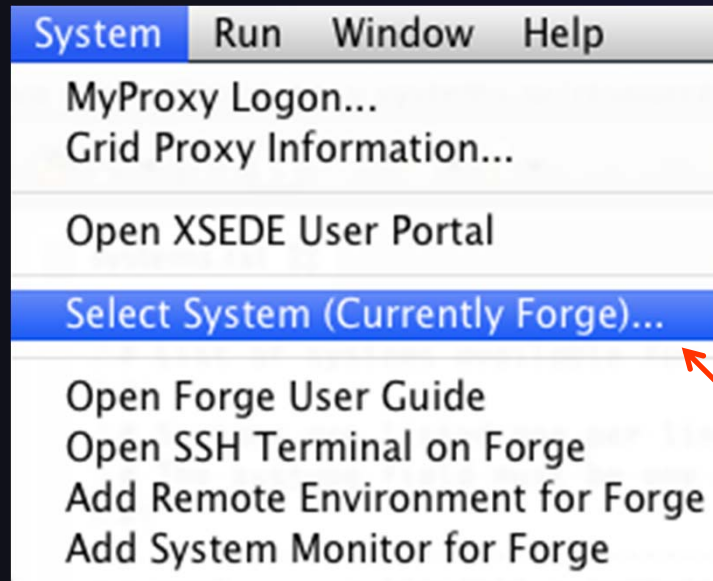


★ Shortcuts for common PTP tasks:

★ **Add Remote Environment** adds a Remote Tools connection for a particular machine

★ **Add System Monitor** opens the System Monitoring perspective and begins monitoring a particular machine

# System Menu



- ✦ The plug-in is preconfigured with information about XSEDE and NCSA resources
- ✦ The bottom four commands generally prompt for a system
- ✦ **Select System** can be used to eliminate this prompt, so these commands always act on a particular system



# MyProxy Logon



- ★ **MyProxy Logon** allows you to authenticate with a MyProxy server
  - ★ Often **myproxy.teragrid.org**
- ★ It stores a “credential,” which is usually valid for 12 hours
- ★ During these 12 hours, SSH connections to XSEDE resources will not require a password; they can use the stored credential
  - ★ However, you **must** enter the correct username for that machine!



## Improvements in the works...

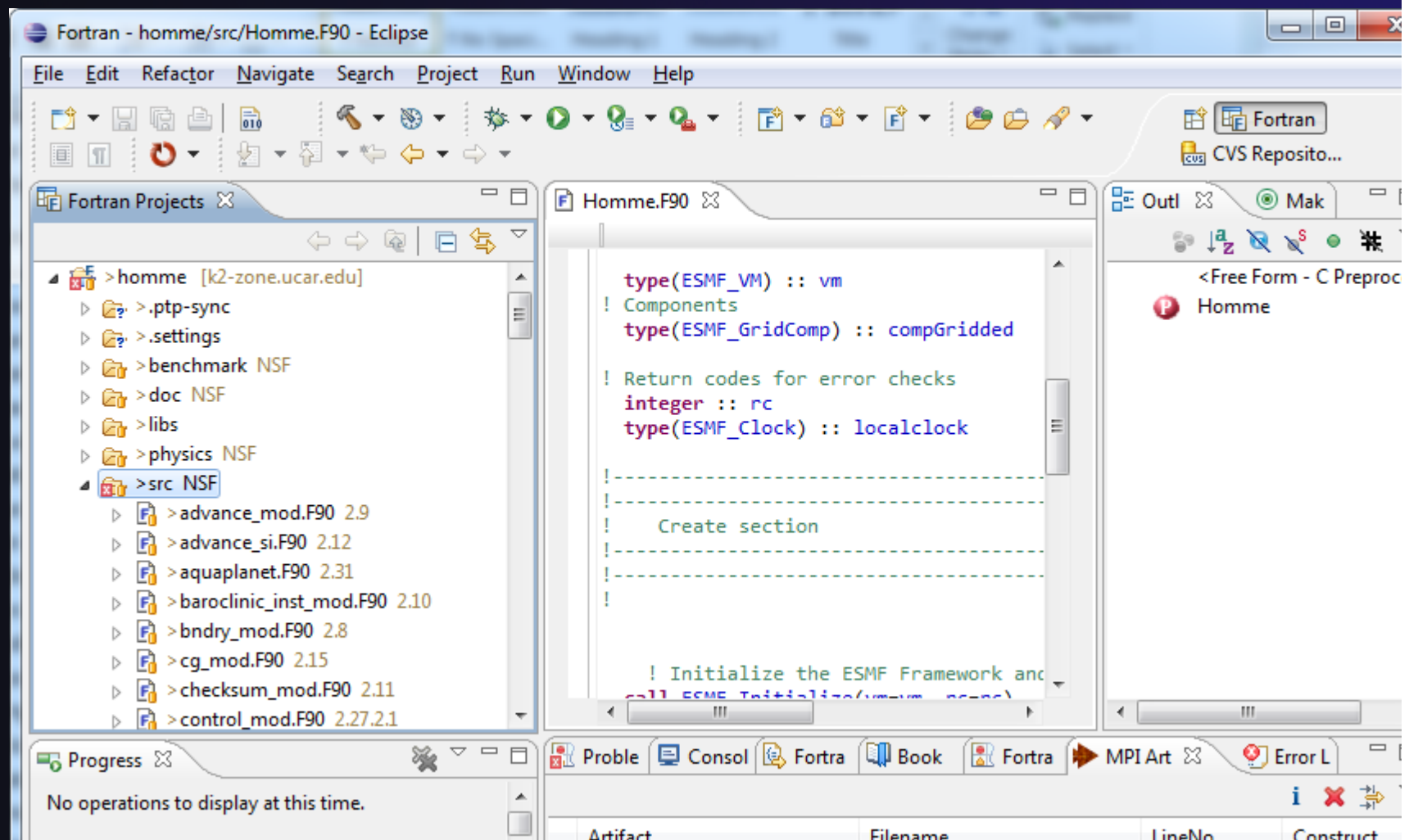
- ✦ External Tools Framework (ETFW) being transitioned to use the resource manager "JAXB" XML descriptions.
  - ✦ Enhanced flexibility for tool integration
- ✦ Environment Management enhancements
  - ✦ Ordering of modules (important for some machines)
  - ✦ Connect environment management and scanner-discovery
- ✦ Separate Build system from CDT build
  - ✦ Cleaner multi-system build management
  - ✦ Builds in any language

# Outline

- ★ Overview of Eclipse and Eclipse Parallel Tools Platform (PTP)
- ★ Motivation for Workbench for High Performance Computing (WHPC)
  - ★ Improvements to Eclipse PTP
- ★ Software Engineering Practices Enabled by Eclipse PTP
  - ★ Code visibility
  - ★ Multi-system build management
  - ★ Performance tuning
  - ★ Source code control
  - ★ Issue Tracking
  - ★ Documentation
  - ★ Earth Science/Weather code example
- ★ Eclipse PTP Resources

# Software Engineering

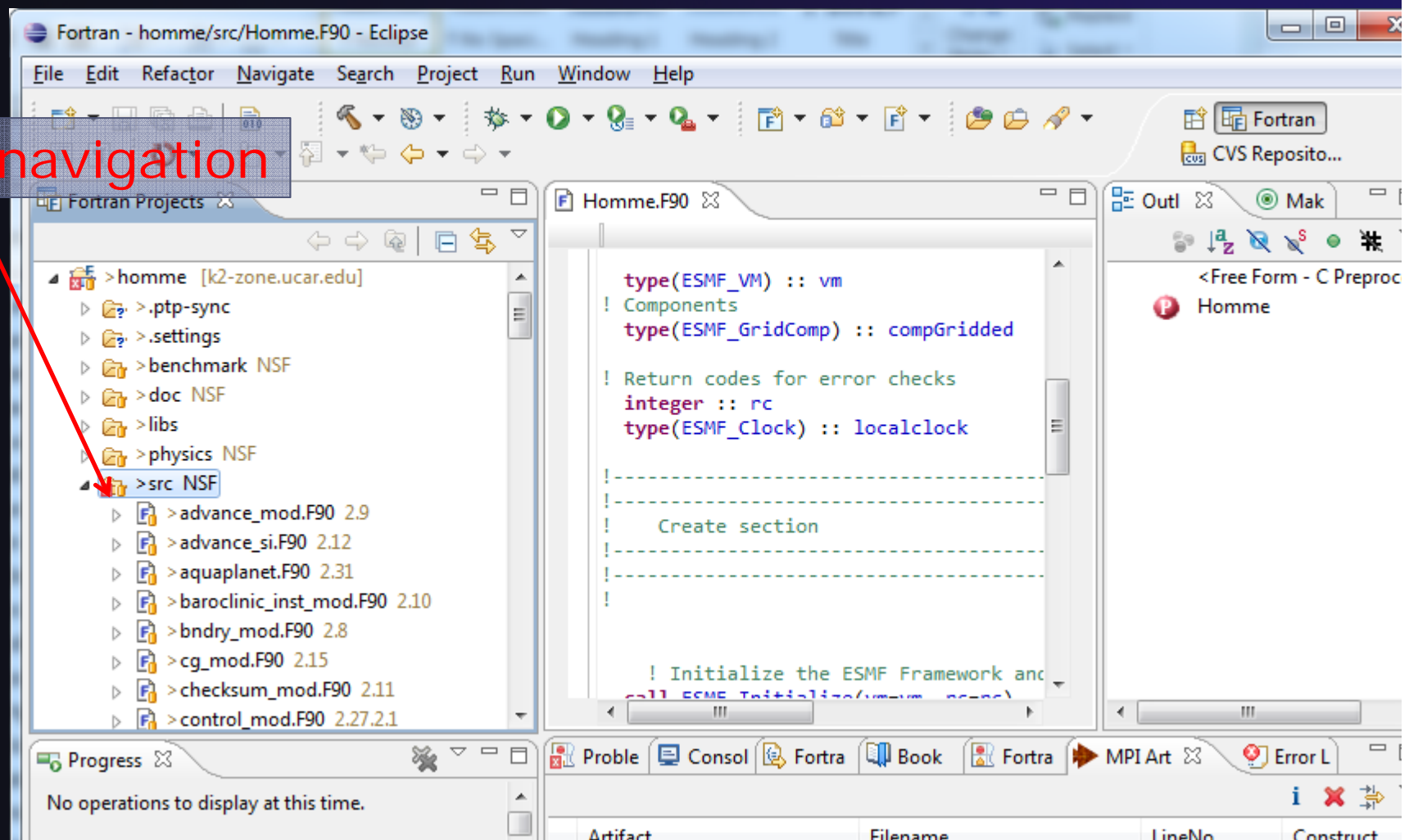
## ★ Code Visibility



# Software Engineering

## ★ Code Visibility

Code navigation



# Software Engineering

## ★ Code Visibility

The screenshot displays the Eclipse IDE interface for a Fortran project named 'homme'. The left-hand 'Project Explorer' pane shows a hierarchical view of the project structure, with the 'src' directory selected. A red arrow points from the 'src' directory to the 'Homme.F90' file in the main editor. The main editor window shows the source code of 'Homme.F90', which includes several Fortran declarations and comments. A red arrow points from the 'Syntax-aware editing' text box to the 'Create section' comment in the code. The right-hand 'Outline' pane shows the project structure, and the bottom status bar indicates 'No operations to display at this time'.

Code navigation

Syntax-aware editing  
(navigate to program  
units and declarations)

```
type(ESMF_VM) :: vm
! Components
type(ESMF_GridComp) :: compGridded

! Return codes for error checks
integer :: rc
type(ESMF_Clock) :: localclock

!
! Create section
!
```



# Software Engineering

## ★ Code Visibility

The screenshot displays the Eclipse IDE interface for a Fortran project. The left-hand 'Project Explorer' shows a tree structure of the project 'homme' with subdirectories like '.ptp-sync', '.settings', 'benchmark', 'doc', 'libs', 'physics', and 'src'. The 'src' directory is expanded, showing various Fortran source files. A red arrow points from the 'Code navigation' label to the 'src' directory. The central 'Editor' window displays the source code of 'Homme.F90', which includes Fortran declarations and comments. A red arrow points from the 'Code Outline' label to the 'Outline' view on the right, which shows a hierarchical view of the code structure. A red arrow points from the 'Syntax-aware editing' label to the code in the editor. The code in the editor includes declarations for 'vm', 'compGridded', 'rc', and 'localclock', along with comments and a section separator.

Code Outline

Code navigation

Syntax-aware editing  
(navigate to program  
units and declarations)

```
type(ESMF_VM) :: vm
! Components
type(ESMF_GridComp) :: compGridded

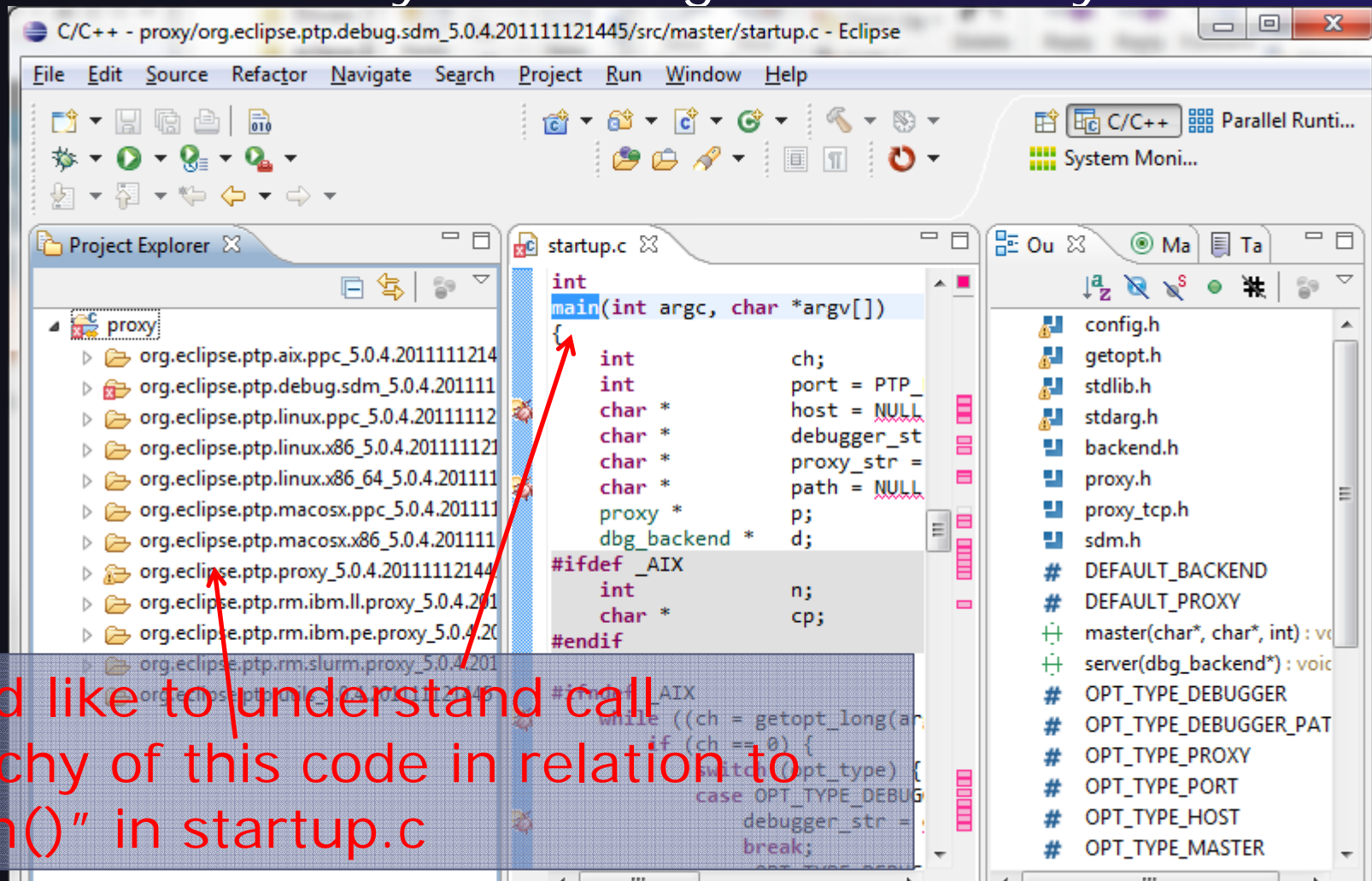
! Return codes for error checks
integer :: rc
type(ESMF_Clock) :: localclock

-----
! Create section
!-----
```



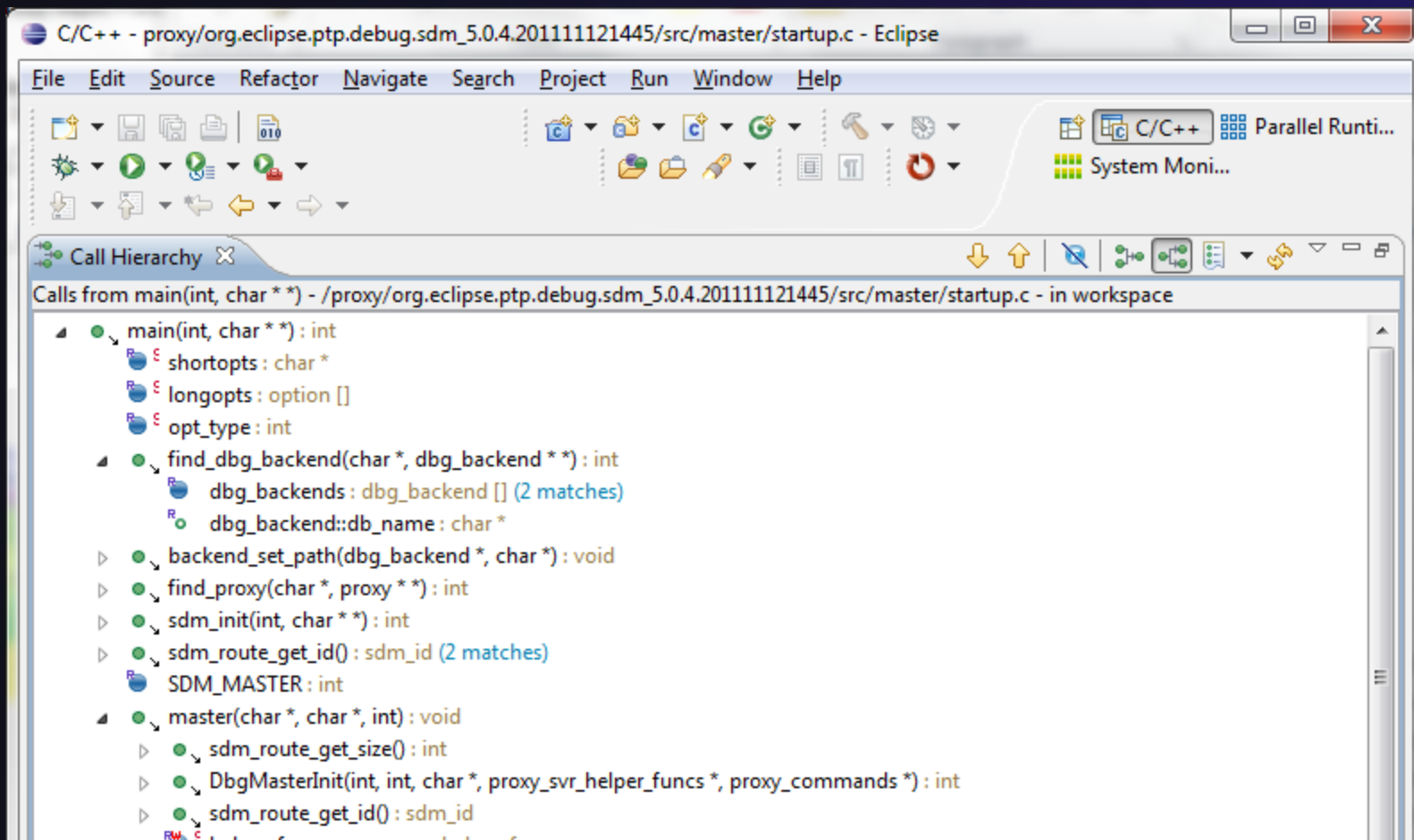
# Software Engineering

## ★ Code visibility: deducing call hierarchy



# Software Engineering: Call Hierarchy (C/C++)

- ★ After selecting main, right click and select <Open Call Hierarchy>



# Multi-machine build management

## ★ Local

- ★ Source is located on local machine, builds happen locally

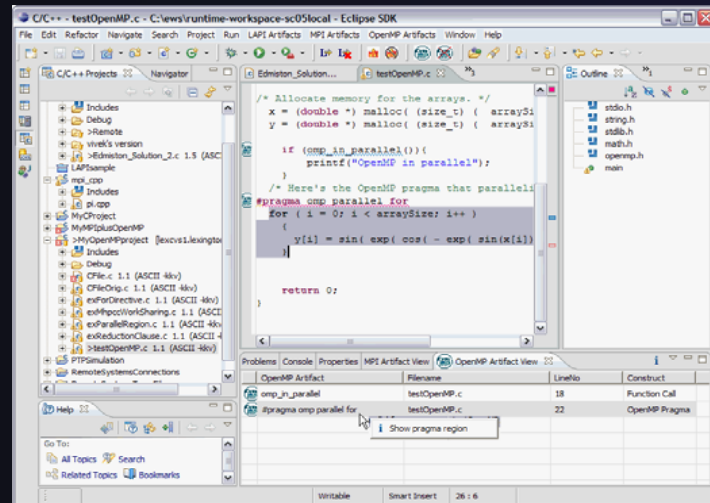
## ★ Synchronized

- ★ Source is local, then synchronized with remote machine(s)
- ★ Building and launching happens remotely (can also happen locally)

## ★ Remote

- ★ Source is located on remote machine(s), build and launch takes place on remote machine(s)

# Synchronized Projects



File Service

Index Service

Edit

Search/Index  
NavigationLocal source  
code

Synchronize

Source code  
copy

Local

Remote

Launch Service

Run

Debug Service

Debug

Build Service

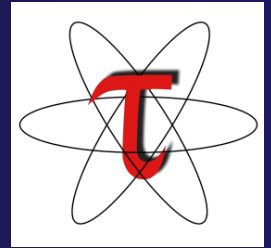
Build

Compute

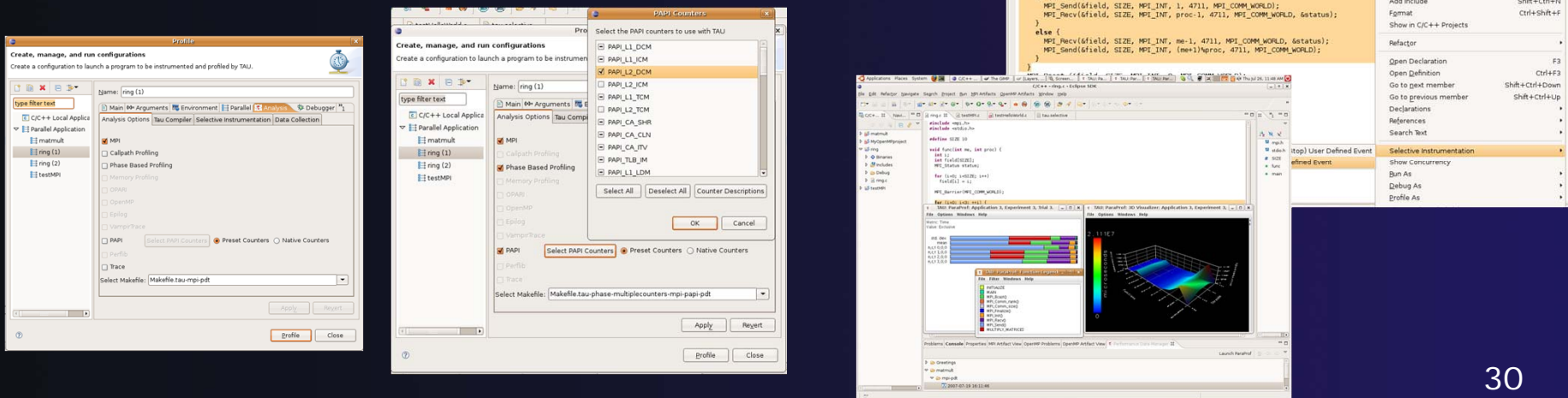
Executable

# Performance Tuning: PTP TAU plug-ins

<http://www.cs.uoregon.edu/research/tau>



- ✦ TAU (Tuning and Analysis Utilities)
- ✦ First implementation of External Tools Framework (ETFw)
  - ✦ Support for additional command-line tools can be added with XML tool definitions
- ✦ Eclipse plug-ins wrap TAU functions, make them available from Eclipse
- ✦ Full GUI support for the TAU command line interface
- ✦ Performance analysis integrated with development environment





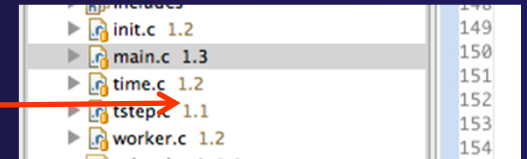
# Source Code Control: “Team” Features

- ✦ Eclipse supports integration with multiple version control systems (VCS)
  - ✦ CVS, SVN, Git, and others
  - ✦ Collectively known as “Team” services
- ✦ Many features are common across VCS
  - ✦ Compare/merge
  - ✦ History
  - ✦ Check-in/check-out
- ✦ Some differences
  - ✦ Version numbers
  - ✦ Branching

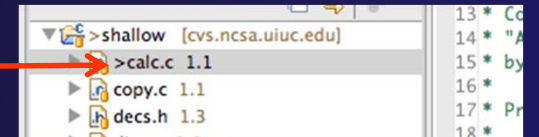


# CVS Features

★ Shows version numbers next to each resource

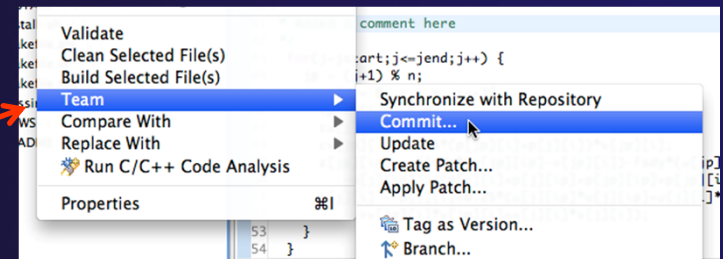


★ Marks resources that have changed

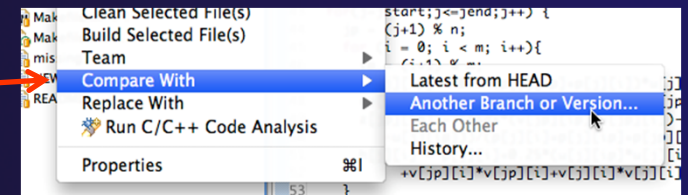


★ Can also change color (preference option)

★ Context menu for Team operations



★ Compare to latest, another branch, or history



★ Synchronize whole project (or any selected resources)

★ Similar support for SVN, Git, ...

# Issue Tracking

## ✦ Mylyn Bridge

- ✦ Tracks tasks, links to source and bug repositories

nit jobs on jyc ( mpp\* arguments are not supported on this sytem ... ) - Eclipse

or Navigate Search Project System Run Window Help

Quick Access C/C++ CVS Repository Exploring System Monitoring Parallel Debug Remote System Explorer

Issue BWDSPCH-515 NCSA Submit

ddt doesn't submit jobs on jyc ( mpp\* arguments are not supported on ... )

Status: In Progress Created: Oct 3, 2012 Modified: Oct 3, 2012 11:38 AM  
Reported by: Galen Arnold Assigned to: Unassigned

Attributes

Project: BW Dispatch Priority: Normal

Type: Ticket Due Date: 0m

Affects Versions: Components: Security Level: Private Ticket

Fix Versions: PI: Kramer

First Responder: gbauer

Environment:

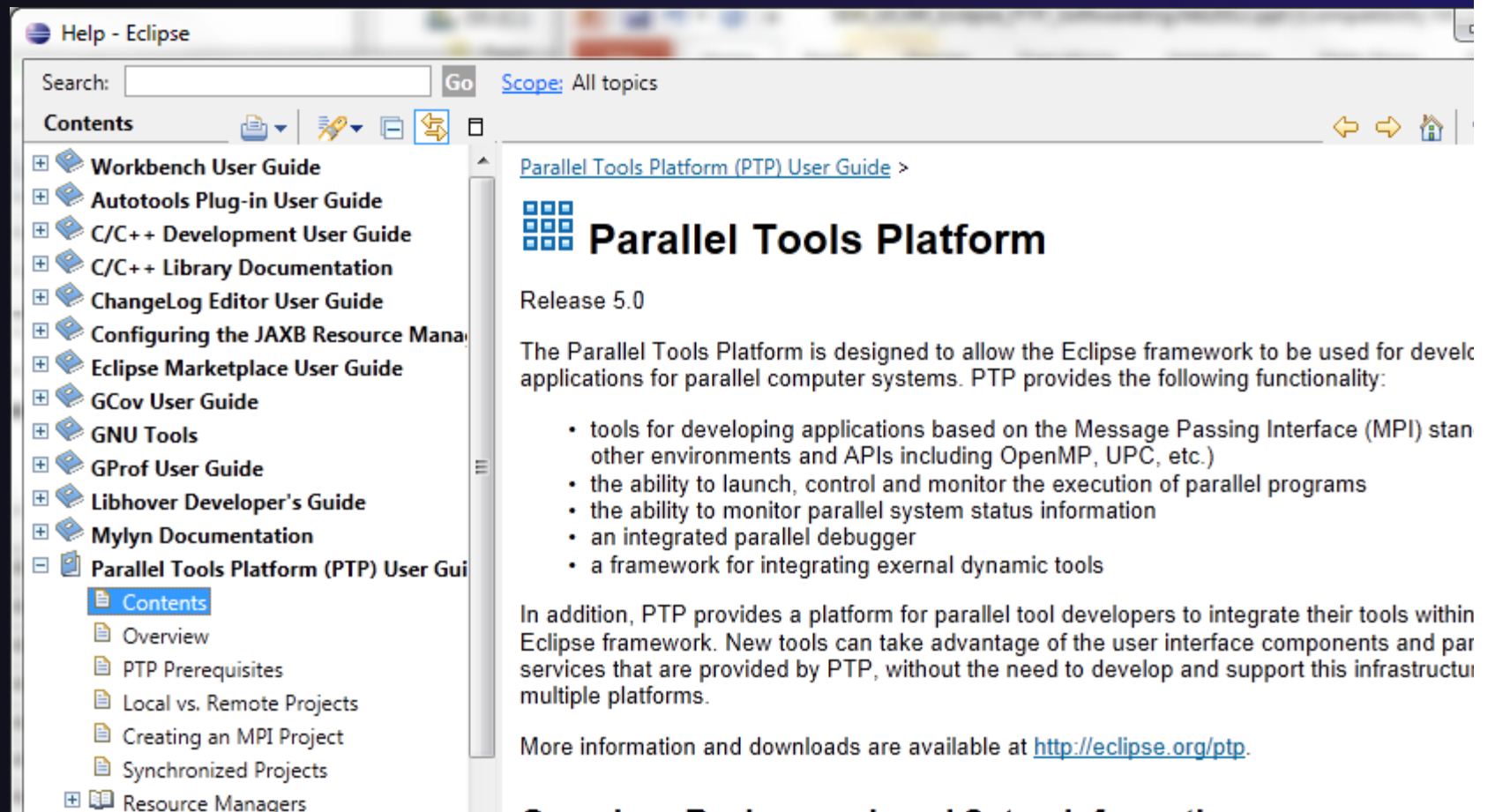
Find All Activa...

Unsubmitted [NCSA]  
Added recently (BW Disp  
Added recently (whpc)  
Assigned to me (All Proj  
BW's unassigned issues  
BWDSPCH-481: sett  
BWAM-525: Re: Blu  
BWDSPCH-502: BW  
BWDSPCH-515: ddt  
Open BW tickets [NCSA  
PTP 6.0 Bugs [Eclipse.or  
PTP version 7.0 [Eclipse  
whpc-anybug [NCSA  
Unmatched [NCSA]

Connections to Jira, bugzilla, ...

# Eclipse Documentation

- ★ Eclipse Help System – built in and standalone (<http://help.eclipse.org>)



# Adapting Eclipse Documentation to Other Projects: QMCPack

★ See <http://code.google.com/p/qmcpack-doc/>

The screenshot shows a web browser window with the address bar displaying <http://code.google.com/p/qmcpack-doc/>. The page title is "qmcpack-doc" and the subtitle is "QMCPACK document". The navigation bar includes links for "Project Home", "Downloads", "Wiki", "Issues", and "Source". Below the navigation bar, there are tabs for "Summary", "Updates", and "People". The "Summary" tab is selected, showing "Project Information" on the left and "Developers' and users' guides" on the right. The "Project Information" section includes links for "Activity" (Low), "Project feeds", "Code license", and "New BSD License". The "Members" section lists several contributors and their email addresses. The "Developers' and users' guides" section describes the development of the Eclipse plug-in and provides a list of build instructions, doxygen code documentation, and other materials on the wiki. A red arrow points to the text "Instructions for viewing help page in eclipse" at the bottom of the page.

Project Home Downloads Wiki Issues Source

Summary Updates People

**Project Information**

[Activity](#) Low  
[Project feeds](#)

**Code license**  
[New BSD License](#)

**Members**  
[jeongnim.kim](#),  
[david.ce...@gmail.com](#),  
[kpes...@gmail.com](#),  
[jmcminis](#), [bkcl...@gmail.com](#),  
[jaronkro...@gmail.com](#),  
[miguel.m...@gmail.com](#),  
[lshulenb...@gmail.com](#)  
5 committers

**Developers' and users' guides**

org.cmscc.qmcpack.doc is developed as an eclipse plug-in for QMCPACK help page. If all goes well, a help document with

- build instructions
- doxygen code documentation
- other materials on wiki

can be downloaded as an eclipse plug-in.

Licensed under [UIUC/NCSA open-source license](#)

See more on [UIUC/NCSA license](#)

**Instructions for viewing help page in eclipse**

## Consider 2 possible types of users of Eclipse Parallel Tools ...

- ★ Weather code users/modelers
  - ★ Need to build weather code
  - ★ May need to modify weather code (and rebuild)
- ★ Software specialists enabling modeling projects
  - ★ Lots of software engineering concerns
- ★ Next set of slides address some of those concerns.



# Weather code users/modelers

## ★ Some of the challenges

- ★ Complex codes (eg WRF)
- ★ Codes + HPC architectures can be daunting
- ★ Adding user code not always easy

## ★ WRF: from [http://wrf-model.org/PRESENTATIONS/2000\\_04\\_18\\_Klemp/sld007.htm](http://wrf-model.org/PRESENTATIONS/2000_04_18_Klemp/sld007.htm)

### WRF Hierarchical Software Architecture

#### ■ Top-level “Driver” layer

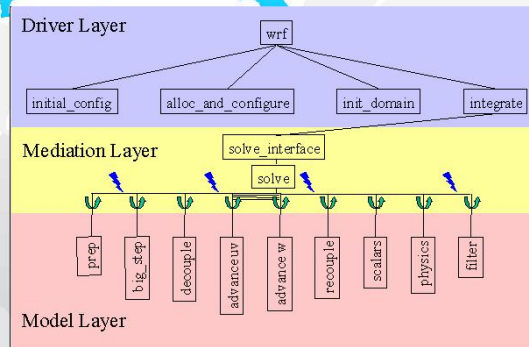
- Isolates computer architecture concerns
- Manages execution over multiple nested domains
- Provides top level control over parallelism
  - » patch-decomposition
  - » inter-processor communication
  - » shared-memory parallelism
- Controls Input/Output

#### ■ “Mediation” Layer

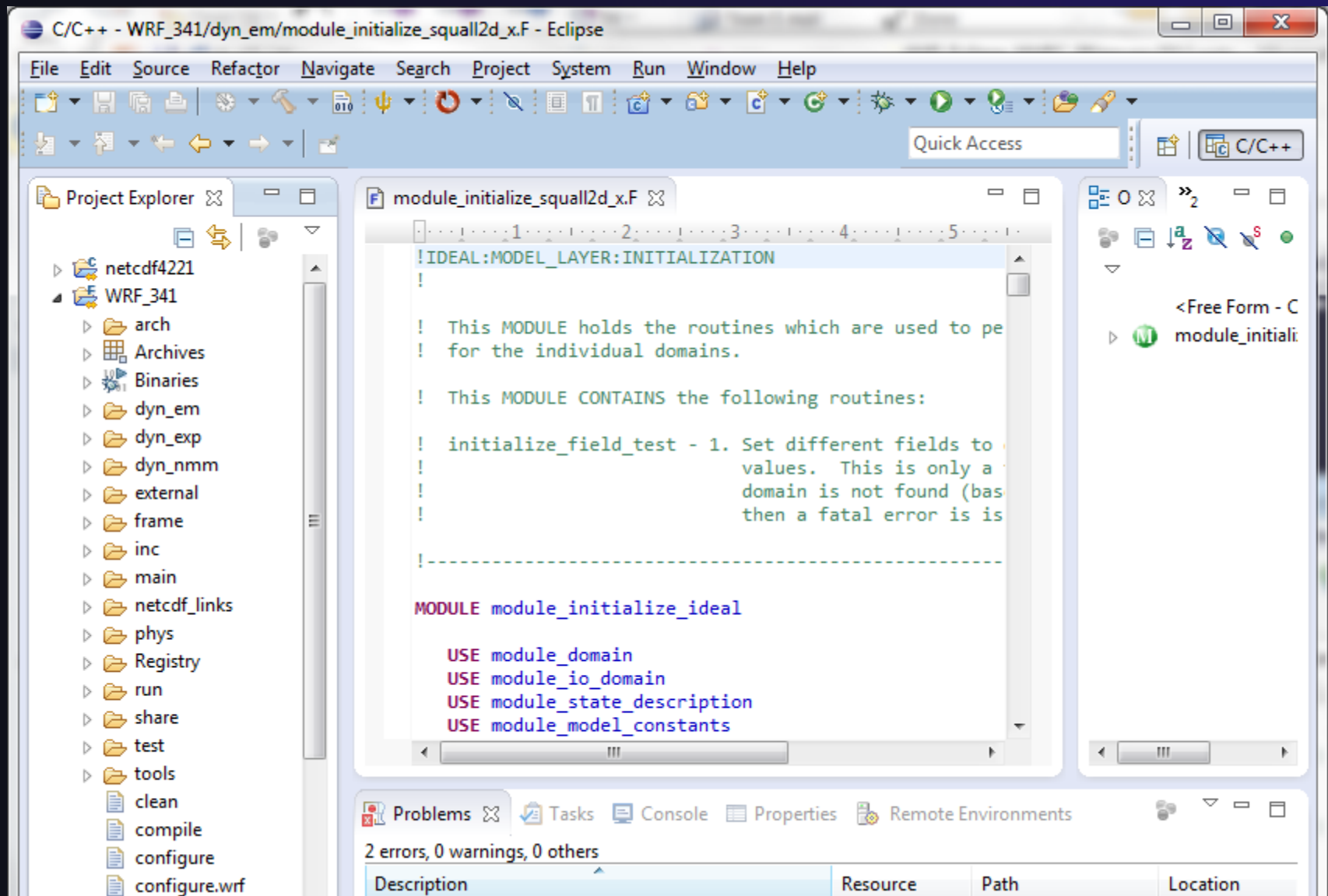
- Specific calls to parallel mechanisms

#### ■ Low-Level “Model” layer

- Performs actual model computations
- Tile-callable
- Scientists insulated from parallelism
- General, fully reusable

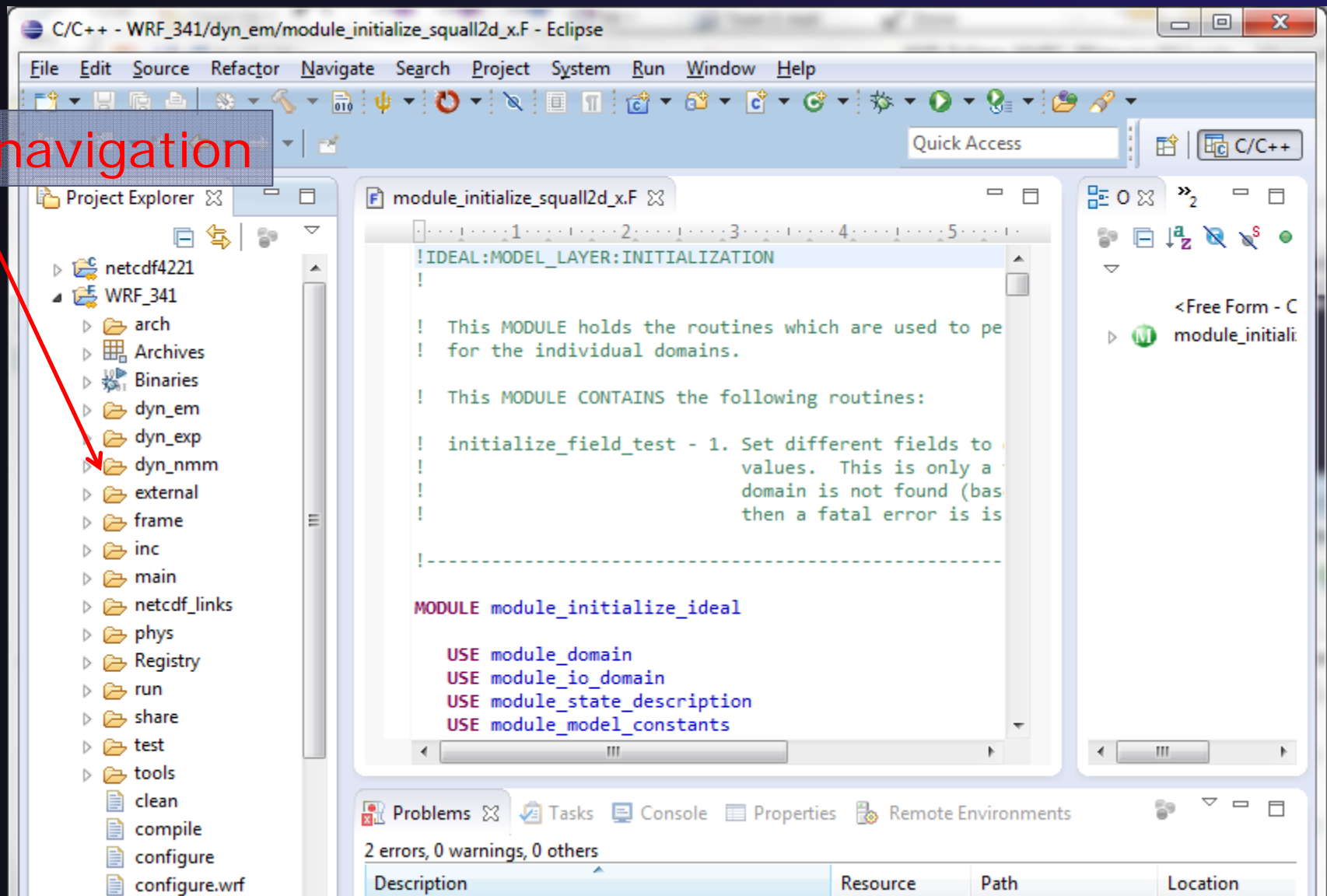


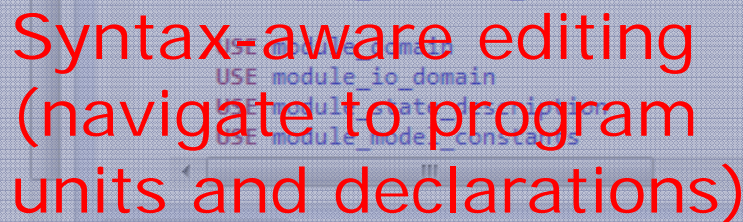
# Navigating Weather Codes



# Navigating Weather Codes

Code navigation





# Navigating Weather Codes

The screenshot displays the Eclipse IDE interface for a C/C++ project named 'WRF\_341'. The Project Explorer on the left shows a hierarchical view of the project files, including folders like 'arch', 'Archives', 'Binaries', 'dyn\_em', 'dyn\_exp', 'dyn\_nmm', 'external', 'frame', 'inc', 'main', 'netcdf\_links', 'phys', 'Registry', 'run', 'share', 'test', 'tools', and files like 'clean', 'compile', 'configure', and 'configure.wrf'. A red arrow points from the 'Code navigation' label to the Project Explorer.

The main editor window shows the source code for 'module\_initialize\_squall2d\_x.F'. The code includes comments and Fortran-style declarations. A red arrow points from the 'Code Outline' label to the Outline view on the right, which shows the structure of the code, including the 'module\_initialize' unit. Another red arrow points from the 'Syntax-aware editing (navigate to program units and declarations)' label to the code editor, specifically pointing to the 'MODULE module\_initialize\_ideal' declaration.

**Code navigation**

**Code Outline**

**Syntax-aware editing (navigate to program units and declarations)**



## Eclipse aiding in the WRF workflow...

- ✦ May want to add a model output variable
- ✦ Eclipse PTP makes it easy to navigate source, make changes
- ✦ WRF Build
  - ✦ Interactive “compile” script – use terminal within Eclipse to complete
  - ✦ Configure Eclipse to drive “make” to iteratively build after modifications
- ✦ WRF Run – can generate a run configuration for particular system, batch environment

# Software Specialists enabling modeling projects

- ★ Need a wider array of software engineering tools
  - ★ Source repository
  - ★ Issue tracking
  - ★ Documentation
  - ★ Performance tuning...
- ★ Eclipse Parallel Tools can help with many of these concerns

# Outline

- ★ Overview of Eclipse and Eclipse Parallel Tools Platform (PTP)
- ★ Motivation for Workbench for High Performance Computing (WHPC)
  - ★ Improvements to Eclipse PTP
- ★ Software Engineering Practices Enabled by Eclipse PTP
  - ★ Code visibility
  - ★ Multi-system build management
  - ★ Performance tuning
  - ★ Source code control
  - ★ Issue Tracking
  - ★ Documentation
  - ★ Earth Science/Weather code example
- ★ Eclipse PTP Resources

# Online Information

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

# Mailing Lists

## ★ 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 - high volume
  - ★ <http://dev.eclipse.org/mailman/listinfo/ptp-dev>

## ★ Photran Mailing lists

- ★ User discussion and queries
  - ★ <http://dev.eclipse.org/mailman/listinfo/photran>
- ★ Developer discussions –
  - ★ <http://dev.eclipse.org/mailman/listinfo/photran-dev>



# Getting Involved

- ★ See <http://eclipse.org/ptp>
- ★ Read the developer documentation on the wiki
- ★ Join the mailing lists
- ★ Attend the monthly developer meetings
  - ★ Conf Call Monthly: Second Tuesday, 1:00 pm ET
  - ★ Details on the PTP wiki
- ★ Attend the monthly user meetings
  - ★ Teleconference Monthly
  - ★ Each 4<sup>th</sup> Wednesday, 2:00 pm ET
  - ★ Details on the PTP wiki

**PTP will only succeed with your participation!**