IBM Research

An Integrated Tools Platform for Multi-Core Enablement

Beth Tibbitts
tibbitts@us.ibm.com

Evelyn Duesterwald
duester@us.ibm.com

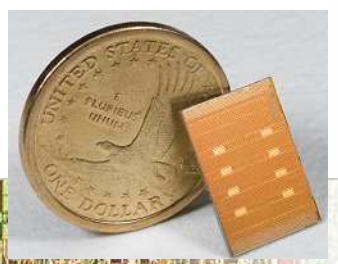
IBM T. J. Watson Research Center

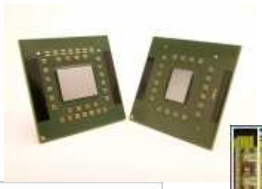
"This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under its Agreement No. HR0011-07-9-0002"

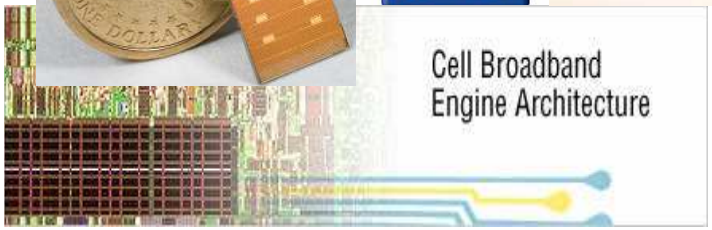
March 2007

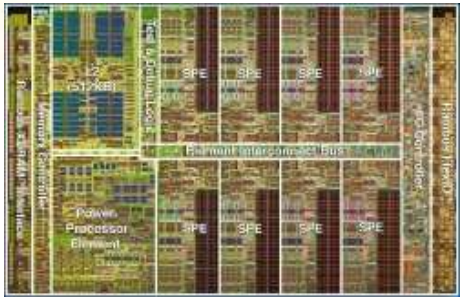
The Challenge of Multi-Core Programming

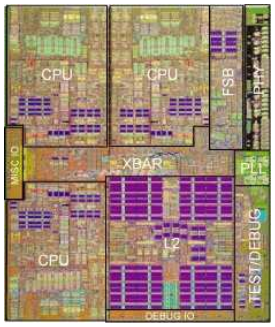
- The growing number and scale of available multi-core systems changes the landscape of parallel computing
- **The richness of commercial IDEs needs to be available to the HPC programmer**
  - Growing a Parallel ecosystem around a common IDE benefits all
  - A common tools base can address the needs of a broad spectrum of HPC users ranging from novice to expert parallel programmers
- Tools to make this domain easy to use and productive are crucial for developers!
- A holistic approach to tools for enhancing productivity provides a full range of features.
- An open and extensible platform encourages further development of tools by IBM and others











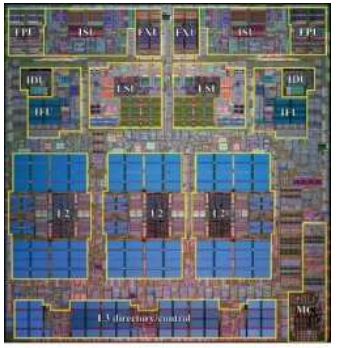


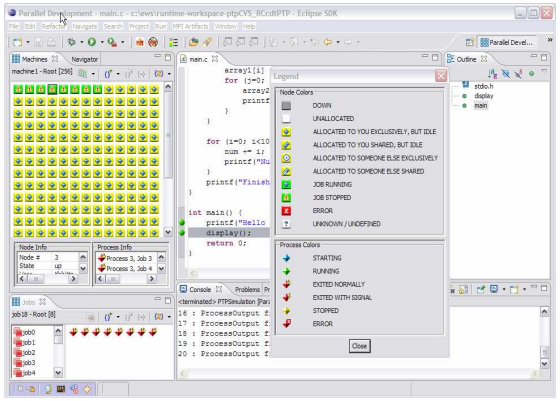
Figure 2

POWER5 die photo showing major chip components. (FXU: fixed-point execution unit; ISU: instruction sequencing unit; IDU: instruction decode unit; LSU: load/store unit; IFU: instruction fetch unit; FPU: floating-point unit; MC: memory controller.) Reprinted with permission from [2].

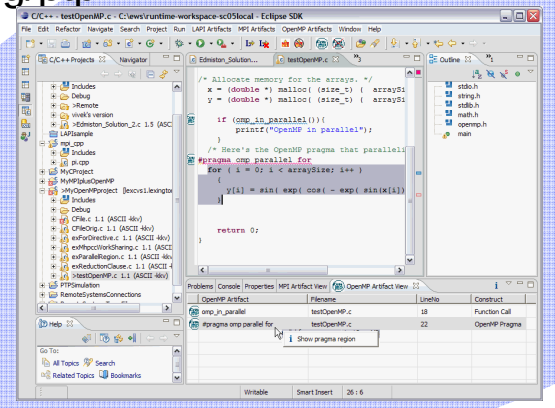
Eclipse PTP: Parallel Tools Platform

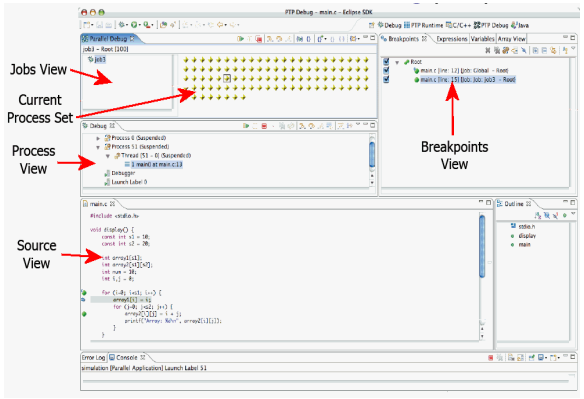
http://eclipse.org/ptp

Parallel Runtime



Parallel Language Dev. Tools (PLDT)

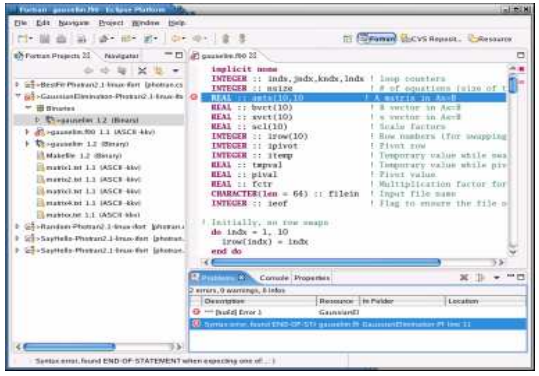




Parallel Debugger

Based on
C/C++ Development Tools: CDT
http://eclipse.org/cdt

Fortran Tools



Performance Tools*

Based on TuningFork:
http://www.alphaworks.ibm.com/tech/tuningfork

IBM Contributions

* Not yet publicly available on eclipse.org

Eclipse

March 5th - 8th
Santa Clara, California

- Eclipse is an open source IDE (Integrated Development Environment) and a generalized tools platform
- EclipseCon 2007 just completed in Santa Clara, 1200+ participants
- Eclipse "Ecosystem" and Open Source Community are a model for large community of developers.
  - Originally Eclipse supported only Java.
  - Now it supports many languages, the C/C++ Development Tools (CDT) being the largest, and the "poster child" of Eclipse projects*
  - The PTP (Parallel Tools Platform) layers specific tools for Parallel Programmers on top of CDT and Photran (Fortran Tools)
- Eclipse releases are the largest one-day release of open source code
  - Simultaneous release of 10+ software projects, yearly (June)

* Mike Milinkovich, Eclipse Foundation, March 2007 at EclipseCon 2007.

Parallel Language Development Tools (PLDT)

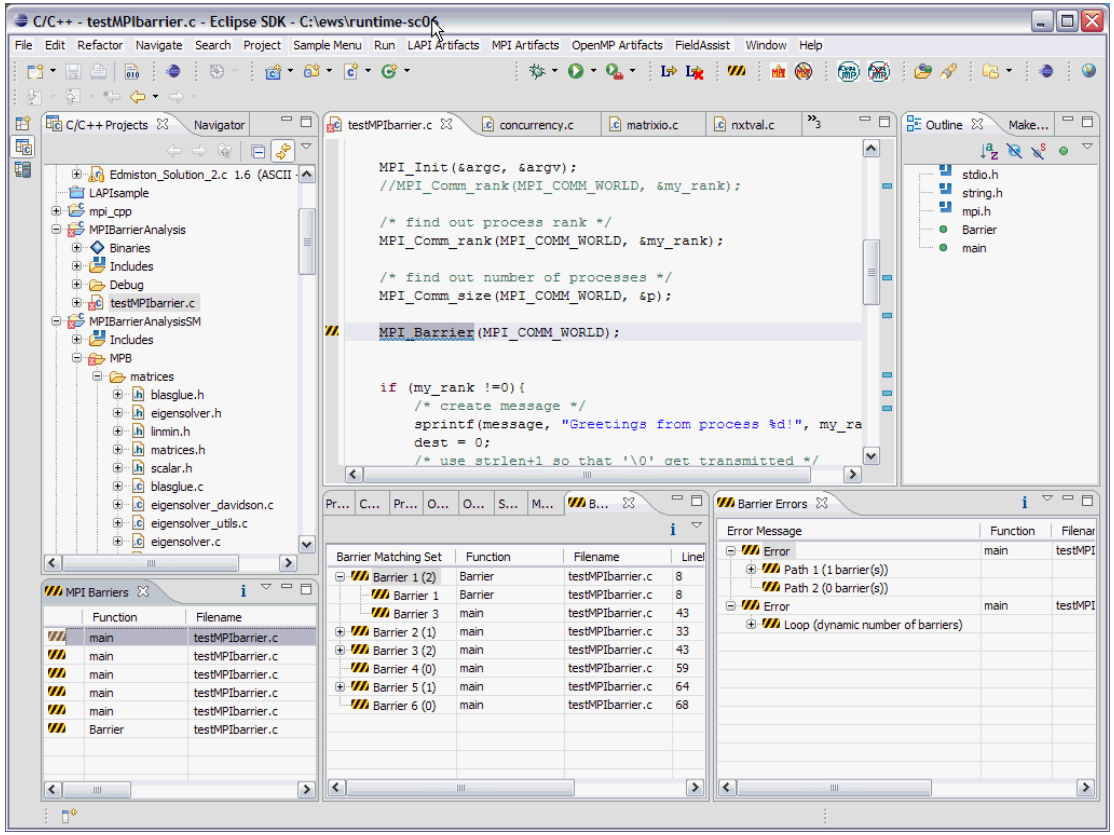
- **Part of the Eclipse Parallel Tools Platform** http://eclipse.org/ptp
- **Cover a spectrum of tools to enhance programmer productivity with OpenMP**
  - Assistance tools
    - Editing services: content assist, automatic completion, help views
    - Project wizards to create parallel projects
  - Syntax analysis tools
    - Find parallel language artifacts
    - Check for common usage errors (e.g., improper nesting of parallel constructs)
  - Sophisticated static semantic analysis tools
    - Concurrency analysis to detect parallel statements
- **Fully integrated through a common base (CDT for C/C++ languages)**
  - **Static analysis features of CDT Utilized: AST; graphs constructed**
- **Fully integrated with tools for distributed memory (MPI) programming models**
  - Complex programs can combine shared and distributed memory programming techniques

Demo (screenshots available at end of slides)

- Eclipse and CDT brief tour
  - Workbench, Projects, Editor, Outline, debugger
- PTP/PLDT for OpenMP:
  - Content Assist, Hover help, F1 Help View
  - OpenMP Artifacts located: source code navigation
  - #pragma regions highlighted
  - OpenMP problems view: “Usage checker”
  - Concurrency Analysis
  - Run OpenMP program (printf output)
  - Debugger: dependent on underlying multi-threaded debugger (gdb) PROBABLY NOT TIME
- Other things to mention:
- PTP’s MPI runtime/debugger
- MPI Barrier Analysis (PPOPP paper/demo)
- Cell IDE

Some related Eclipse and PTP work…

PTP PLDT: MPI Barrier Analysis



Verify barrier synchronization in C/MPI programs

- Interprocedural static analysis.

- Output:
  1) For verified programs, lists barrier statements that synchronization together (match)

  2) For synchronization errors, reports counter example that illustrates and explains the error.

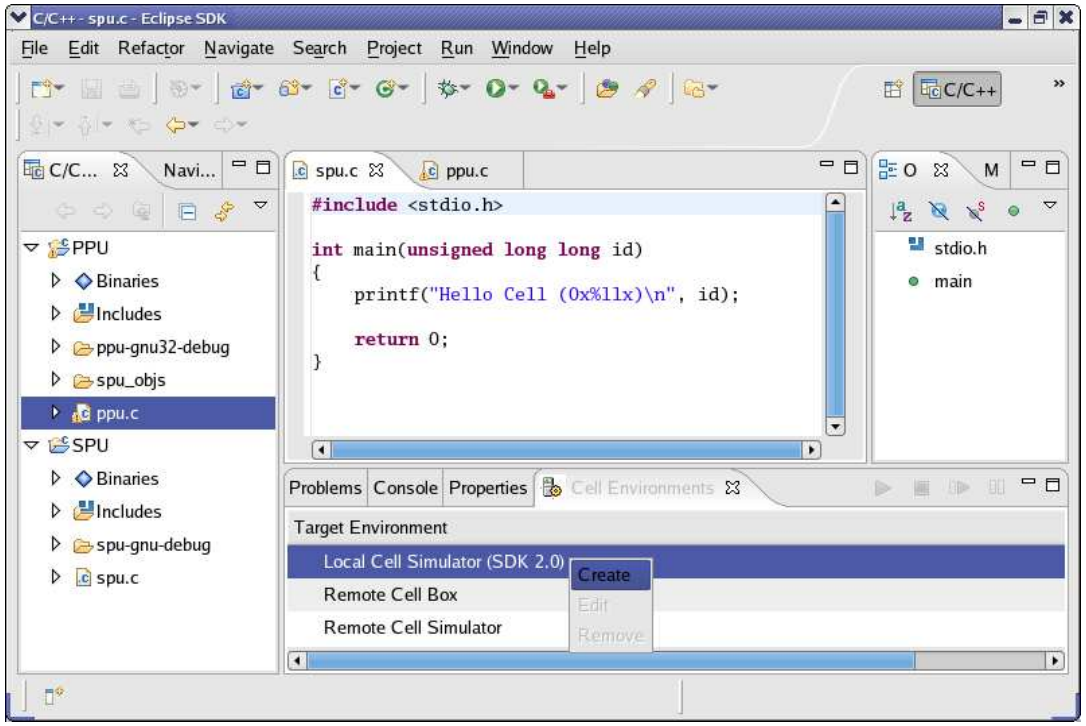
See PPOPP paper: Yuan Zhang and Evelyn Duesterwald. “Barrier matching for programs with textually unaligned barriers.” In Proceedings of the Symposium on **Principles and Practice of Parallel Programming**, March 2007.

Eclipse IBM Cell IDE

http://www.ibm.com/developerworks/power/cell

* IBM Cell Broadband Engine is a heterogeneous multi-core chip with one PPE (Power Processing Element) and 8 SPEs (Synergistic Processing Elements)

- Available on IBM alphaworks in Cell SDK 2.0
- "Managed Build" supported for Cell SPE,PPE processors
- Seamless integration of Cell BE Simulator
- Debugger for both PPE,SPE processors



IBM Research's single source compiler: (utilizes OpenMP directives)
http://www.research.ibm.com/cellcompiler/

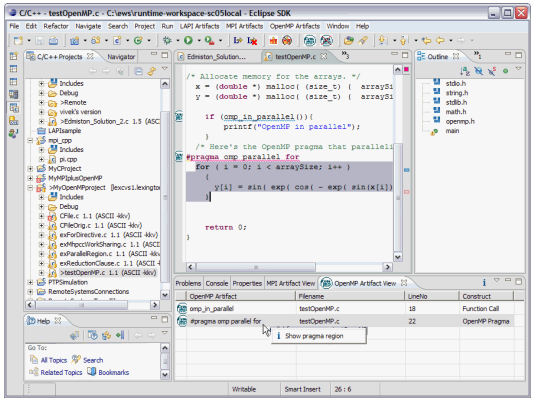
Eclipse as a tool integrator

- **Just presented at EclipseCon last week:**
  - **Eclipse plug-in to aid generating OpenMP and Pthreads code through Visual Programming**
    - **Javier Gonzalez Sanchez (Tecnologico de Monterrey, campus Guadalajara), Maria Elena (Helen) Chavez Echeagaray (Tecnologico de Monterrey campus Guadalajara)**
- Hope to integrate with PTP tools to provide visual building tools for OpenMP
- Eclipse makes this not difficult to do!

parallel tools platform
http://eclipse.org/ptp

Summary

- PTP is an open-source, extensible platform
  - based on Eclipse and its CDT (C/C++ Development Tools)
- PTP provides tools for parallel programming
  - OpenMP tools ease development on multi-core system



Ongoing Work

•Integration with more compilers/debuggers that support OpenMP is needed
  •Including multi-threaded debuggers

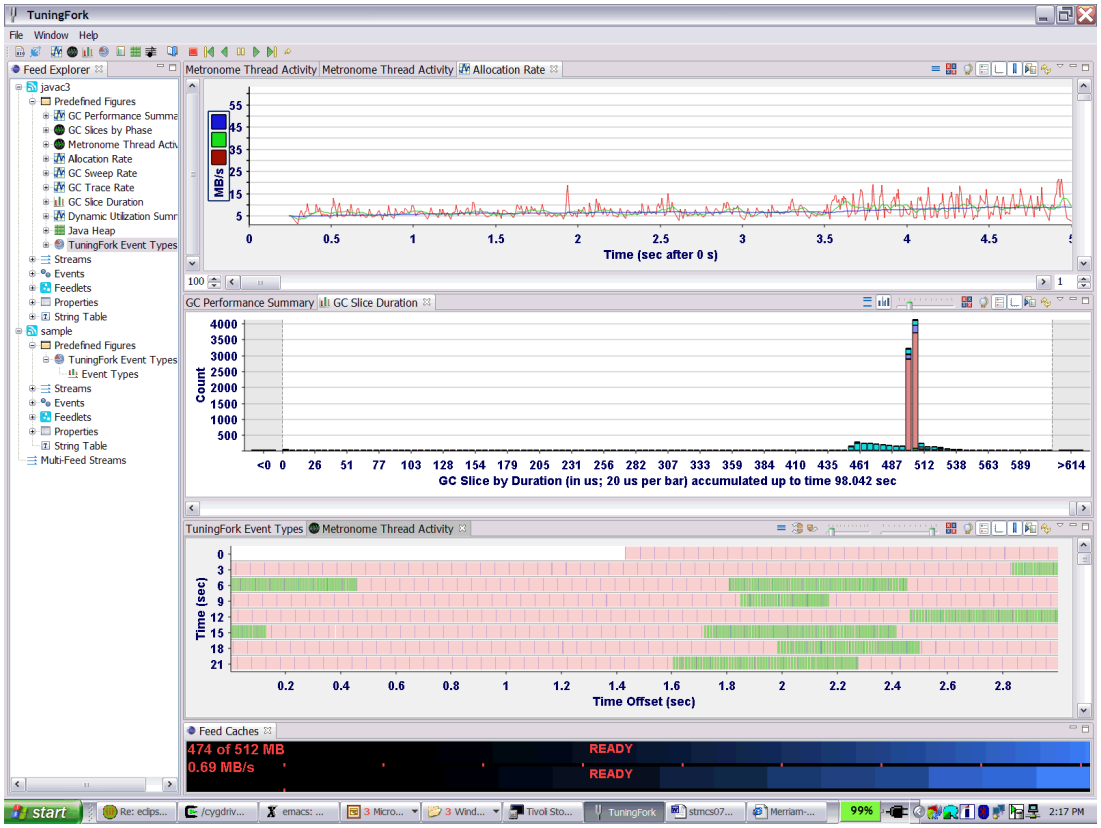
•Integration with IBM Cell* IDE (also Eclipse and CDT-based) is being studied

•More analysis tools for detecting common errors

•More feedback is needed on usefulness of features
  •Please download and try it!

backup

Performance Visualization Tools



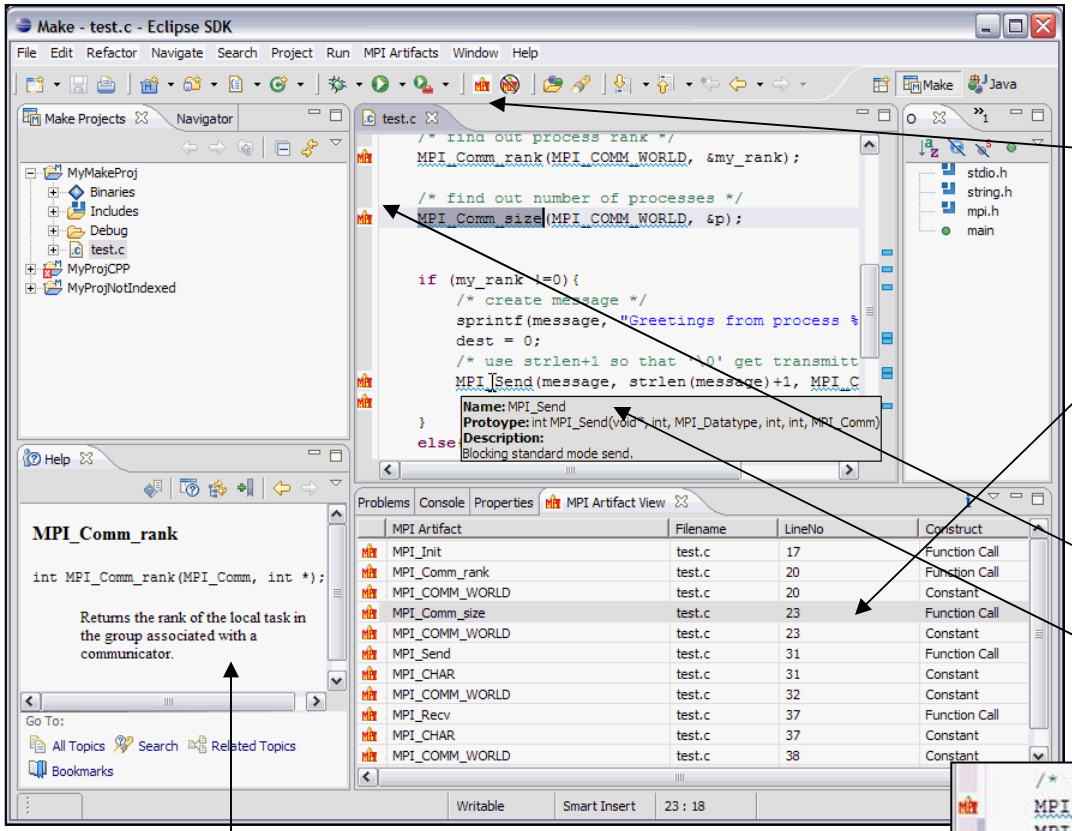
Visualization based on TuningFork:
http://www.alphaworks.ibm.com/tech/tuningfork

Demo Content backup

- The following slides summarize demo points

Parallel Language Development Tools: OpenMP tools – are similar to MPI tools shown here

Based on the CDT (C/C++ Development Toolkit),



Actions to find OpenMP Artifacts via Syntactic analysis

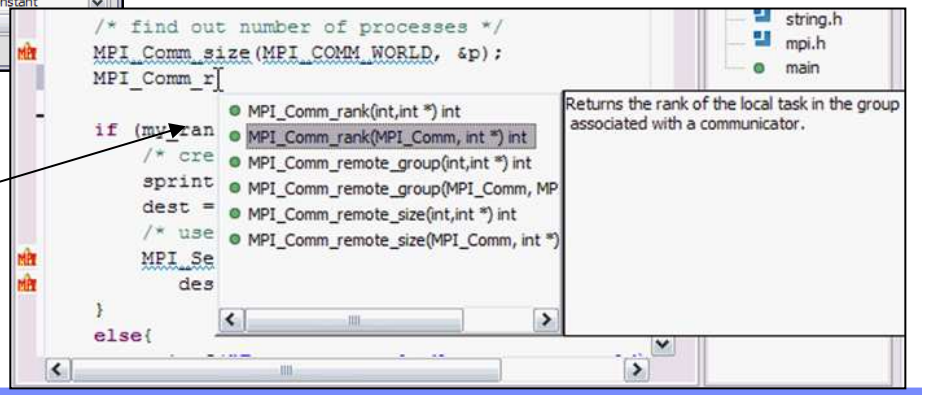
OpenMP Artifacts found by analysis

Source Markers for Navigation & ID

Mouse hover help And content assist

Context Sensitive Help (F1) provides API info

Content Assist Ctrl-space Suggests completions



OpenMP Tools

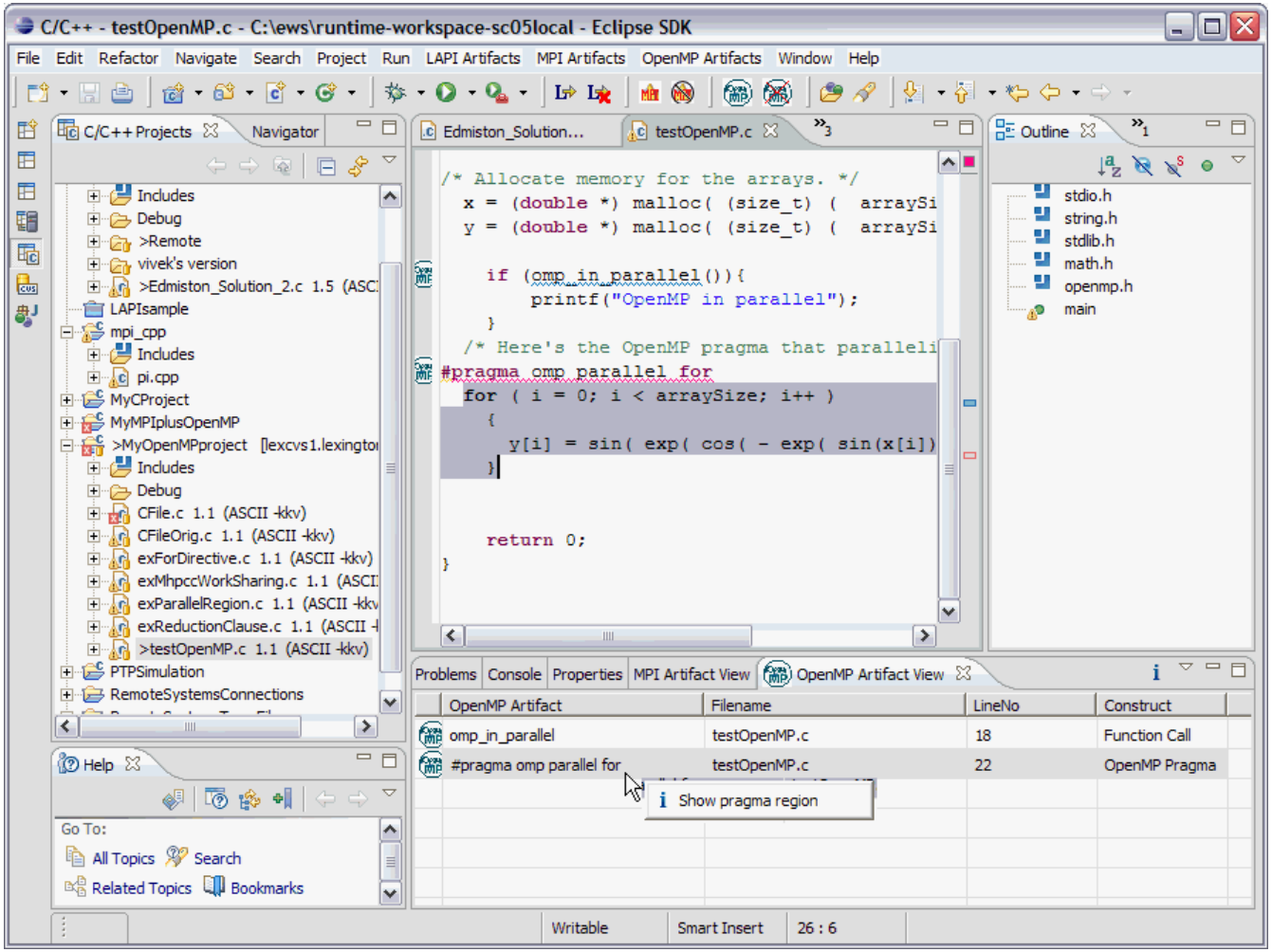
OpenMP - Simple, Portable, Scalable SMP Programming An API for multi-platform shared-memory parallel programming in C/C++ and Fortran.

- **Identify constructs**
- **List OpenMP constructs**
- **Link to source code**

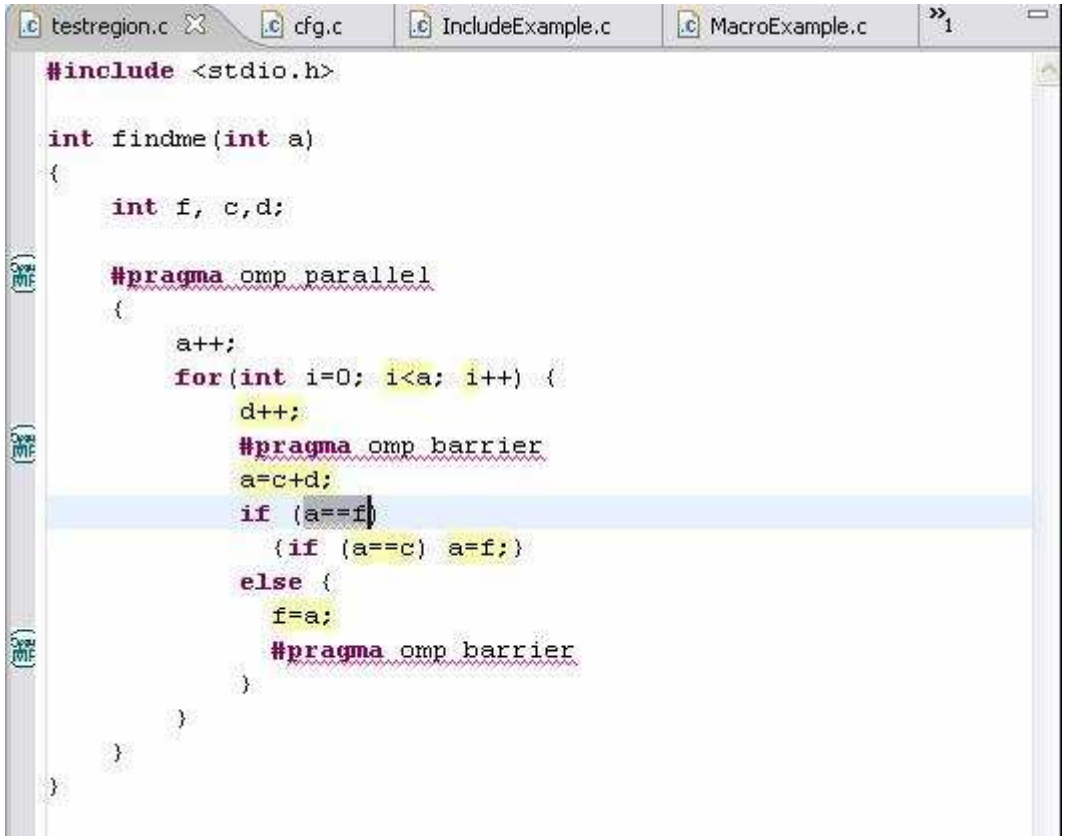
Analysis

- **identify scope of #pragma**
- **Identify common problems**
- **Concurrency analysis**

See next slide ….



OpenMP Concurrency Analysis

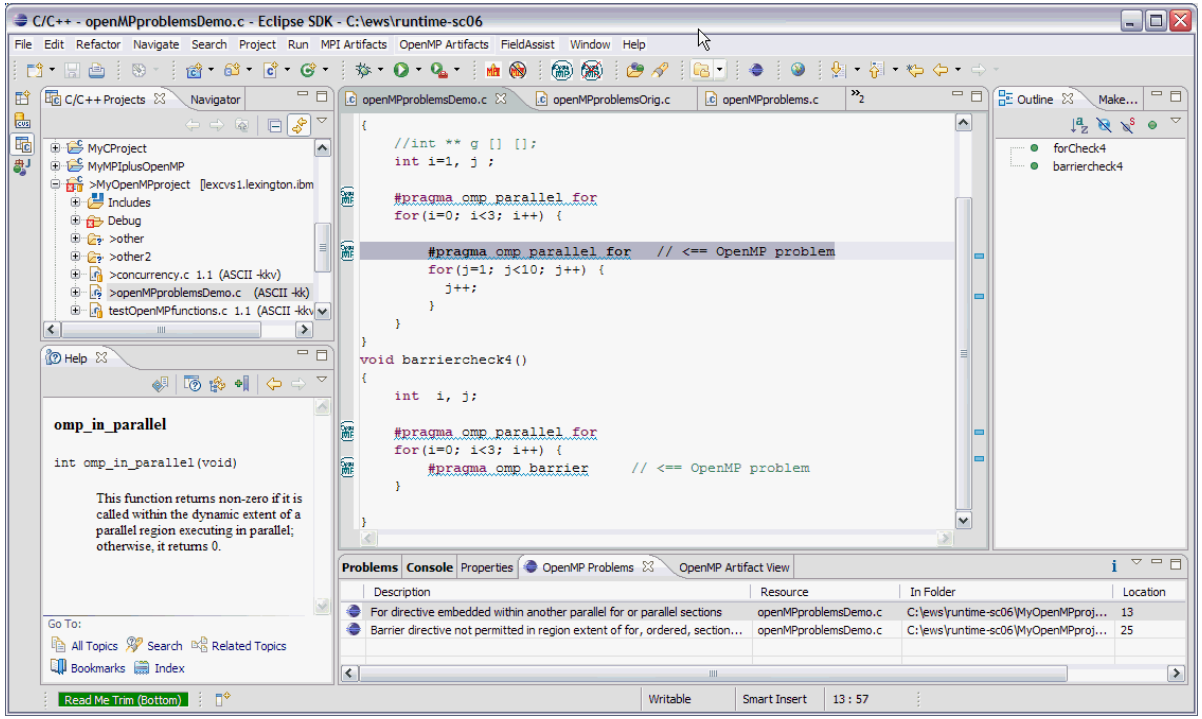


- Analysis of which statements could execute in parallel (based on concurrency analysis of Yuan Lin)

Possible future extension:

- Analysis to develop strategy for parallelizing

OpenMP problems



Types of problems targeted for analysis include:

- Parallel directive dynamically inside another parallel, establishes single thread context
- For directive embedded within critical, ordered, or master extents
- For directive embedded within another parallel for or parallel sections
- For directive embedded within another for, sections, or single directive
- Barrier directive not permitted in region extent of for, ordered, sections, single, master, and critical
- Master directive not permitted in dynamic extent of for, sections, or single directives
- Ordered directive not permitted in dynamic extent of critical region

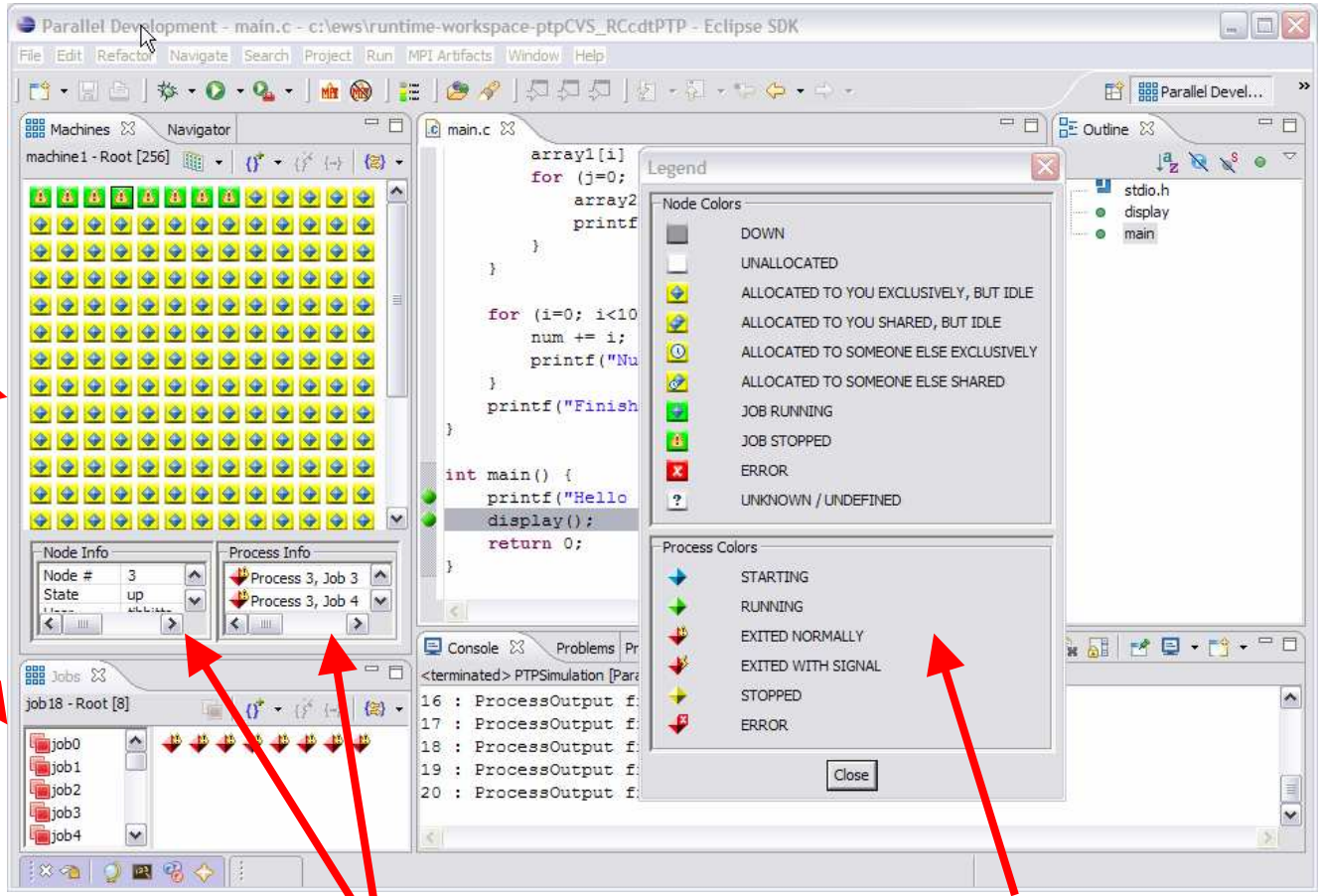
Along with the analysis for finding OpenMP artifacts, common problems are also located, and shown in the OpenMP Problems view.
Like the OpenMP Artifacts view, the OpenMP Problems view can be used to navigate to the source code line by double-clicking on the line in the problems view.

- **End of demo content**

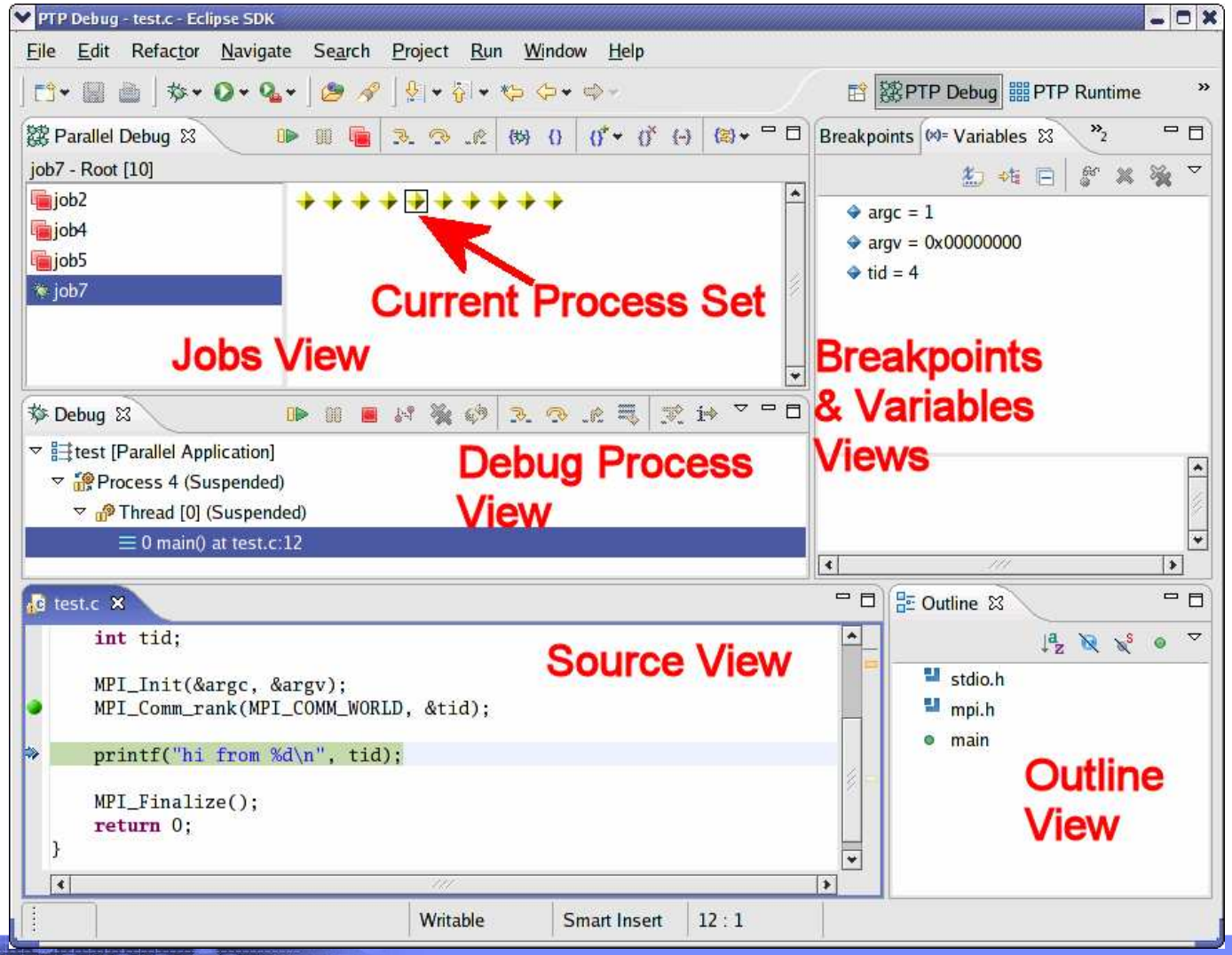
More backup

- Information about PTP Core features (for MPI)

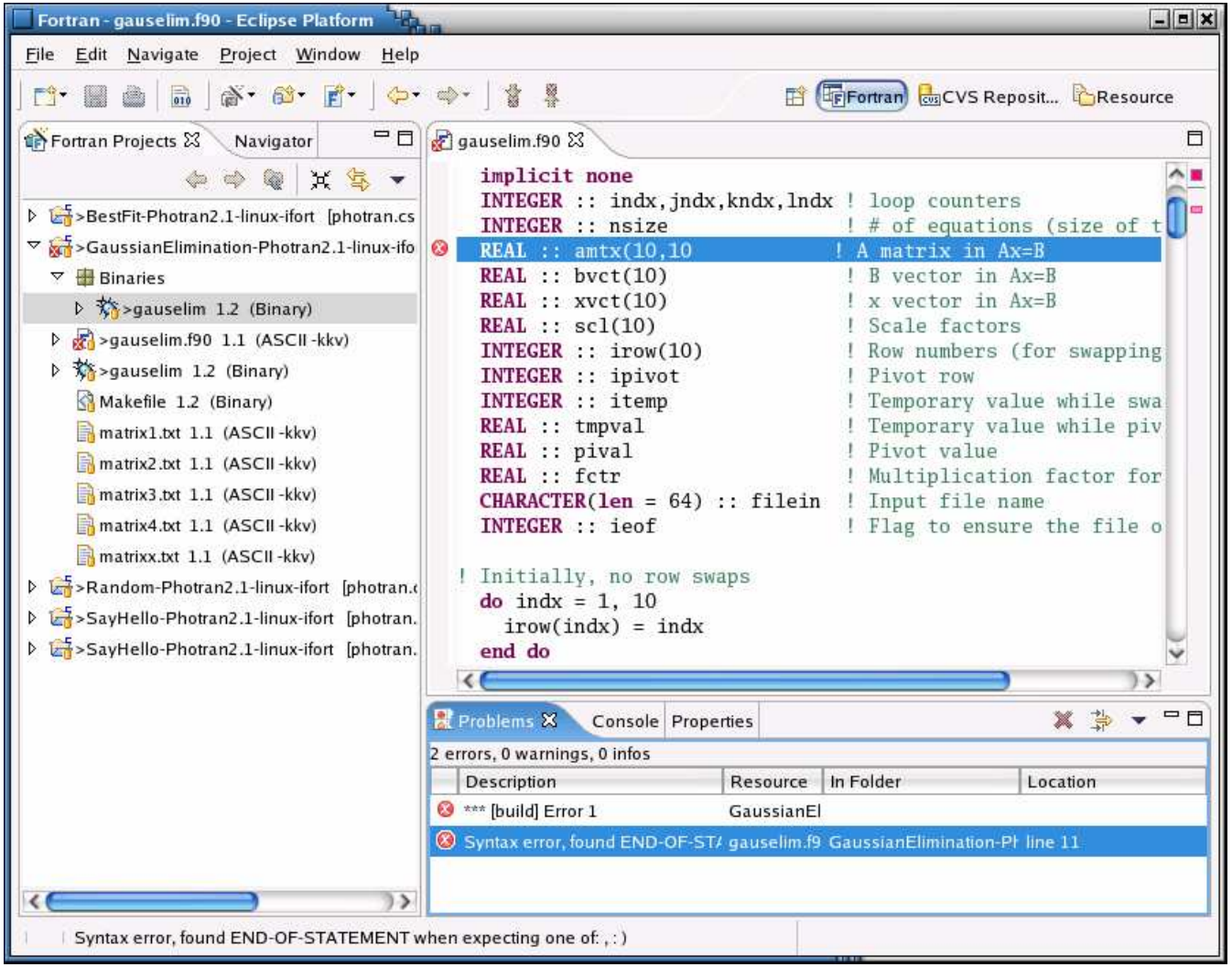
PTP – Parallel Tools Platform
Runtime Perspective



PTP – Parallel Tools Platform
Parallel Debugger



Fortran support: Photran status



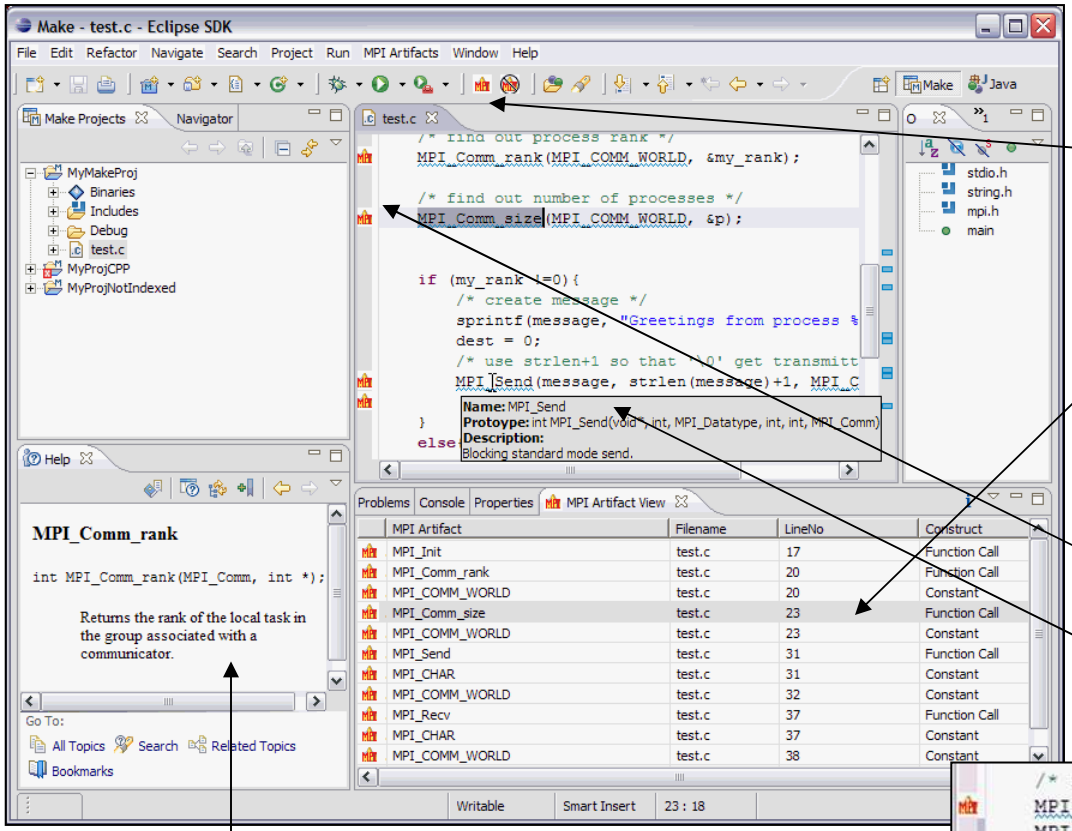
- Support for editing both Fortran-90 and Fortran-77.
- Good integration with cvs and make.
- Works with gdb, but debugger support needs improvement.

- Refactorings available soon
- Dependent on internal program representation work.

- http://eclipse.org/photran
- Photran is being developed at the University of Illinois at Urbana-Champaign and Los Alamos National Laboratory. For further information, contact: Brian Foote, Ralph Johnson, Jeff Overbey or Spiros Xanthos at UIUC, or Craig Rasmussen at LANL.

Parallel Language Development Tools: PTP MPI Development Tools

Based on the CDT (C/C++ Development Toolkit),



Actions to find MPI Artifacts via Static analysis

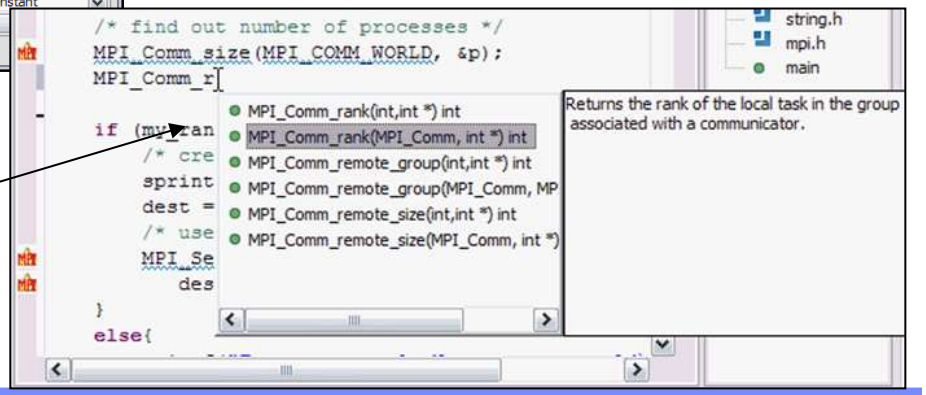
MPI Artifacts found by analysis

Source Markers for Navigation & ID

Mouse hover help And content assist

Context Sensitive Help (F1) provides API info

Content Assist Ctrl-space Suggests completions



OpenMP Tools

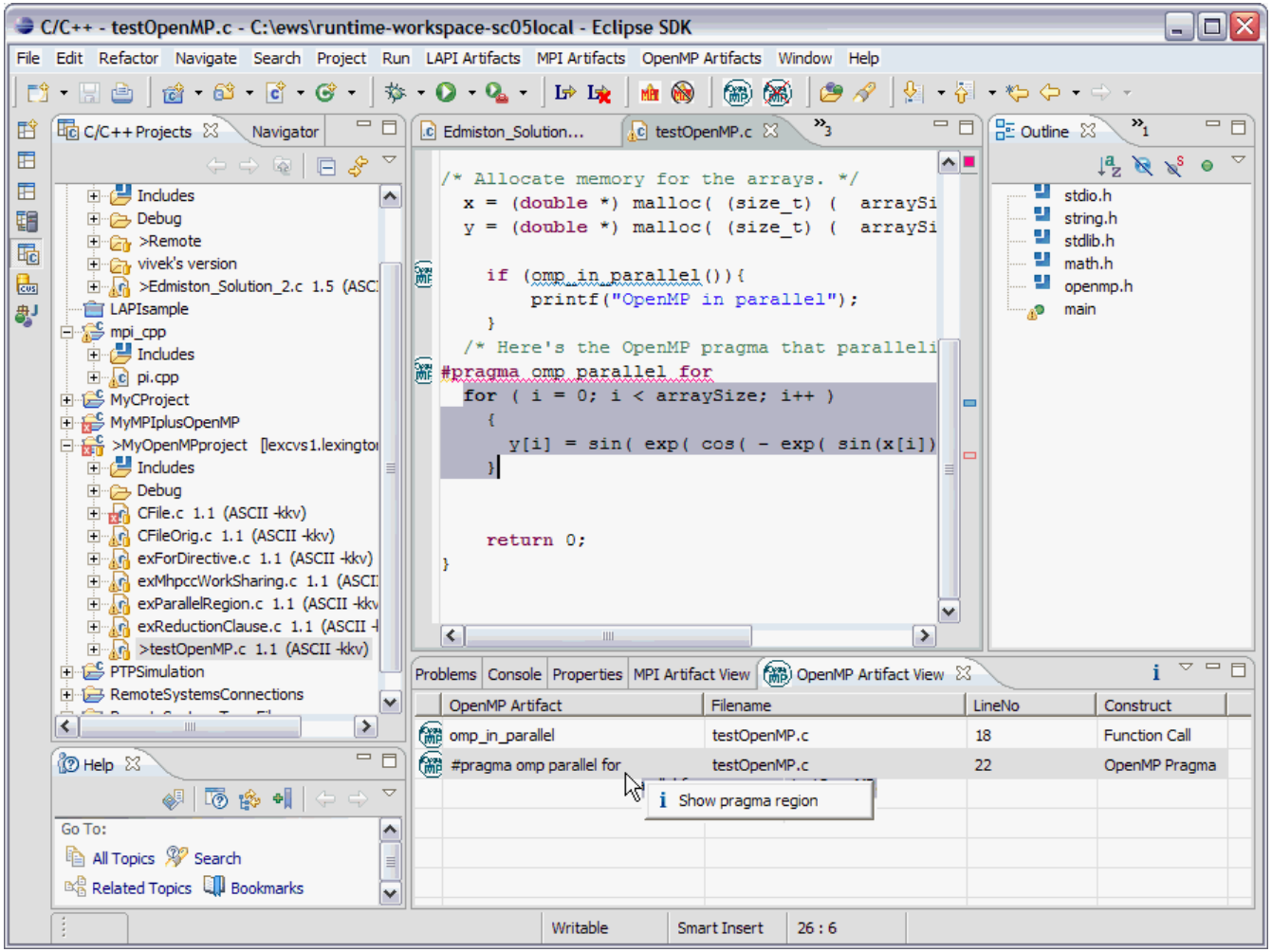
OpenMP - Simple, Portable, Scalable SMP Programming An API for multi-platform shared-memory parallel programming in C/C++ and Fortran.

- **Identify constructs**
- **List OpenMP constructs**
- **Link to source code**

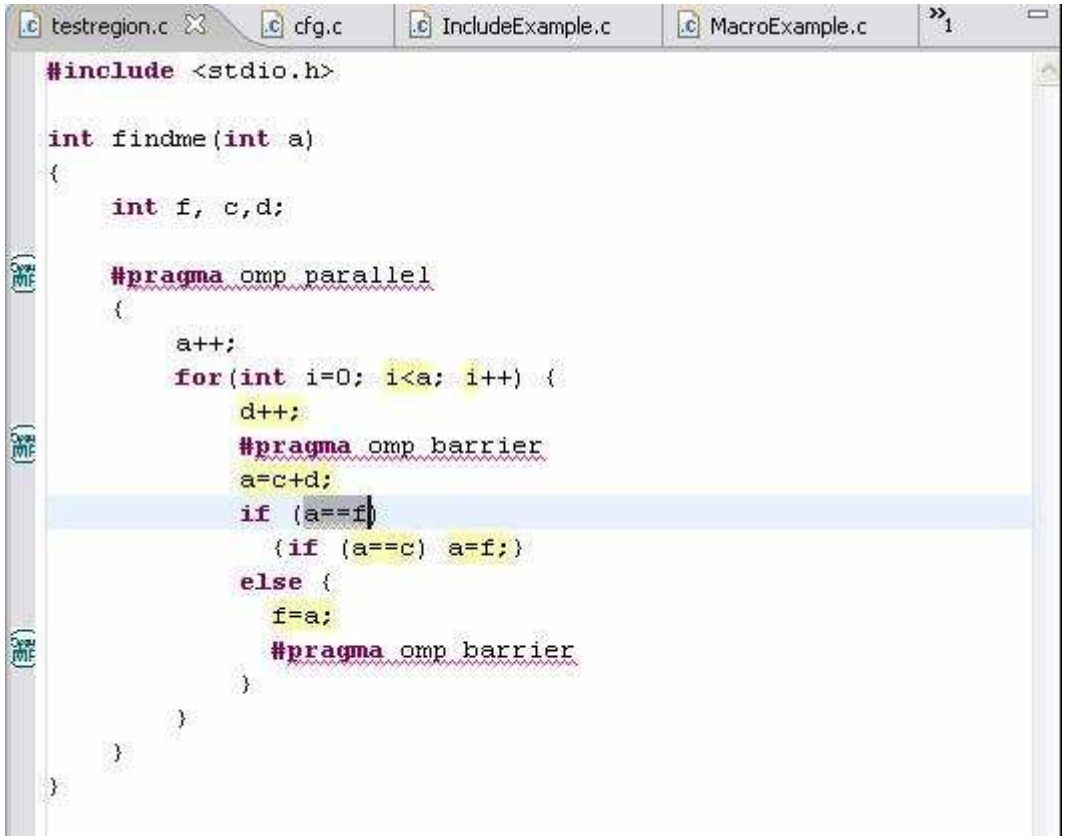
Analysis

- **identify scope of #pragma**
- **Identify common problems**
- **Concurrency analysis**

See next slide ….



OpenMP Concurrency Analysis

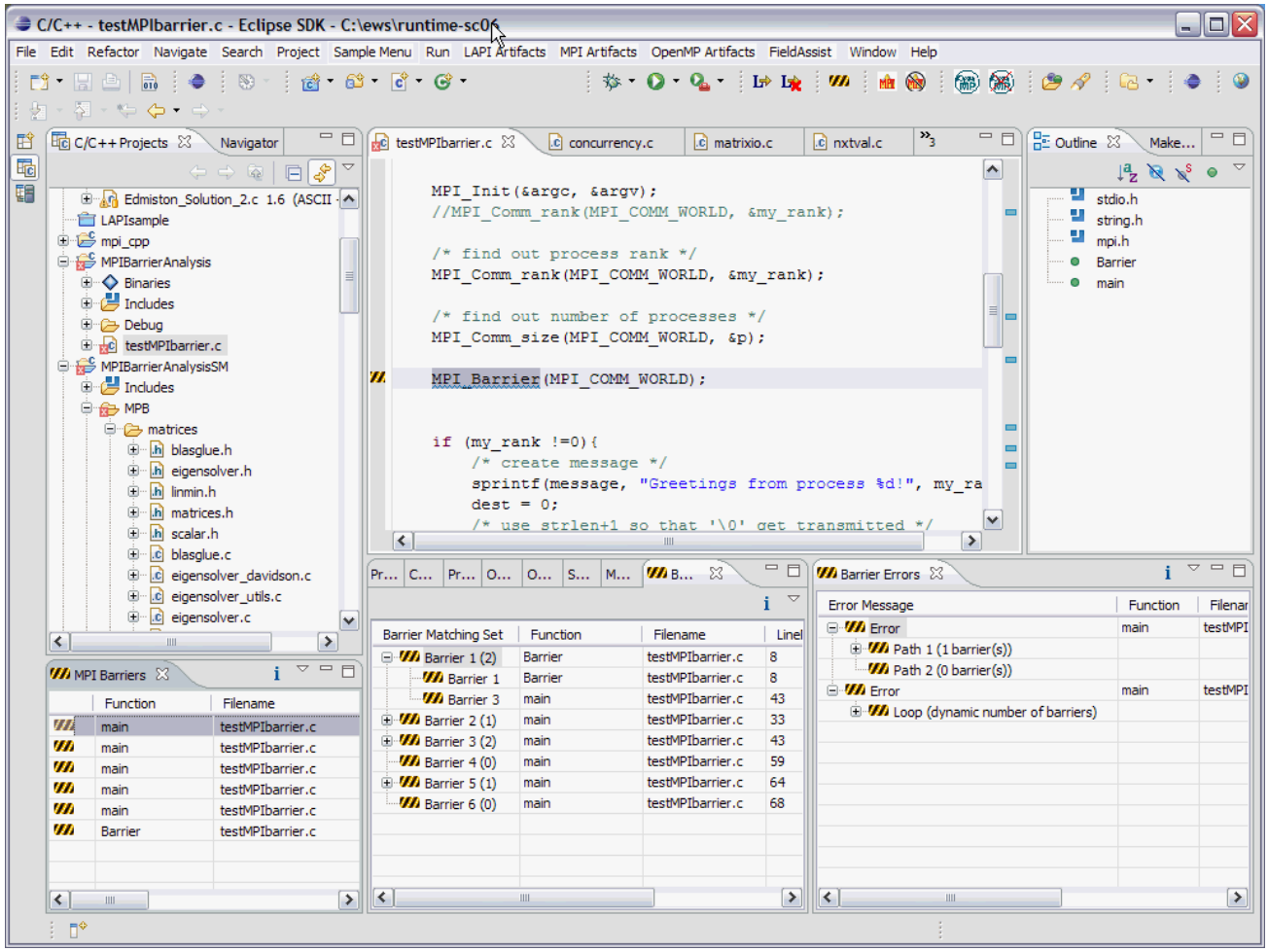


- Analysis of which statements could execute in parallel (based on concurrency analysis of Yuan Lin)

Possible future extension:

- Analysis to develop strategy for parallelizing

PTP PLDT: MPI Barrier Analysis



Contact: Evelyn Duesterwald, Yuan Zhang

Verify barrier synchronization in C/MPI programs

- Interprocedural static analysis.

- Output:
  1) For verified programs, lists barrier statements that synchronization together (match)

  2) For synchronization errors, reports counter example that illustrates and explains the error.

Eclipse PTP: Parallel Tools Platform
More Information

- http://eclipse.org/ptp
- User mailing list: ptp-dev@eclipse.org
- Developer mailing list: ptp-dev@eclipse.org

Related info:

- CDT (C/C++ tools): eclipse.org/cdt
- Photran (Fortran) eclipse.org/photran
- Open MPI open-mpi.org

- **PTP Tutorials**:
  - Supercomputing 2006
    (Nov. 11-17, Tampa)
    http://sc06.supercomputing.org/schedule/event_detail.php?evid=5063
  - LACSI (Los Alamos Computer Science Institute)
    Oct 17-19, Santa Fe New Mexico
    http://lacsi.rice.edu/symposium/
  - OSCON July 2007