



IBM Research

PTP - PLDT Parallel Language Development Tools Overview, Status & Plans

Beth Tibbitts

tibbitts@us.ibm.com

High Productivity Tools Group, IBM Research



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

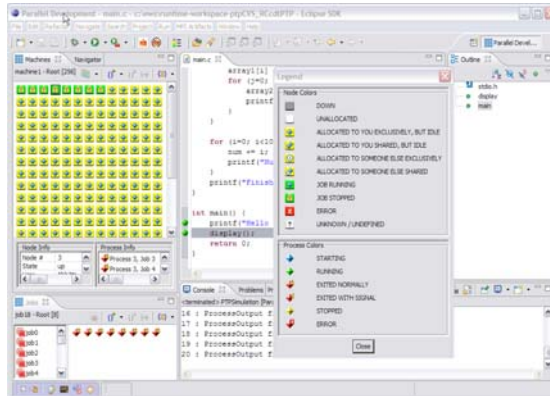
May 2007

© 2007 IBM
Corporation

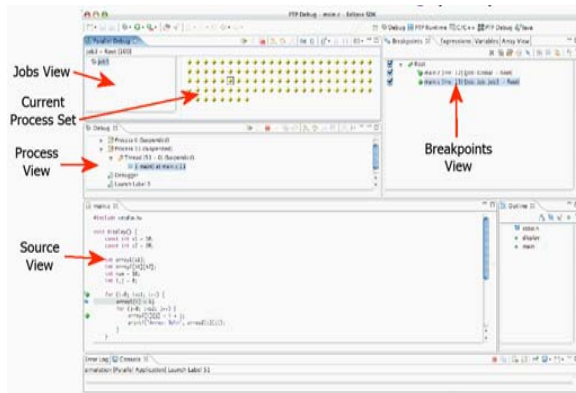
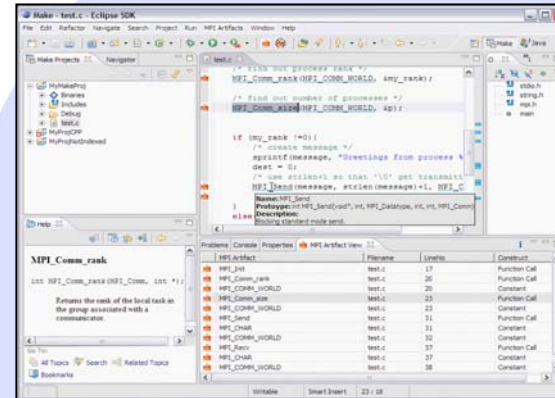
Eclipse PTP: Parallel Tools Platform

<http://eclipse.org/ptp>

Parallel Runtime



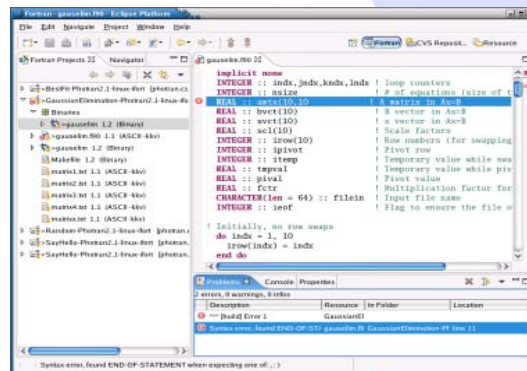
Parallel Language Dev. Tools (PLDT)



Parallel Debugger



Fortran Tools



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



Performance Tools *: based on TuningFork: <http://www.alphaworks.ibm.com/tech/tuningfork>

IBM Contributions

* Not yet publicly available on eclipse.org

Parallel Language Development Tools: PTP MPI Development Tools



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

MPI Artifact	Filename	LineNo	Construct
MPI_Init	test.c	17	Function Call
MPI_Comm_rank	test.c	20	Function Call
MPI_COMM_WORLD	test.c	20	Constant
MPI_Comm_size	test.c	23	Function Call
MPI_COMM_WORLD	test.c	23	Constant
MPI_Send	test.c	31	Function Call
MPI_CHAR	test.c	31	Constant
MPI_COMM_WORLD	test.c	32	Constant
MPI_Recv	test.c	37	Function Call
MPI_CHAR	test.c	37	Constant
MPI_COMM_WORLD	test.c	38	Constant

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

The screenshot shows the Eclipse IDE interface. The main editor displays the following C++ code:

```

/* Allocate memory for the arrays. */
x = (double *) malloc( (size_t) ( arraySi
y = (double *) malloc( (size_t) ( arraySi

if (omp_in_parallel()) {
    printf("OpenMP in parallel");
}

/* Here's the OpenMP pragma that paralleli
#pragma omp parallel for
for ( i = 0; i < arraySize; i++ )
{
    y[i] = sin( exp( cos( - exp( sin(x[i]
}

return 0;

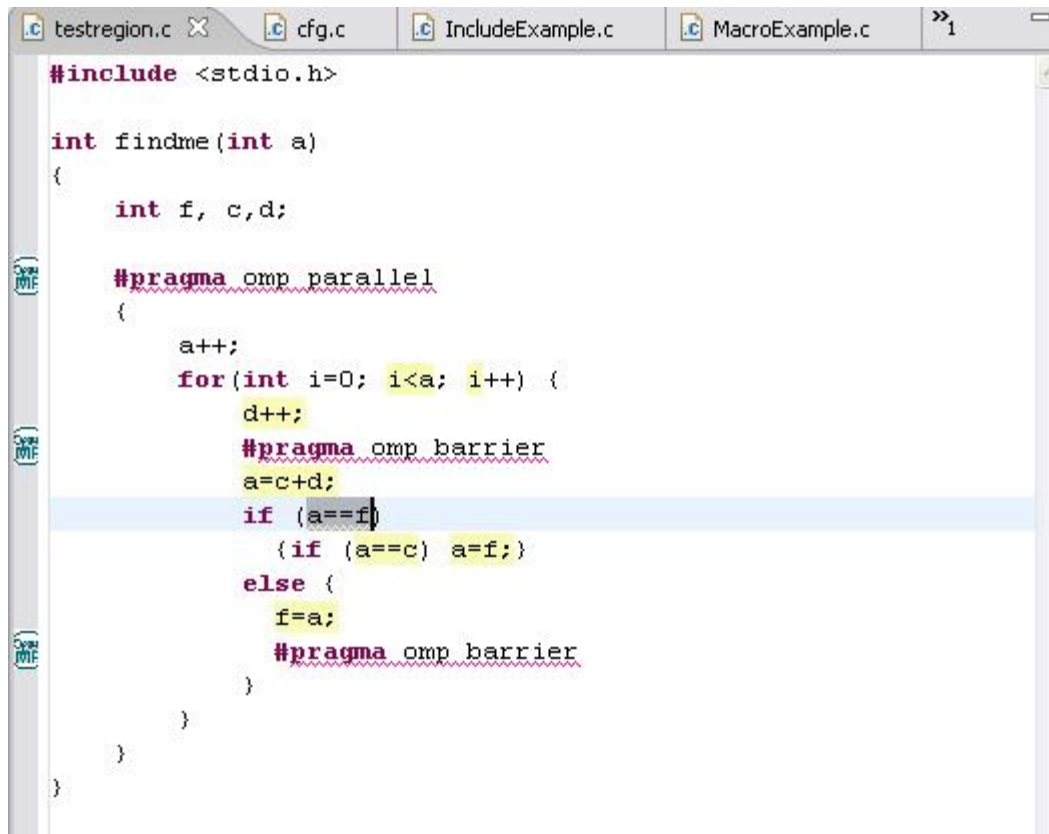
```

The OpenMP Artifact View at the bottom shows the following table:

OpenMP Artifact	Filename	LineNo	Construct
omp_in_parallel	testOpenMP.c	18	Function Call
#pragma omp parallel for	testOpenMP.c	22	OpenMP Pragma

A red arrow points from the '#pragma omp parallel for' entry in the table to the corresponding code in the editor. A tooltip 'Show pragma region' is visible over the arrow.

OpenMP Concurrency Analysis



```
.c testregion.c x .c cfg.c .c IncludeExample.c .c MacroExample.c »1
#include <stdio.h>

int findme(int a)
{
    int f, c,d;

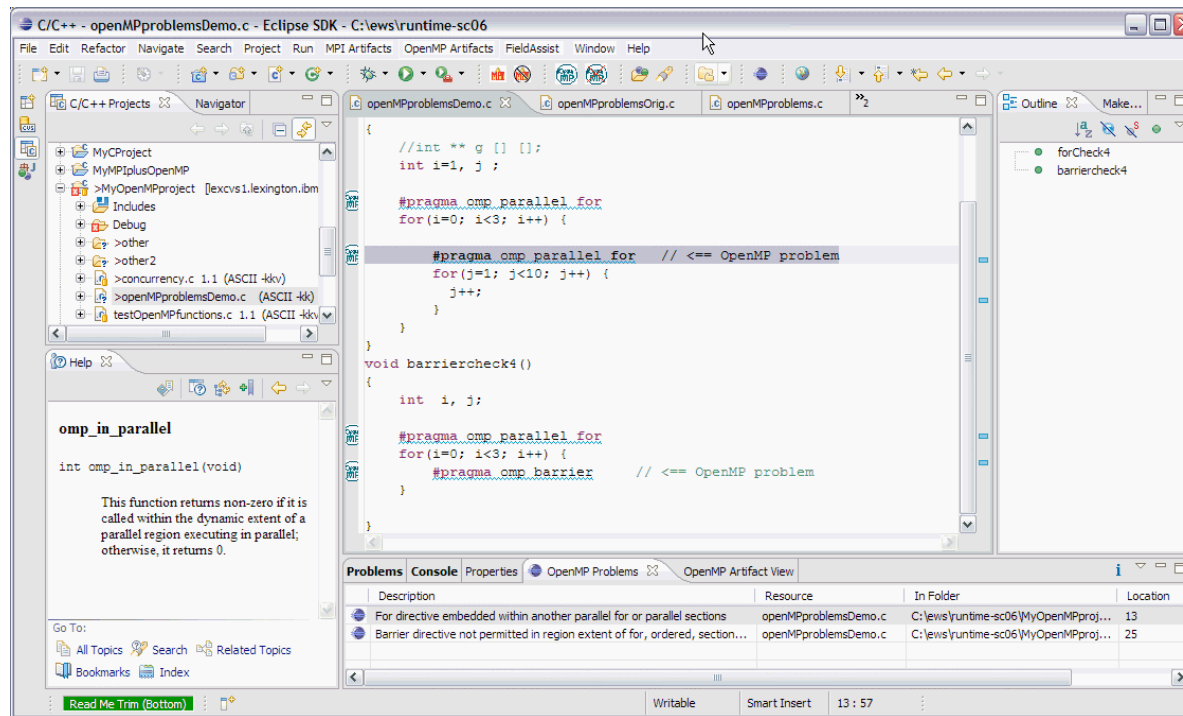
    #pragma omp parallel
    {
        a++;
        for(int i=0; i<a; i++) {
            d++;
            #pragma omp barrier
            a=c+d;
            if (a==f)
                (if (a==c) a=f;)
            else {
                f=a;
                #pragma omp barrier
            }
        }
    }
}
```

- 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:

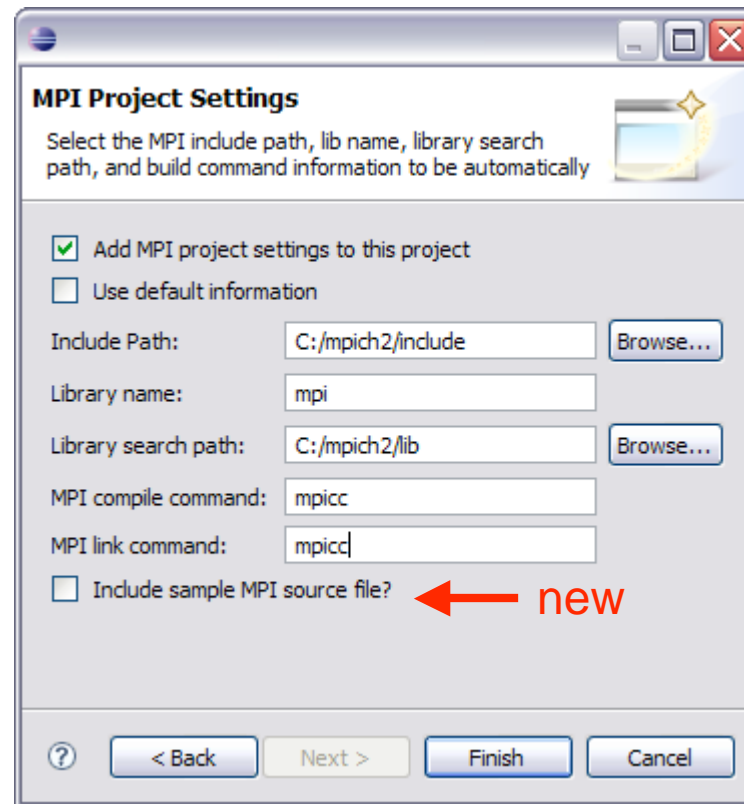
- Parallel directive dynamically inside another parallel, establishes single thread context
- For directive embedded within critical, ordered, or master extents
- For directive embedded within another parallel for or parallel sections
- For directive embedded within another for, sections, or single directive
- 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.

While these are along the lines of what compilers would probably also find, they are a hint of future features to find less obvious problems.

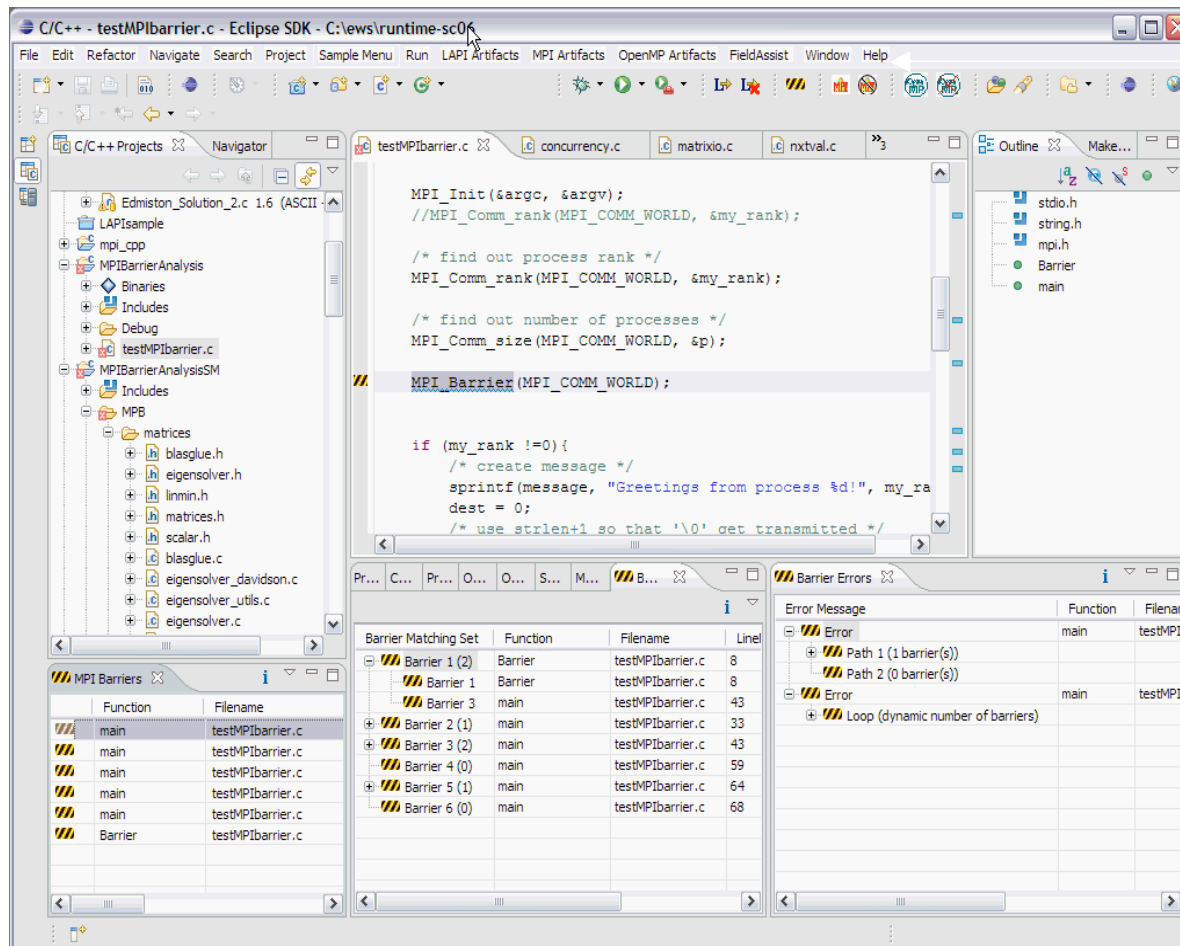
New MPI Project Wizard

- Simplifies creation of new (managed make) MPI projects
- Would it be more useful as a “New MPI Project” wizard instead of an added page to “New C Project” wizard?
- And “New OpenMP Project” etc?



Parallel Language Development Tools: MPI Barrier Verification Tool

Action to run
Barrier Verifier



Verify barrier synchronization in C/MPI programs

- Synchronization errors lead to deadlocks and stalls.
- Programmers may have to spend hours trying to find the source of a deadlock
- Static verification tools help to eliminate errors before the program is executed

Contact: Evelyn Duesterwald, Yuan Zhang

MPI Barrier Verification Tool

Program Verified

The screenshot shows the Eclipse IDE with the following components:

- Editor:** Displays the source code for `matrixio.c`. Key lines include:


```

                if (mpi_is_master()) {
                    sub_id = H5Gcreate(id, name, 0 /* ==> default size */);
                    matrixio_write_string_attr(sub_id, "description", description);

                    H5Fflush(sub_id, H5F_SCOPE_GLOBAL);

                    MPI_Barrier(MPI_COMM_WORLD);
                }
            
```
- Navigator:** Shows the project structure on the left, including `matrixio.h`, `fieldio.c`, and `matrixio.c`.
- Outline:** Shows the symbol table on the right, listing various functions and macros.
- Barrier Matches Table:** Located at the bottom, it lists the following data:

Barrier Match	Function	Filename	LineNo	IndexNo
Barrier 1 (2)	matrixio_create_subdataset	matrixio.c	387	1
Barrier 2 (2)	matrixio_create_subdataset	matrixio.c	390	2
Barrier 1 (1)	matrixio_create_subdataset	matrixio.c	387	1
Barrier 2 (2)	matrixio_create_subdataset	matrixio.c	390	2
Barrier 3 (1)	matrixio_create_subdataset	matrixio.c	456	3

Barrier Match Listing

Show Barrier Matching Set

MPI Barrier Verification Tool

Barrier Synchronization Error Found

The screenshot shows the Eclipse IDE with the following components:

- Code Editor:** Shows the source code for `matrixio.c`. A red arrow points to the `if (mpi_is_master()) {` block, which contains an `MPI_Barrier(MPI_COMM_WORLD);` call.
- Outline:** Lists project files including `matrixio.h`, `matrixio.c`, and various utility files.
- Console:** Displays MPI barrier verification results. A red circle highlights an error entry:

Error Message	Function	Filename	LineNo	IndexNo
Barrier 1 (1 barrier(s))	matrixio_create...	matrixio.c	381	0
Path 1 (1 barrier(s))	Barrier 1	matrixio_create...	387	1
Path 2 (0 barrier(s))			0	0

Barrier Synchronization Error Counter Example

Error Listing