BLUE WATERS

SUSTAINED PETASCALE COMPUTING

May 21, 2013

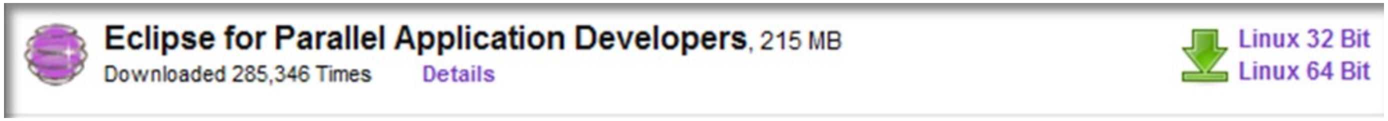
Eclipse IDE for Blue Waters, demos:

- Eclipse Kepler release
  - Cray Loopmark
  - OpenACC support
- Nvidia Nsight for C/CUDA

GREAT LAKES CONSORTIUM FOR PETASCALE COMPUTATION

Eclipse kepler release

- Eclipse downloads are available for Linux, Mac and Windows platforms
- www.eclipse.org
  - Downloads
    - Developer builds
      - Eclipse for Parallel Application Developers
        - Select your OS and architecture (32 or 64 bit)

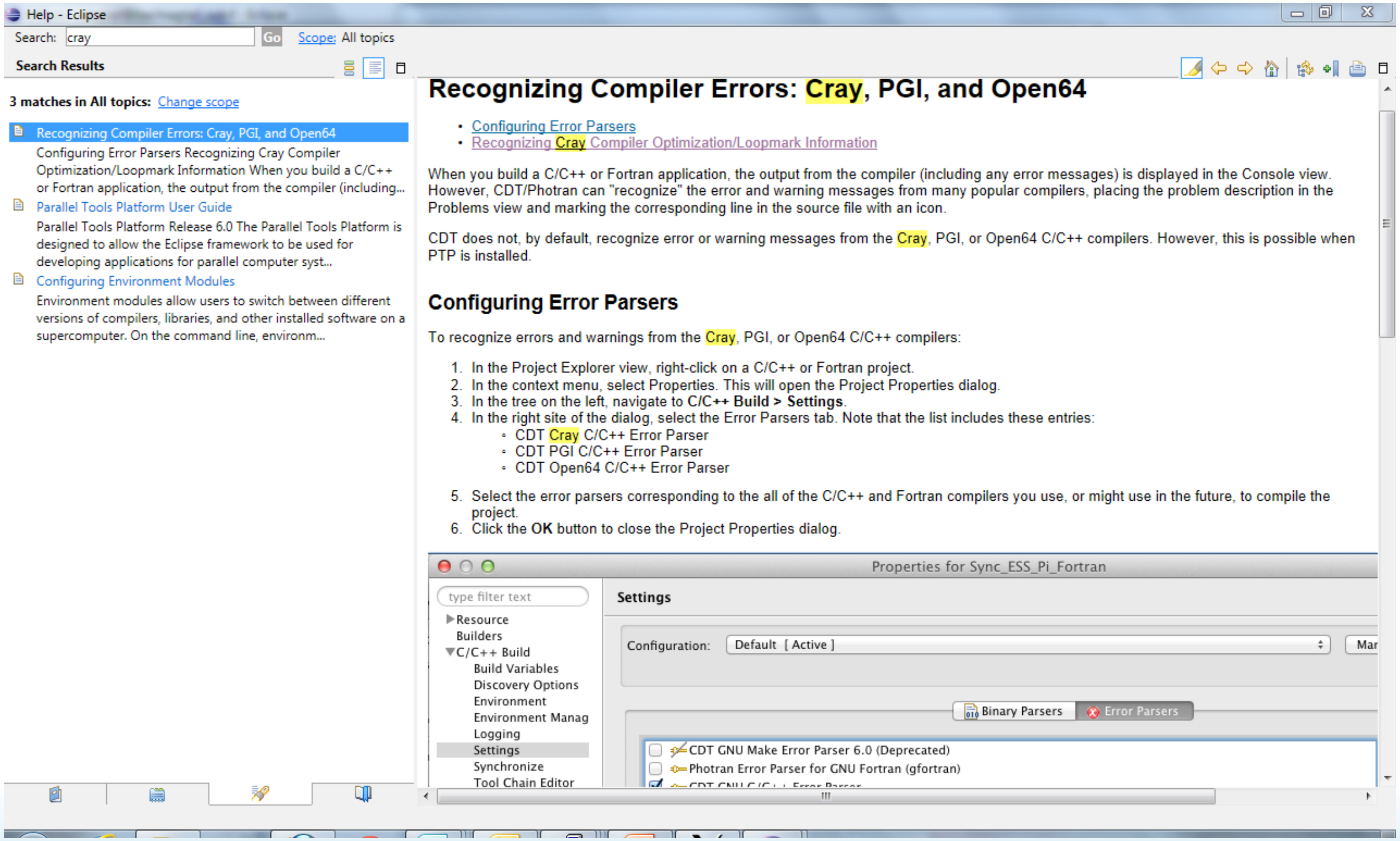


Eclipse for Parallel Application Developers, 215 MB
Downloaded 285,346 Times Details

Synchronized projects with loopmark and Openacc

- Eclipse supports client-server development via synchronized projects
  - Create a new synchronized project
    - Makefile project (empty)
    - Setup filters if the project contains large files because /usr/bin/git doesn’t handle large files well
    - Fix remote include paths if desired
  - Confused? Look at eclipse help—search box.

Eclipse help: searching for Cray



3 matches in All topics: Change scope

- Recognizing Compiler Errors: Cray, PGI, and Open64
  Configuring Error Parsers Recognizing Cray Compiler Optimization/Loopmark Information When you build a C/C++ or Fortran application, the output from the compiler (including...
- Parallel Tools Platform User Guide
  Parallel Tools Platform Release 6.0 The Parallel Tools Platform is designed to allow the Eclipse framework to be used for developing applications for parallel computer syst...
- Configuring Environment Modules
  Environment modules allow users to switch between different versions of compilers, libraries, and other installed software on a supercomputer. On the command line, environm...

Recognizing Compiler Errors: Cray, PGI, and Open64

- Configuring Error Parsers
- Recognizing Cray Compiler Optimization/Loopmark Information

When you build a C/C++ or Fortran application, the output from the compiler (including any error messages) is displayed in the Console view. However, CDT/Photran can "recognize" the error and warning messages from many popular compilers, placing the problem description in the Problems view and marking the corresponding line in the source file with an icon.

CDT does not, by default, recognize error or warning messages from the Cray, PGI, or Open64 C/C++ compilers. However, this is possible when PTP is installed.

Configuring Error Parsers

To recognize errors and warnings from the Cray, PGI, or Open64 C/C++ compilers:

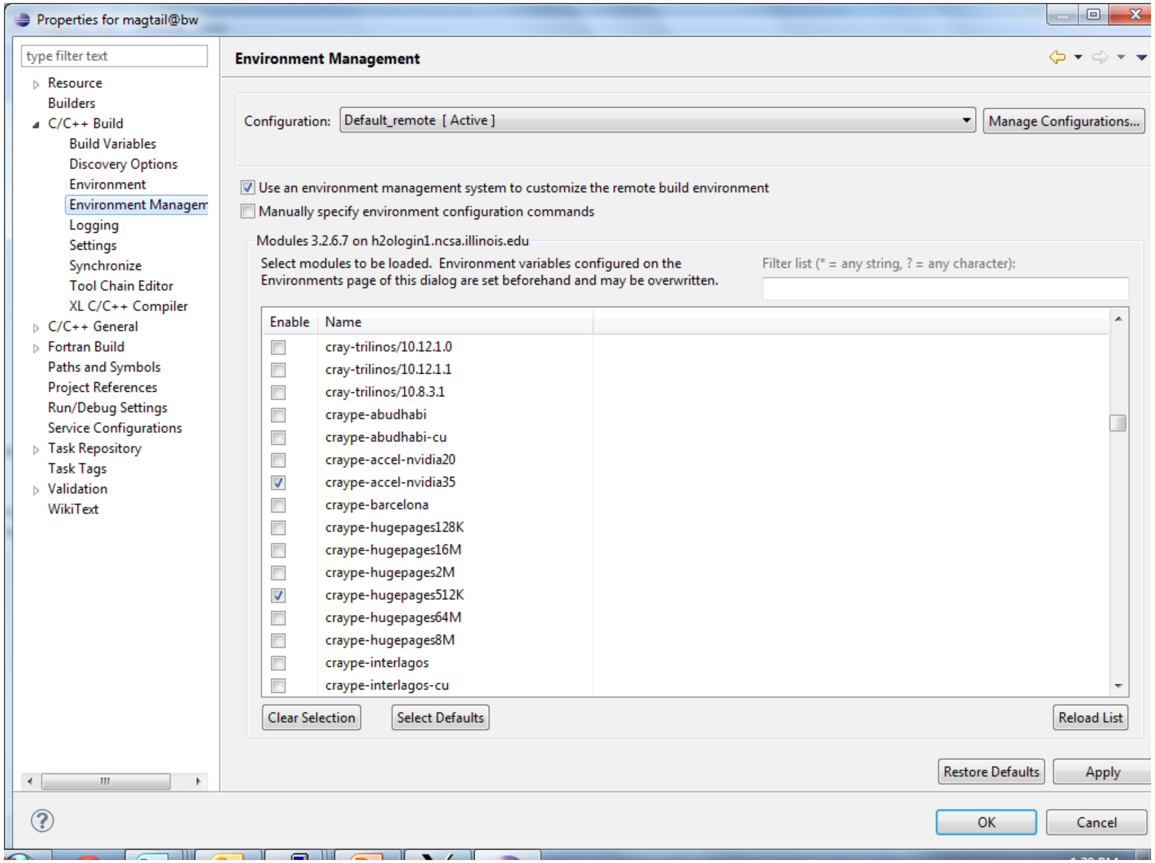
1. In the Project Explorer view, right-click on a C/C++ or Fortran project.
2. In the context menu, select Properties. This will open the Project Properties dialog.
3. In the tree on the left, navigate to **C/C++ Build > Settings**.
4. In the right site of the dialog, select the Error Parsers tab. Note that the list includes these entries:
   - CDT Cray C/C++ Error Parser
   - CDT PGI C/C++ Error Parser
   - CDT Open64 C/C++ Error Parser
5. Select the error parsers corresponding to the all of the C/C++ and Fortran compilers you use, or might use in the future, to compile the project.
6. Click the **OK** button to close the Project Properties dialog.

Configuration: Default [Active]

□ CDT GNU Make Error Parser 6.0 (Deprecated)
□ Photran Error Parser for GNU Fortran (gfortran)
☑ CDT GNU C/C++ Error Parser

Working with modules (Blue Waters and Xsede)

- Project
  - Properties
    - c/c++ build (also for fortran)
      - Environment management



Configuration: Default_remote [Active]

☑ Use an environment management system to customize the remote build environment

□ Manually specify environment configuration commands

Modules 3.2.6.7 on h2ologin1.ncsa.illinois.edu

Select modules to be loaded. Environment variables configured on the Environments page of this dialog are set beforehand and may be overwritten.

Filter list (* = any string, ? = any character):

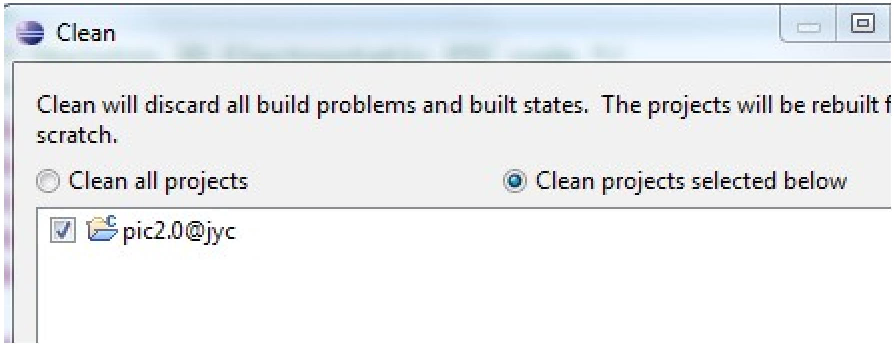
Enable	Name
□	cray-trilinos/10.12.1.0
□	cray-trilinos/10.12.1.1
□	cray-trilinos/10.8.3.1
□	craype-abudhabi
□	craype-abudhabi-cu
□	craype-accel-nvidia20
☑	craype-accel-nvidia35
□	craype-barcelona
□	craype-hugepages128K
□	craype-hugepages16M
□	craype-hugepages2M
☑	craype-hugepages512K
□	craype-hugepages64M
□	craype-hugepages8M
□	craype-interlagos
□	craype-interlagos-cu

Cdt editor tips

- Function calls
  - Hover over a function or subroutine to see definition
  - Select it to see occurrences
  - Right-click for more options like call hierarchy
- For line numbering, right click near the left edge of the edit window
- Tab indent (un-indent) code sections

Driving makefile: clean and build

- Project menu
  - Clean
  - Build
- Hammer time (build)

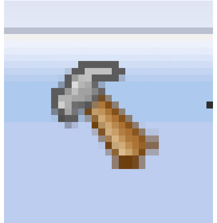


Clean

Clean will discard all build problems and built states. The projects will be rebuilt f
scratch.

○ Clean all projects ● Clean projects selected below

☑ pic2.0@jyc



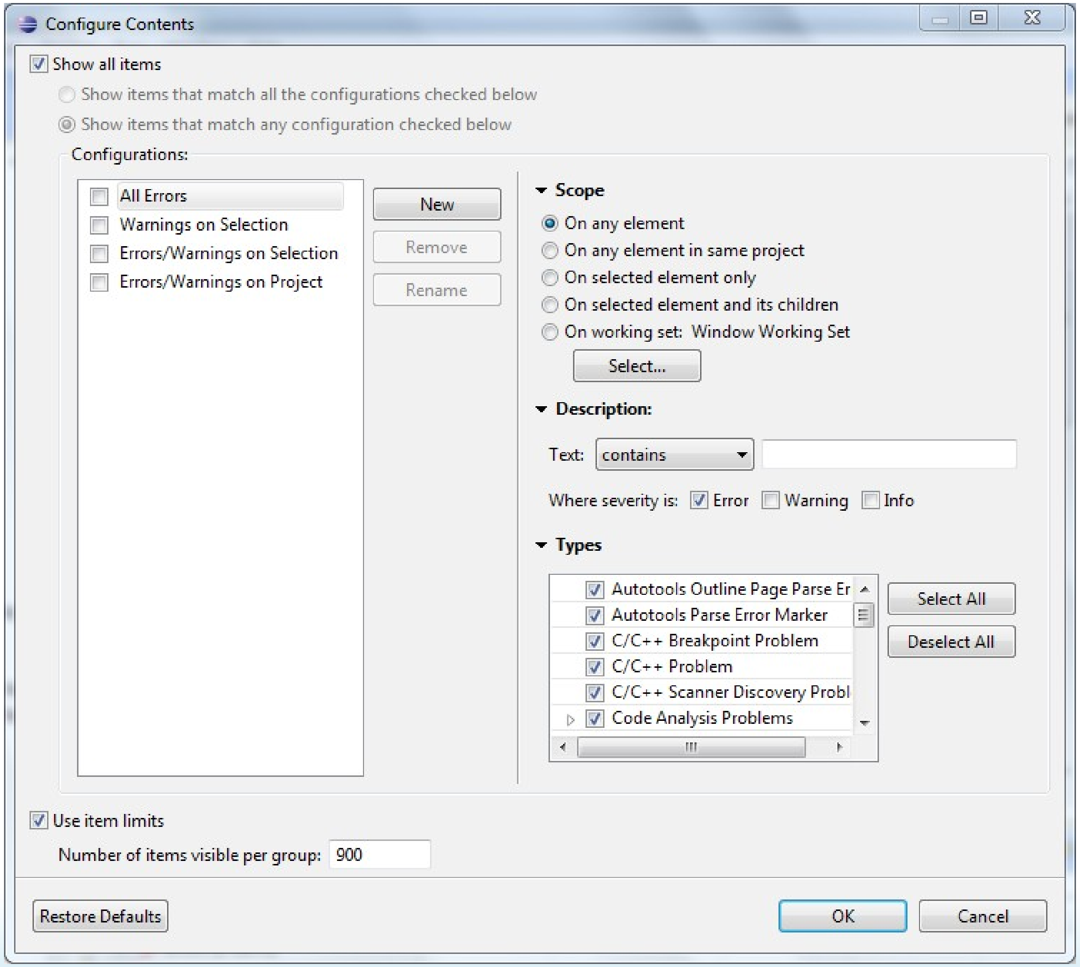
- There are often multiple methods of doing the same thing in eclipse: it’s the unix of IDEs

Cray loopmark demo

- Cray c and fortran compilers can annotate source code with compiler optimization information
- Both compilers can also emit optimization messages to stderr
  - Drop –g as it inhibits optimization
  - c/c++
    - -h msgs [negmsgs will report unoptimized code]
  - Fortran
    - -O msgs

Optimization report info

- Problems view
  - Info
    - Defaults to 100 items
    - view menu -> configure contents (to increase)



Configure Contents

☑ Show all items

○ Show items that match all the configurations checked below

● Show items that match any configuration checked below

Configurations:

☐ All Errors
☐ Warnings on Selection
☐ Errors/Warnings on Selection
☐ Errors/Warnings on Project

Scope

● On any element
○ On any element in same project
○ On selected element only
○ On selected element and its children
○ On working set: Window Working Set

Description:

Text: contains

Where severity is: ☑ Error ☐ Warning ☐ Info

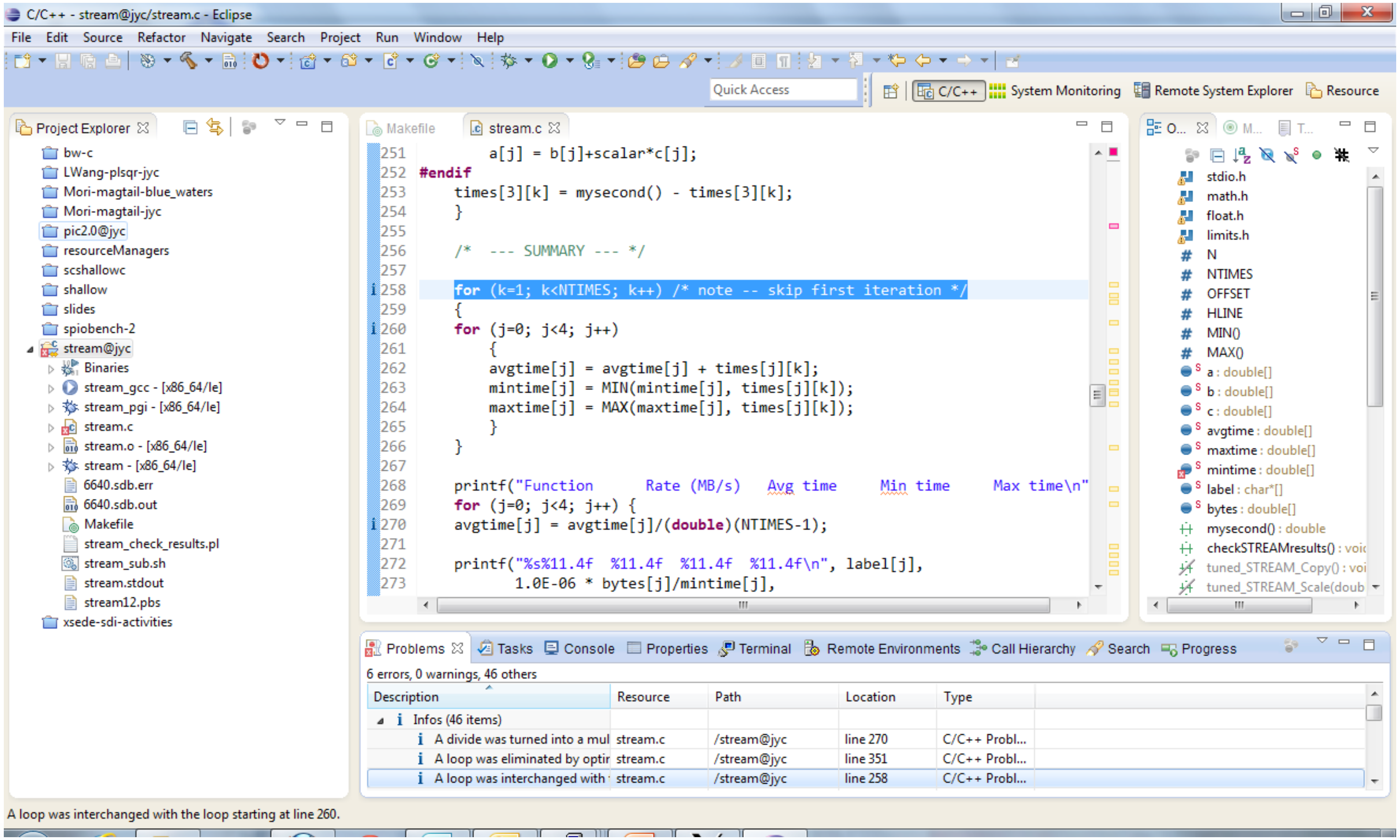
Types

☑ Autotools Outline Page Parse Er
☑ Autotools Parse Error Marker
☑ C/C++ Breakpoint Problem
☑ C/C++ Problem
☑ C/C++ Scanner Discovery Probl
☑ Code Analysis Problems

☑ Use item limits

Number of items visible per group: 900

C optimization report view



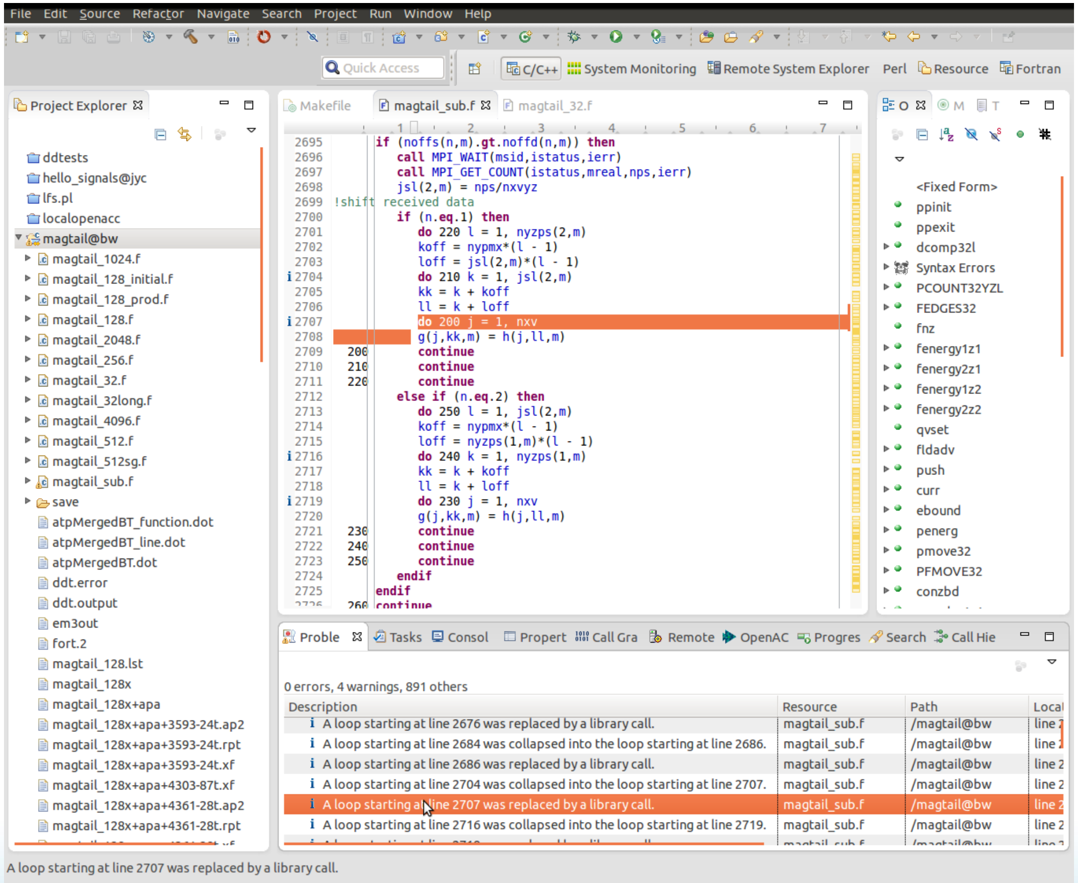
```
251         a[j] = b[j]+scalar*c[j];
252 #endif
253     times[3][k] = mysecond() - times[3][k];
254     }
255
256     /*  --- SUMMARY --- */
257
258     for (k=1; k<NTIMES; k++) /* note -- skip first iteration */
259     {
260     for (j=0; j<4; j++)
261         {
262         avgtime[j] = avgtime[j] + times[j][k];
263         mintime[j] = MIN(mintime[j], times[j][k]);
264         maxtime[j] = MAX(maxtime[j], times[j][k]);
265         }
266     }
267
268     printf("Function      Rate (MB/s)   Avg time     Min time     Max time\n"
269     for (j=0; j<4; j++) {
270     avgtime[j] = avgtime[j]/(double)(NTIMES-1);
271
272     printf("%s%11.4f  %11.4f  %11.4f  %11.4f\n", label[j],
273            1.0E-06 * bytes[j]/mintime[j],
```

6 errors, 0 warnings, 46 others

Description	Resource	Path	Location	Type
Infos (46 items)				
A divide was turned into a mul	stream.c	/stream@jyc	line 270	C/C++ Probl...
A loop was eliminated by optir	stream.c	/stream@jyc	line 351	C/C++ Probl...
A loop was interchanged with	stream.c	/stream@jyc	line 258	C/C++ Probl...

A loop was interchanged with the loop starting at line 260.

Fortran optimization report view



```
2695       if (noffs(n,m).gt.noffd(n,m)) then
2696          call MPI_WAIT(msid,istatus,ierr)
2697          call MPI_GET_COUNT(istatus,mreal,nps,ierr)
2698          jsl(2,m) = nps/nxvyz
2699 !shift received data
2700          if (n.eq.1) then
2701             do 220 l = 1, nyzps(2,m)
2702             koff = nypmx*(l - 1)
2703             loff = jsl(2,m)*(l - 1)
2704             do 210 k = 1, jsl(2,m)
2705             kk = k + koff
2706             ll = k + loff
2707             do 200 j = 1, nxv
2708             g(j,kk,m) = h(j,ll,m)
2709   200       continue
2710   210       continue
2711   220       continue
2712          else if (n.eq.2) then
2713             do 250 l = 1, jsl(2,m)
2714             koff = nypmx*(l - 1)
2715             loff = nyzps(1,m)*(l - 1)
2716             do 240 k = 1, nyzps(1,m)
2717             kk = k + koff
2718             ll = k + loff
2719             do 230 j = 1, nxv
2720             g(j,kk,m) = h(j,ll,m)
2721   230       continue
2722   240       continue
2723   250       continue
2724          endif
2725       endif
       260 continue
```

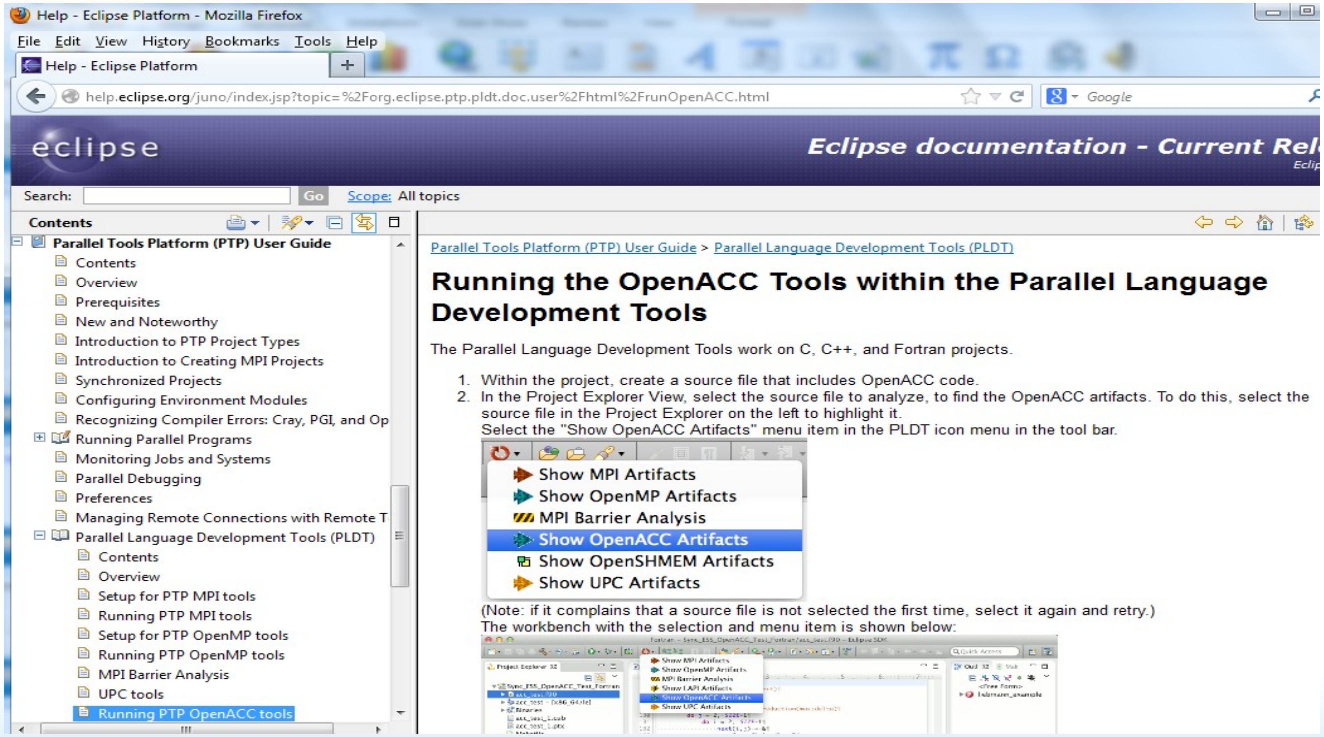
0 errors, 4 warnings, 891 others

Description	Resource	Path	Local
A loop starting at line 2676 was replaced by a library call.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2684 was collapsed into the loop starting at line 2686.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2686 was replaced by a library call.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2704 was collapsed into the loop starting at line 2707.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2707 was replaced by a library call.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2716 was collapsed into the loop starting at line 2719.	magtail_sub.f	/magtail@bw	line 2

A loop starting at line 2707 was replaced by a library call.

OpenACC artifacts

- Search for OpenACC in eclipse help


Help - Eclipse Platform - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Help - Eclipse Platform
help.eclipse.org/juno/index.jsp?topic=%2Forg.eclipse.ptp.pldt.doc.user%2Fhtml%2FrunOpenACC.html
Google
eclipse
Eclipse documentation - Current Rel
Search:
Go
Scope: All topics
Contents
Parallel Tools Platform (PTP) User Guide
Contents
Overview
Prerequisites
New and Noteworthy
Introduction to PTP Project Types
Introduction to Creating MPI Projects
Synchronized Projects
Configuring Environment Modules
Recognizing Compiler Errors: Cray, PGI, and Op
Running Parallel Programs
Monitoring Jobs and Systems
Parallel Debugging
Preferences
Managing Remote Connections with Remote T
Parallel Language Development Tools (PLDT)
Contents
Overview
Setup for PTP MPI tools
Running PTP MPI tools
Setup for PTP OpenMP tools
Running PTP OpenMP tools
MPI Barrier Analysis
UPC tools
Running PTP OpenACC tools

Parallel Tools Platform (PTP) User Guide > Parallel Language Development Tools (PLDT)

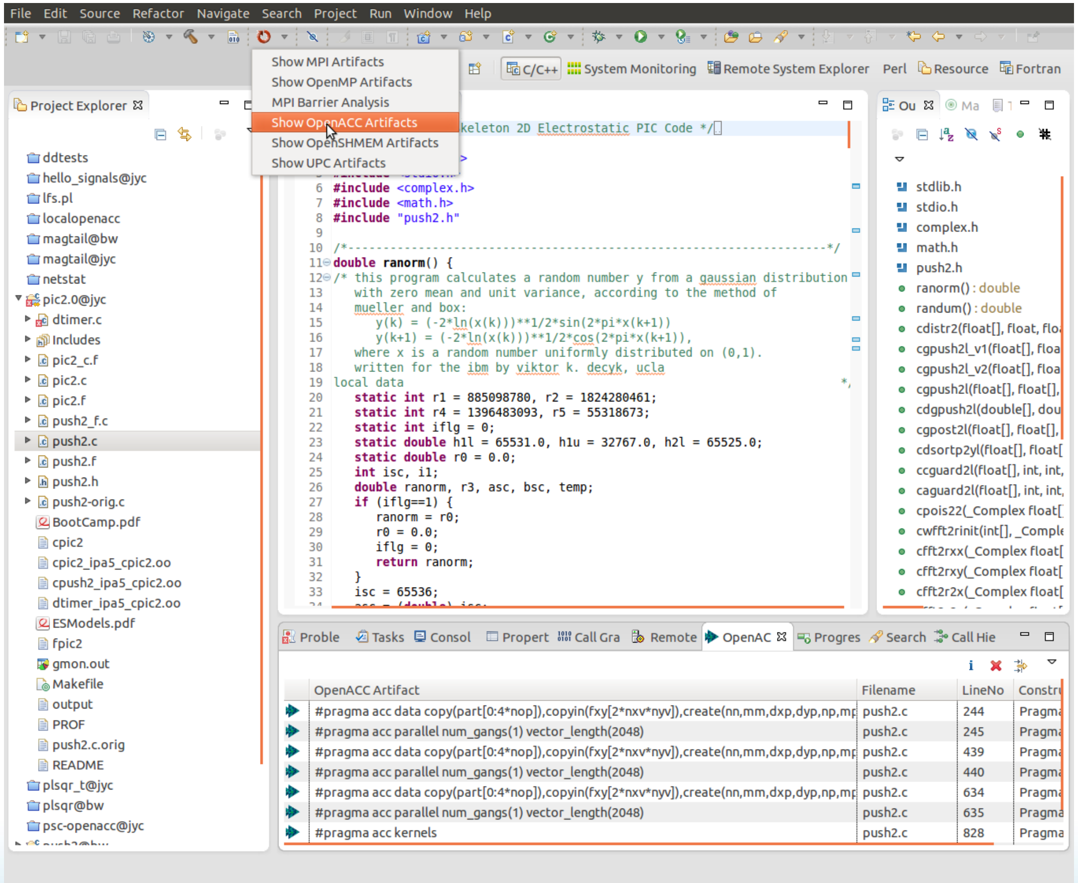
Running the OpenACC Tools within the Parallel Language Development Tools

The Parallel Language Development Tools work on C, C++, and Fortran projects.

1. Within the project, create a source file that includes OpenACC code.
2. In the Project Explorer View, select the source file to analyze, to find the OpenACC artifacts. To do this, select the source file in the Project Explorer on the left to highlight it. Select the "Show OpenACC Artifacts" menu item in the PLDT icon menu in the tool bar.

(Note: if it complains that a source file is not selected the first time, select it again and retry.)
The workbench with the selection and menu item is shown below:

OpenACC c code



```
keleton 2D Electrostatic PIC Code */
>
#include <complex.h>
#include <math.h>
#include "push2.h"

/*--------------------------------------------------------------------*/
double ranorm() {
/* this program calculates a random number y from a gaussian distribution
   with zero mean and unit variance, according to the method of
   mueller and box:
      y(k) = (-2*ln(x(k)))**1/2*sin(2*pi*x(k+1))
      y(k+1) = (-2*ln(x(k)))**1/2*cos(2*pi*x(k+1)),
   where x is a random number uniformly distributed on (0,1).
   written for the ibm by viktor k. decyk, ucla
local data                                                             */
   static int r1 = 885098780, r2 = 1824280461;
   static int r4 = 1396483093, r5 = 55318673;
   static int iflg = 0;
   static double h1l = 65531.0, h1u = 32767.0, h2l = 65525.0;
   static double r0 = 0.0;
   int isc, i1;
   double ranorm, r3, asc, bsc, temp;
   if (iflg==1) {
      ranorm = r0;
      r0 = 0.0;
      iflg = 0;
      return ranorm;
   }
   isc = 65536;
```

	OpenACC Artifact	Filename	LineNo	Constru
▶	#pragma acc data copy(part[0:4*nop]),copyin(fxy[2*nxv*nyv]),create(nn,mm,dxp,dyp,np,mp	push2.c	244	Pragma
▶	#pragma acc parallel num_gangs(1) vector_length(2048)	push2.c	245	Pragma
▶	#pragma acc data copy(part[0:4*nop]),copyin(fxy[2*nxv*nyv]),create(nn,mm,dxp,dyp,np,mp	push2.c	439	Pragma
▶	#pragma acc parallel num_gangs(1) vector_length(2048)	push2.c	440	Pragma
▶	#pragma acc data copy(part[0:4*nop]),copyin(fxy[2*nxv*nyv]),create(nn,mm,dxp,dyp,np,mp	push2.c	634	Pragma
▶	#pragma acc parallel num_gangs(1) vector_length(2048)	push2.c	635	Pragma
▶	#pragma acc kernels	push2.c	828	Pragma

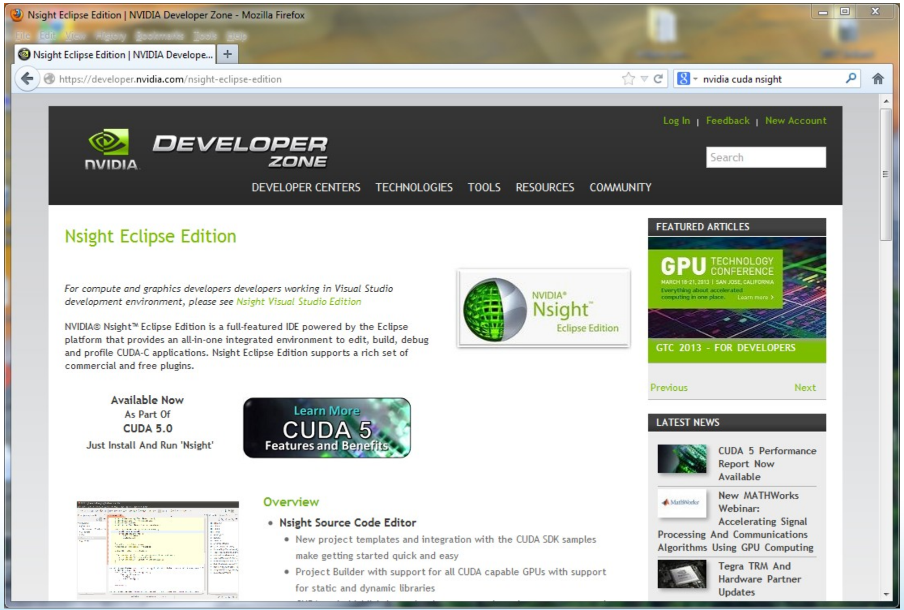
Questions before moving on to Nsight ?

Nvidia Nsight cuda development IDE

- Based on eclipse, but customized for CUDA
  - Handles kernels
  - A slightly simplified version of eclipse
  - Contains no parallel tools components (yet)
- Mac and Linux versions
- Windows Nsight is for visual studio – no demo today

Nsight demo – run remote or install ?

- Demo: Running locally from jyc or bw
  - Module load cudatoolkit ; nsight
- You can install your own version by downloading cuda5
- My linux laptop has a copy from bw obtained using Globus Online


Nsight Eclipse Edition | NVIDIA Developer Zone - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Nsight Eclipse Edition | NVIDIA Develope...
https://developer.nvidia.com/nsight-eclipse-edition
nvidia cuda nsight
Log In | Feedback | New Account
NVIDIA
DEVELOPER ZONE
Search
DEVELOPER CENTERS
TECHNOLOGIES
TOOLS
RESOURCES
COMMUNITY
NVIDIA® Nsight™ Eclipse Edition
FEATURED ARTICLES
GPU TECHNOLOGY CONFERENCE
MARCH 18-21, 2013 | SAN JOSE, CALIFORNIA
Everything about accelerated computing in one place. Learn more ›
GTC 2013 - FOR DEVELOPERS
Previous
Next
LATEST NEWS
Learn More
CUDA 5
Features and Benefits

Nsight Eclipse Edition

For compute and graphics developers developers working in Visual Studio development environment, please see Nsight Visual Studio Edition

NVIDIA® Nsight™ Eclipse Edition is a full-featured IDE powered by the Eclipse platform that provides an all-in-one integrated environment to edit, build, debug and profile CUDA-C applications. Nsight Eclipse Edition supports a rich set of commercial and free plugins.

Available Now
As Part Of
CUDA 5.0
Just Install And Run 'Nsight'

CUDA 5 Performance Report Now Available

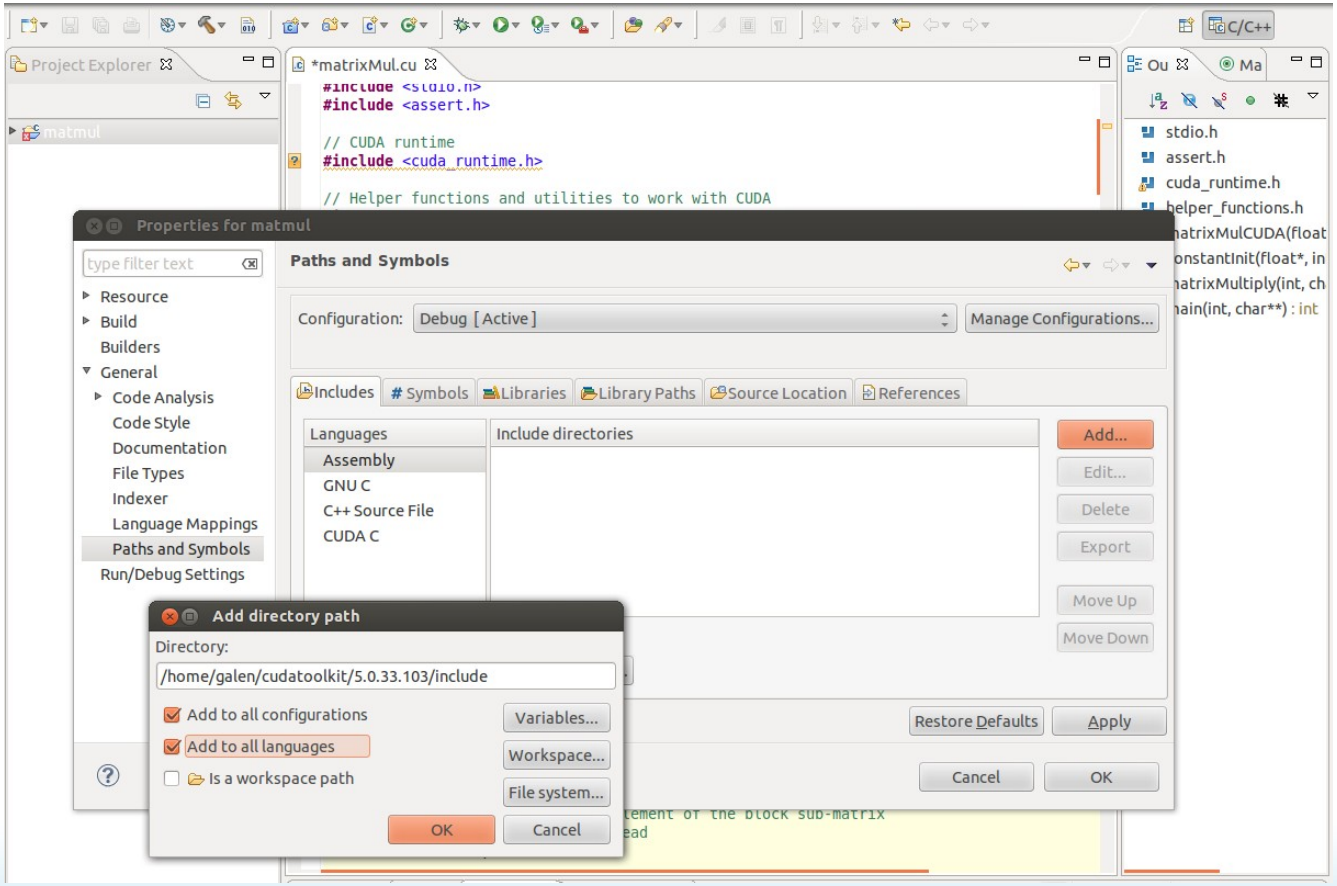
New MATHWorks Webinar: Accelerating Signal Processing And Communications Algorithms Using GPU Computing

Tegra TRM And Hardware Partner Updates

Overview

- Nsight Source Code Editor
  - New project templates and integration with the CUDA SDK samples make getting started quick and easy
  - Project Builder with support for all CUDA capable GPUs with support for static and dynamic libraries

Nsight customizations when running local



```
#include <stdio.h>
#include <assert.h>

// CUDA runtime
#include <cuda_runtime.h>

// Helper functions and utilities to work with CUDA
```

Properties for matmul — Paths and Symbols

Configuration: Debug [Active]

Includes — Languages: Assembly, GNU C, C++ Source File, CUDA C — Include directories

Add directory path

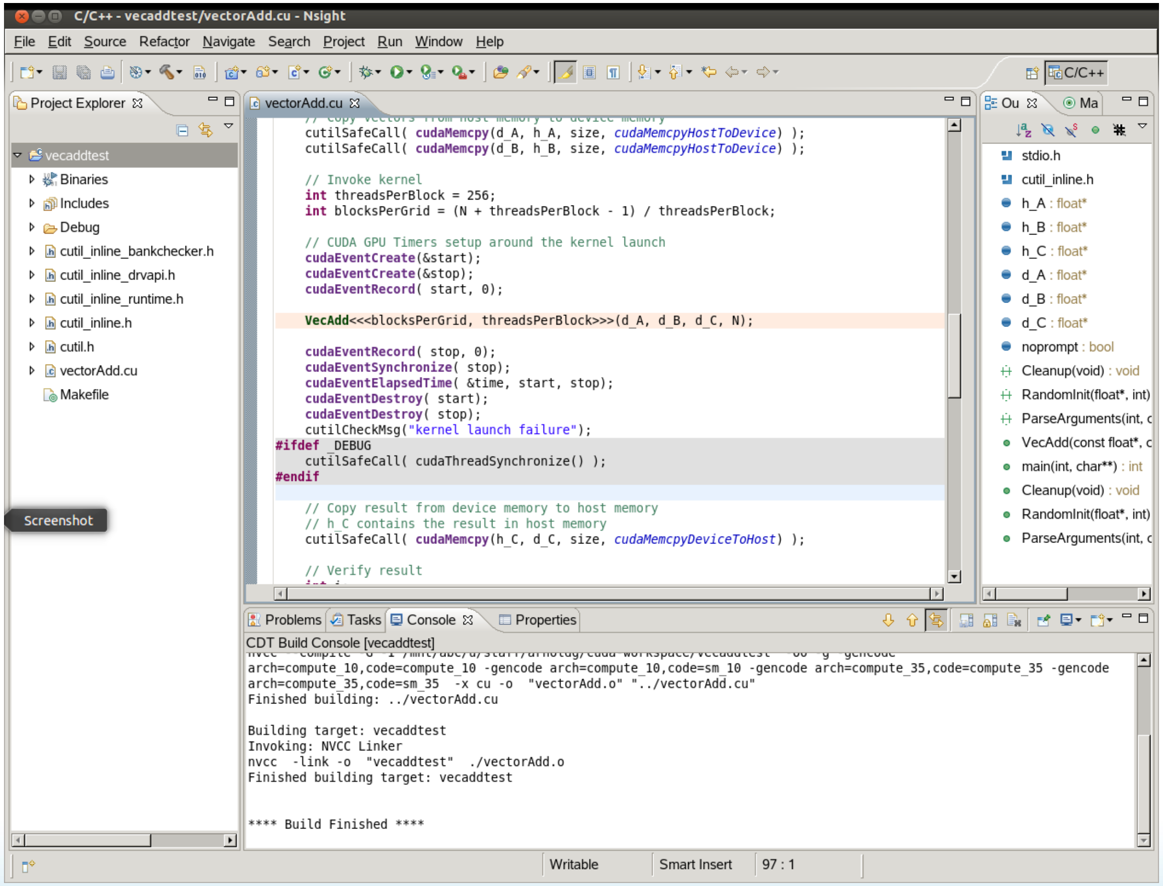
Directory: /home/galen/cudatoolkit/5.0.33.103/include

☑ Add to all configurations

☑ Add to all languages

□ Is a workspace path

Nsight on bw or jyc

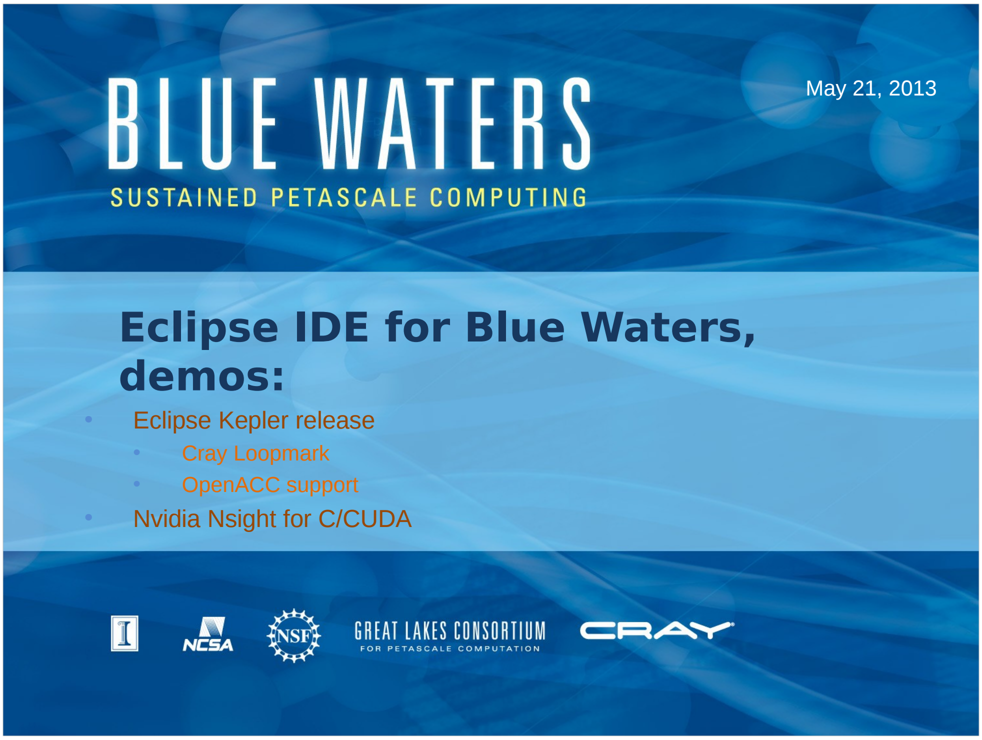


```
    // Copy vectors from host memory to device memory
    cutilSafeCall( cudaMemcpy(d_A, h_A, size, cudaMemcpyHostToDevice) );
    cutilSafeCall( cudaMemcpy(d_B, h_B, size, cudaMemcpyHostToDevice) );

    // Invoke kernel
    int threadsPerBlock = 256;
    int blocksPerGrid = (N + threadsPerBlock - 1) / threadsPerBlock;

    // CUDA GPU Timers setup around the kernel launch
    cudaEventCreate(&start);
    cudaEventCreate(&stop);
    cudaEventRecord( start, 0);

    VecAdd<<<blocksPerGrid, threadsPerBlock>>>(d_A, d_B, d_C, N);

    cudaEventRecord( stop, 0);
    cudaEventSynchronize( stop);
    cudaEventElapsedTime( &time, start, stop);
    cudaEventDestroy( start);
    cudaEventDestroy( stop);
    cutilCheckMsg("kernel launch failure");
#ifdef _DEBUG
    cutilSafeCall( cudaThreadSynchronize() );
#endif

    // Copy result from device memory to host memory
    // h_C contains the result in host memory
    cutilSafeCall( cudaMemcpy(h_C, d_C, size, cudaMemcpyDeviceToHost) );

    // Verify result
```

```
nvcc --compile -G -I /mnt/abc/u/staff/arnoldg/cuda-workspace/vecaddtest -O0 -g -gencode
arch=compute_10,code=compute_10 -gencode arch=compute_10,code=sm_10 -gencode arch=compute_35,code=compute_35 -gencode
arch=compute_35,code=sm_35  -x cu -o  "vectorAdd.o" "../vectorAdd.cu"
Finished building: ../vectorAdd.cu

Building target: vecaddtest
Invoking: NVCC Linker
nvcc  -link -o  "vecaddtest"  ./vectorAdd.o
Finished building target: vecaddtest


**** Build Finished ****
```

Nsight features

- Hover over kernel invocation, bring up definition
- Understands .cu file extension
- Can build code with nvcc
- Cudasamples/ (from Nvidia) contains ready-to-build Nsight projects of most of the sample codes used in Nvidia documentation and tutorials



BLUE WATERS

SUSTAINED PETASCALE COMPUTING

May 21, 2013

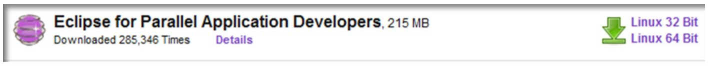
Eclipse IDE for Blue Waters, demos:

- Eclipse Kepler release
  - Cray Loopmark
  - OpenACC support
- Nvidia Nsight for C/CUDA

I
NCSA
NSF
GREAT LAKES CONSORTIUM
FOR PETASCALE COMPUTATION
CRAY

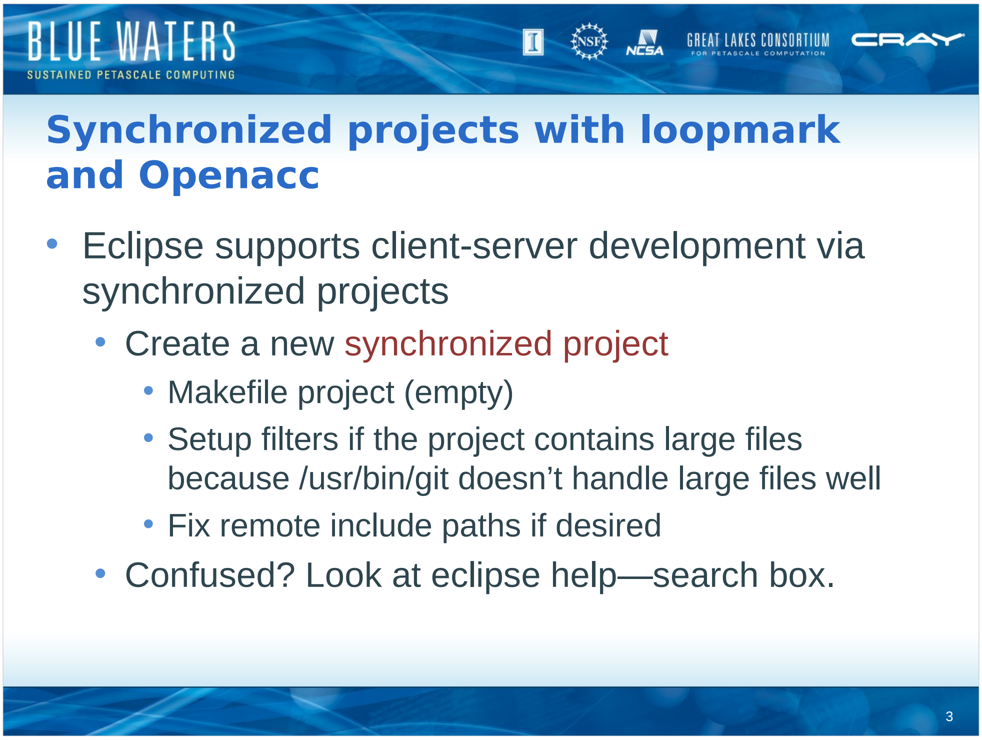
Eclipse kepler release

- Eclipse downloads are available for Linux, Mac and Windows platforms
- www.eclipse.org
  - Downloads
    - Developer builds
      - Eclipse for Parallel Application Developers
        - Select your OS and architecture (32 or 64 bit)



Eclipse for Parallel Application Developers, 215 MB
Downloaded 285,346 Times Details

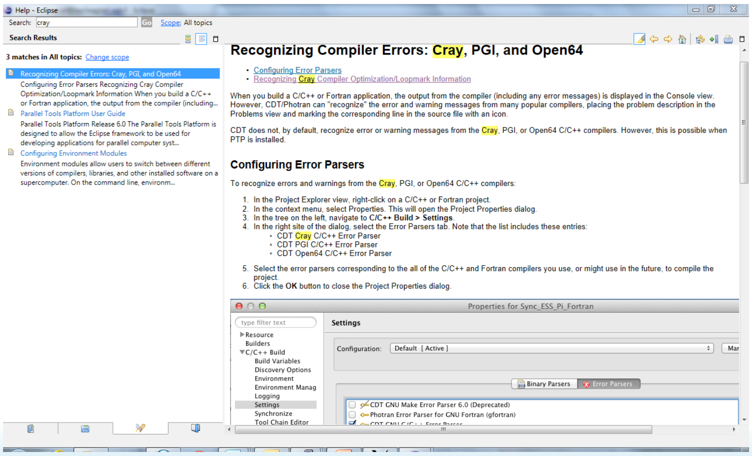
Linux 32 Bit
Linux 64 Bit



Synchronized projects with loopmark and Openacc

- Eclipse supports client-server development via synchronized projects
  - Create a new synchronized project
    - Makefile project (empty)
    - Setup filters if the project contains large files because /usr/bin/git doesn’t handle large files well
    - Fix remote include paths if desired
  - Confused? Look at eclipse help—search box.

Eclipse help: searching for Cray



Help - Eclipse

Search: cray Go Scope: All topics

Search Results

3 matches in All topics: Change scope

- Recognizing Compiler Errors: Cray, PGI, and Open64
  Configuring Error Parsers Recognizing Cray Compiler Optimization/Loopmark Information When you build a C/C++ or Fortran application, the output from the compiler (including...
- Parallel Tools Platform User Guide
  Parallel Tools Platform Release 6.0 The Parallel Tools Platform is designed to allow the Eclipse framework to be used for developing applications for parallel computer syst...
- Configuring Environment Modules
  Environment modules allow users to switch between different versions of compilers, libraries, and other installed software on a supercomputer. On the command line, environm...

Recognizing Compiler Errors: Cray, PGI, and Open64

- Configuring Error Parsers
- Recognizing Cray Compiler Optimization/Loopmark Information

When you build a C/C++ or Fortran application, the output from the compiler (including any error messages) is displayed in the Console view. However, CDT/Photran can "recognize" the error and warning messages from many popular compilers, placing the problem description in the Problems view and marking the corresponding line in the source file with an icon.

CDT does not, by default, recognize error or warning messages from the Cray, PGI, or Open64 C/C++ compilers. However, this is possible when PTP is installed.

Configuring Error Parsers

To recognize errors and warnings from the Cray, PGI, or Open64 C/C++ compilers:

1. In the Project Explorer view, right-click on a C/C++ or Fortran project.
2. In the context menu, select Properties. This will open the Project Properties dialog.
3. In the tree on the left, navigate to **C/C++ Build > Settings**.
4. In the right site of the dialog, select the Error Parsers tab. Note that the list includes these entries:
   - CDT Cray C/C++ Error Parser
   - CDT PGI C/C++ Error Parser
   - CDT Open64 C/C++ Error Parser
5. Select the error parsers corresponding to the all of the C/C++ and Fortran compilers you use, or might use in the future, to compile the project.
6. Click the **OK** button to close the Project Properties dialog.

Properties for Sync_ESS_Pi_Fortran

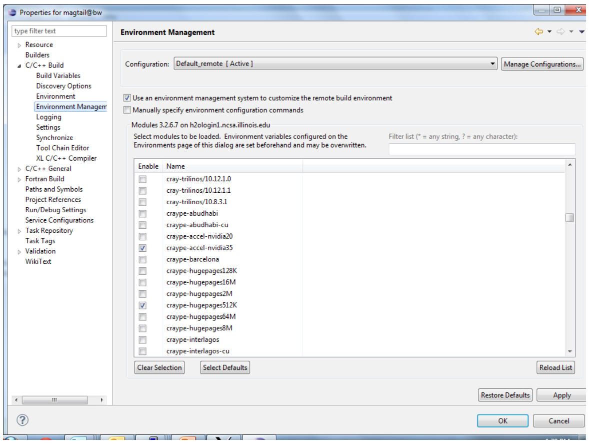
Settings

Configuration: Default [Active]

☐ CDT GNU Make Error Parser 6.0 (Deprecated)
☐ Photran Error Parser for GNU Fortran (gfortran)
☑ CDT GNU C/C++ Error Parser

Working with modules (Blue Waters and Xsede)

- Project
  - Properties
    - c/c++ build (also for fortran)
      - Environment management



Environment Management

Configuration: Default_remote [Active]

☑ Use an environment management system to customize the remote build environment

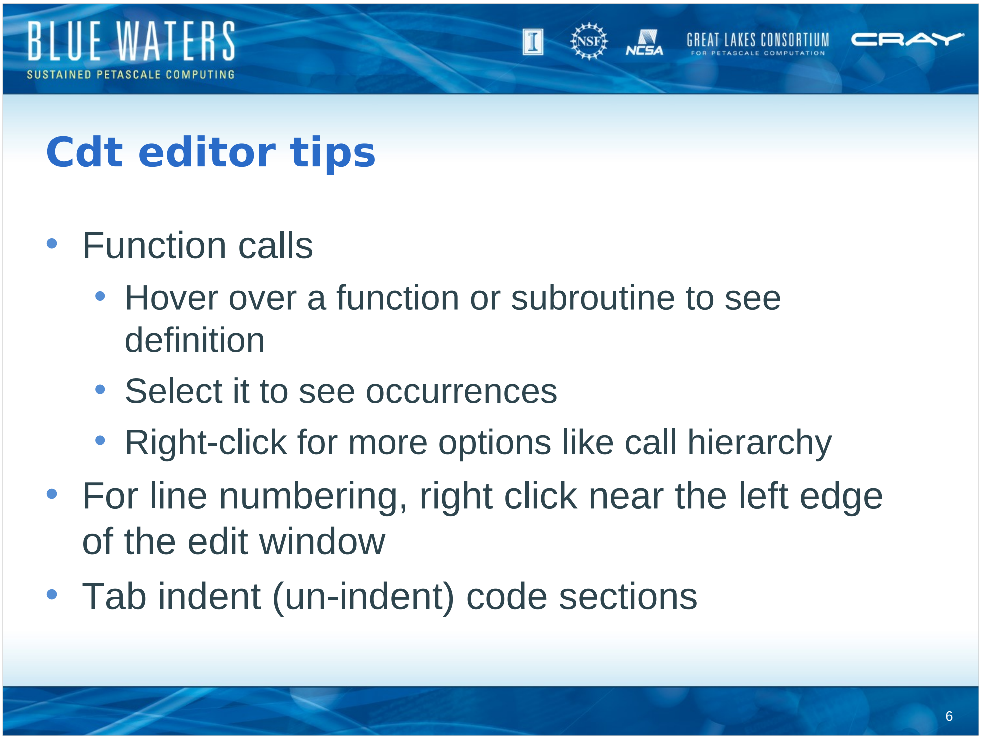
□ Manually specify environment configuration commands

Modules 3.2.6.7 on h2ologin1.ncsa.illinois.edu

Select modules to be loaded. Environment variables configured on the Environments page of this dialog are set beforehand and may be overwritten.

Filter list (* = any string, ? = any character):

Enable	Name
□	cray-trilinos/10.12.1.0
□	cray-trilinos/10.12.1.1
□	cray-trilinos/10.8.3.1
□	craype-abudhabi
□	craype-abudhabi-cu
□	craype-accel-nvidia20
☑	craype-accel-nvidia35
□	craype-barcelona
□	craype-hugepages128K
□	craype-hugepages16M
□	craype-hugepages2M
☑	craype-hugepages512K
□	craype-hugepages64M
□	craype-hugepages8M
□	craype-interlagos
□	craype-interlagos-cu



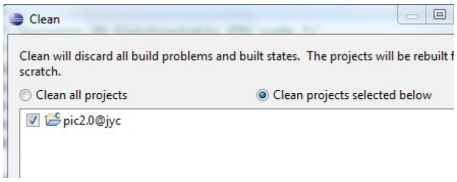
BLUE WATERS
SUSTAINED PETASCALE COMPUTING
NSF
NCSA
GREAT LAKES CONSORTIUM
FOR PETASCALE COMPUTATION
CRAY

Cdt editor tips

- Function calls
  - Hover over a function or subroutine to see definition
  - Select it to see occurrences
  - Right-click for more options like call hierarchy
- For line numbering, right click near the left edge of the edit window
- Tab indent (un-indent) code sections

Driving makefile: clean and build

- Project menu
  - Clean
  - Build

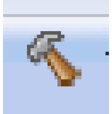


Clean will discard all build problems and built states. The projects will be rebuilt f
scratch.

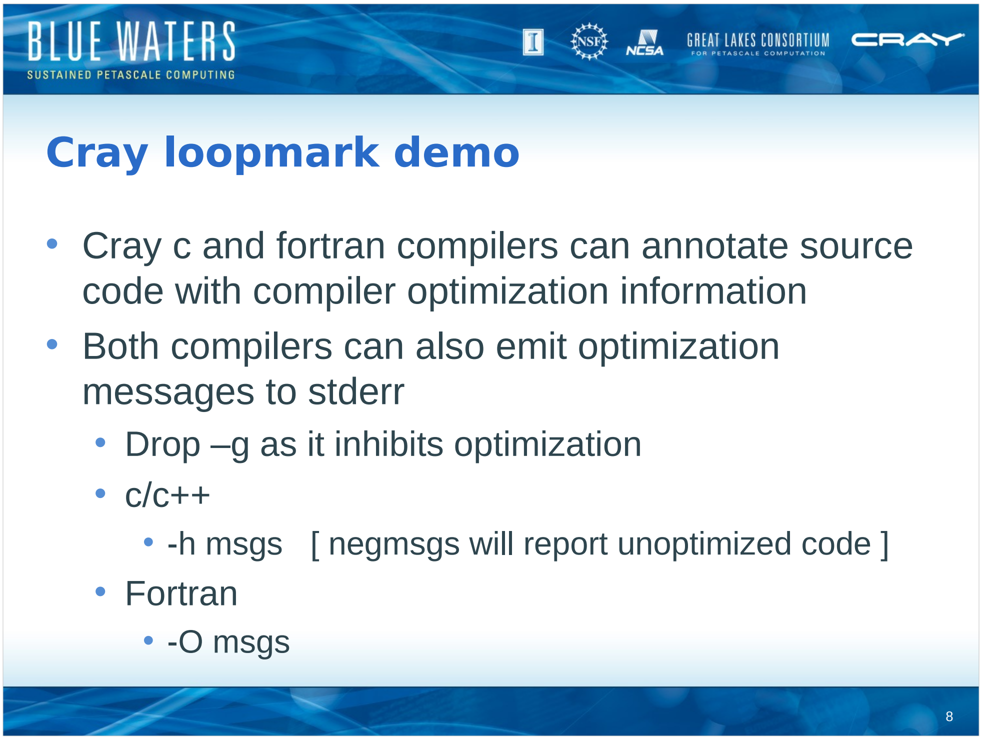
○ Clean all projects ● Clean projects selected below

☑ pic2.0@jyc

- Hammer time (build)



- There are often multiple methods of doing the same thing in eclipse: it’s the unix of IDEs

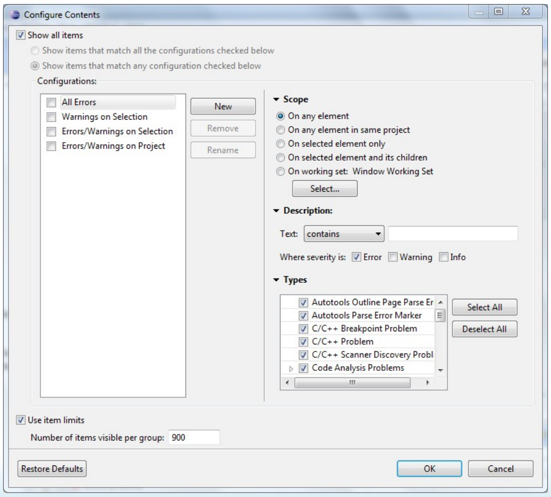


Cray loopmark demo

- Cray c and fortran compilers can annotate source code with compiler optimization information
- Both compilers can also emit optimization messages to stderr
  - Drop –g as it inhibits optimization
  - c/c++
    - -h msgs [negmsgs will report unoptimized code]
  - Fortran
    - -O msgs

Optimization report info

- Problems view
  - Info
    - Defaults to 100 items
    - view menu -> configure contents (to increase)



Configure Contents

☑ Show all items

○ Show items that match all the configurations checked below

● Show items that match any configuration checked below

Configurations:

□ All Errors
□ Warnings on Selection
□ Errors/Warnings on Selection
□ Errors/Warnings on Project

Scope

● On any element
○ On any element in same project
○ On selected element only
○ On selected element and its children
○ On working set: Window Working Set

Description:

Text: contains

Where severity is: ☑ Error □ Warning □ Info

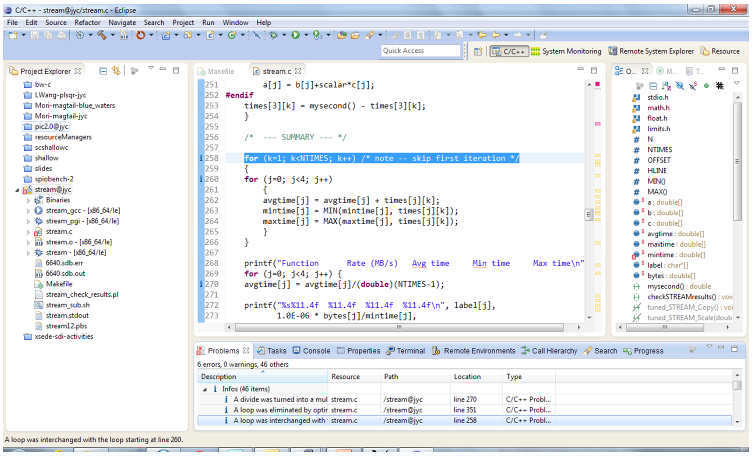
Types

☑ Autotools Outline Page Parse Er
☑ Autotools Parse Error Marker
☑ C/C++ Breakpoint Problem
☑ C/C++ Problem
☑ C/C++ Scanner Discovery Probl
☑ Code Analysis Problems

☑ Use item limits

Number of items visible per group: 900

C optimization report view



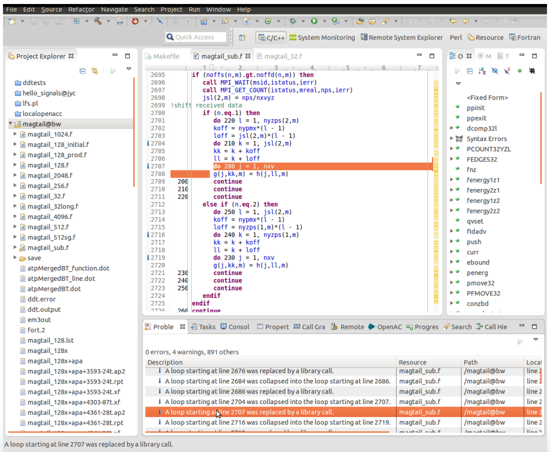
```
251         a[j] = b[j]+scalar*c[j];
252 #endif
253     times[3][k] = mysecond() - times[3][k];
254     }
255 
256     /*  --- SUMMARY --- */
257 
258     for (k=1; k<NTIMES; k++) /* note -- skip first iteration */
259     {
260     for (j=0; j<4; j++)
261         {
262         avgtime[j] = avgtime[j] + times[j][k];
263         mintime[j] = MIN(mintime[j], times[j][k]);
264         maxtime[j] = MAX(maxtime[j], times[j][k]);
265         }
266     }
267 
268     printf("Function      Rate (MB/s)   Avg time     Min time     Max time\n"
269     for (j=0; j<4; j++) {
270     avgtime[j] = avgtime[j]/(double)(NTIMES-1);
271 
272     printf("%s%11.4f  %11.4f  %11.4f  %11.4f\n", label[j],
273            1.0E-06 * bytes[j]/mintime[j],
```

6 errors, 0 warnings, 46 others

Description	Resource	Path	Location	Type
Infos (46 items)				
A divide was turned into a mul	stream.c	/stream@jyc	line 270	C/C++ Probl...
A loop was eliminated by optir	stream.c	/stream@jyc	line 351	C/C++ Probl...
A loop was interchanged with	stream.c	/stream@jyc	line 258	C/C++ Probl...

A loop was interchanged with the loop starting at line 260.

Fortran optimization report view



```
2695         if (noffs(n,m).gt.noffd(n,m)) then
2696            call MPI_WAIT(msid,istatus,ierr)
2697            call MPI_GET_COUNT(istatus,mreal,nps,ierr)
2698            jsl(2,m) = nps/nxvyz
2699 !shift received data
2700            if (n.eq.1) then
2701               do 220 l = 1, nyzps(2,m)
2702               koff = nypmx*(l - 1)
2703               loff = jsl(2,m)*(l - 1)
2704               do 210 k = 1, jsl(2,m)
2705               kk = k + koff
2706               ll = k + loff
2707               do 200 j = 1, nxv
2708               g(j,kk,m) = h(j,ll,m)
2709   200         continue
2710   210         continue
2711   220         continue
2712            else if (n.eq.2) then
2713               do 250 l = 1, jsl(2,m)
2714               koff = nypmx*(l - 1)
2715               loff = nyzps(1,m)*(l - 1)
2716               do 240 k = 1, nyzps(1,m)
2717               kk = k + koff
2718               ll = k + loff
2719               do 230 j = 1, nxv
2720               g(j,kk,m) = h(j,ll,m)
2721   230         continue
2722   240         continue
2723   250         continue
2724            endif
2725         endif
2726   260   continue
```

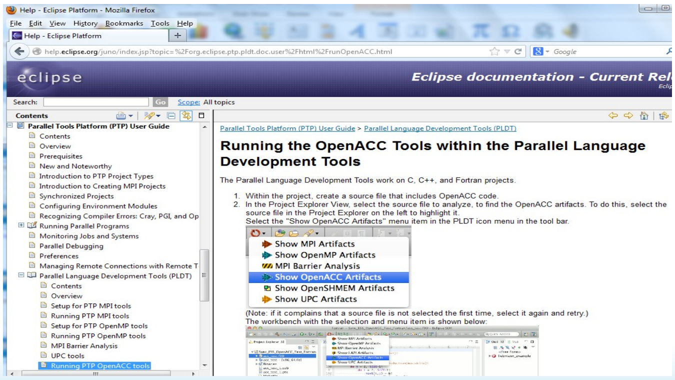
0 errors, 4 warnings, 891 others

Description	Resource	Path	Locat
A loop starting at line 2676 was replaced by a library call.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2684 was collapsed into the loop starting at line 2686.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2686 was replaced by a library call.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2704 was collapsed into the loop starting at line 2707.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2707 was replaced by a library call.	magtail_sub.f	/magtail@bw	line 2
A loop starting at line 2716 was collapsed into the loop starting at line 2719.	magtail_sub.f	/magtail@bw	line 2

A loop starting at line 2707 was replaced by a library call.

OpenACC artifacts

- Search for OpenACC in eclipse help


Help - Eclipse Platform - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Help - Eclipse Platform
help.eclipse.org/juno/index.jsp?topic=%2Forg.eclipse.ptp.pldt.doc.user%2Fhtml%2FrunOpenACC.html
Google
eclipse
Eclipse documentation - Current Rel
Search: Go Scope: All topics
Contents
Parallel Tools Platform (PTP) User Guide
Contents
Overview
Prerequisites
New and Noteworthy
Introduction to PTP Project Types
Introduction to Creating MPI Projects
Synchronized Projects
Configuring Environment Modules
Recognizing Compiler Errors: Cray, PGI, and Op
Running Parallel Programs
Monitoring Jobs and Systems
Parallel Debugging
Preferences
Managing Remote Connections with Remote T
Parallel Language Development Tools (PLDT)
Contents
Overview
Setup for PTP MPI tools
Running PTP MPI tools
Setup for PTP OpenMP tools
Running PTP OpenMP tools
MPI Barrier Analysis
UPC tools
Running PTP OpenACC tools
Parallel Tools Platform (PTP) User Guide > Parallel Language Development Tools (PLDT)

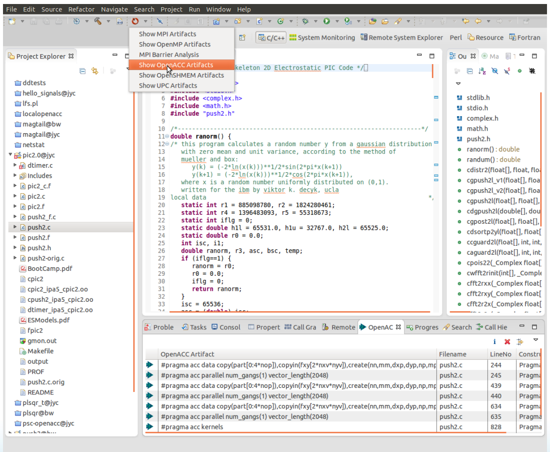
Running the OpenACC Tools within the Parallel Language Development Tools

The Parallel Language Development Tools work on C, C++, and Fortran projects.

1. Within the project, create a source file that includes OpenACC code.
2. In the Project Explorer View, select the source file to analyze, to find the OpenACC artifacts. To do this, select the source file in the Project Explorer on the left to highlight it.
   Select the "Show OpenACC Artifacts" menu item in the PLDT icon menu in the tool bar.

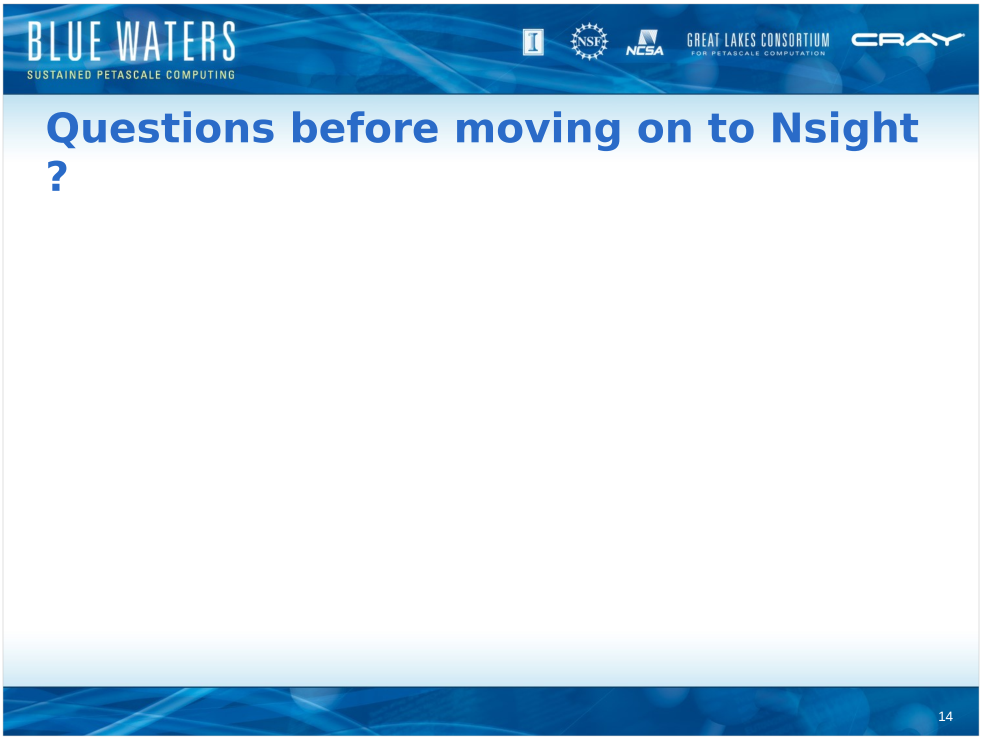
(Note: if it complains that a source file is not selected the first time, select it again and retry.)
The workbench with the selection and menu item is shown below:

OpenACC c code

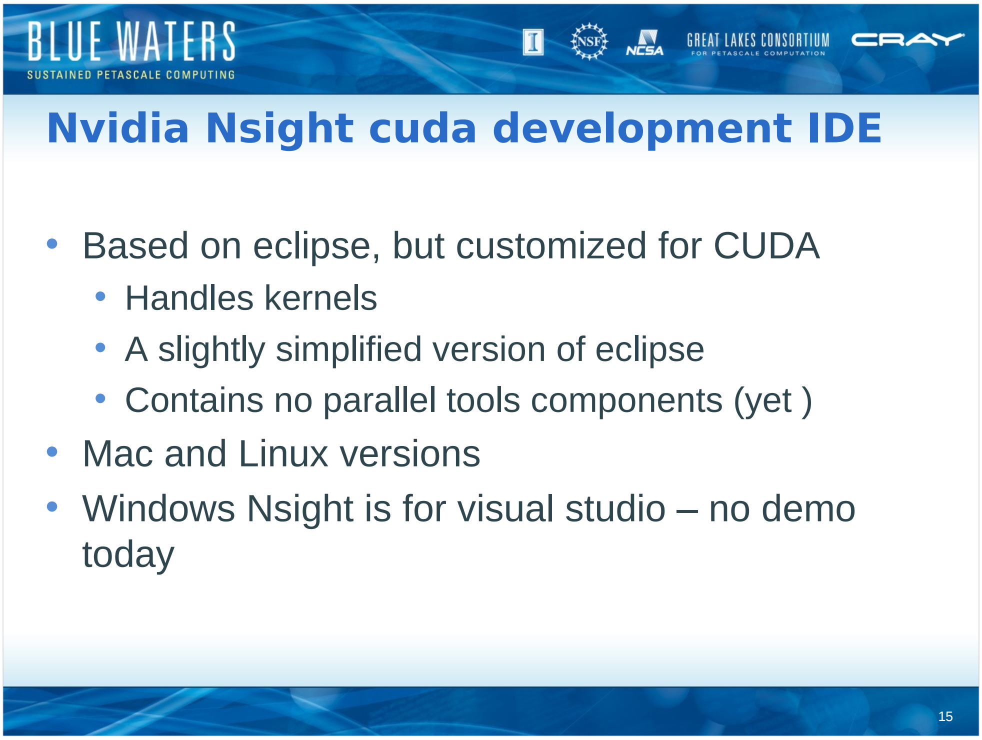


```
keleton 2D Electrostatic PIC Code */
#include <complex.h>
#include <math.h>
#include "push2.h"

/*-----------------------------------------------------------------------*/
double ranorm() {
/* this program calculates a random number y from a gaussian distribution
   with zero mean and unit variance, according to the method of
   mueller and box:
      y(k) = (-2*ln(x(k)))**1/2*sin(2*pi*x(k+1))
      y(k+1) = (-2*ln(x(k)))**1/2*cos(2*pi*x(k+1)),
   where x is a random number uniformly distributed on (0,1).
   written for the ibm by viktor k. decyk, ucla
local data                                                              *,
   static int r1 = 885098780, r2 = 1824280461;
   static int r4 = 1396483093, r5 = 55318673;
   static int iflg = 0;
   static double h1l = 65531.0, h1u = 32767.0, h2l = 65525.0;
   static double r0 = 0.0;
   int isc, i1;
   double ranorm, r3, asc, bsc, temp;
   if (iflg==1) {
      ranorm = r0;
      r0 = 0.0;
      iflg = 0;
      return ranorm;
   }
   isc = 65536;
```

OpenACC Artifact	Filename	LineNo	Constru
#pragma acc data copy(part[0:4*nop]),copyin(fxy[2*nxv*nyv]),create(nn,mm,dxp,dyp,np,mp	push2.c	244	Pragma
#pragma acc parallel num_gangs(1) vector_length(2048)	push2.c	245	Pragma
#pragma acc data copy(part[0:4*nop]),copyin(fxy[2*nxv*nyv]),create(nn,mm,dxp,dyp,np,mp	push2.c	439	Pragma
#pragma acc parallel num_gangs(1) vector_length(2048)	push2.c	440	Pragma
#pragma acc data copy(part[0:4*nop]),copyin(fxy[2*nxv*nyv]),create(nn,mm,dxp,dyp,np,mp	push2.c	634	Pragma
#pragma acc parallel num_gangs(1) vector_length(2048)	push2.c	635	Pragma
#pragma acc kernels	push2.c	828	Pragma



Questions before moving on to Nsight ?



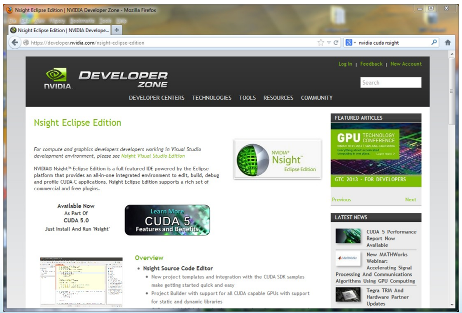
BLUE WATERS
SUSTAINED PETASCALE COMPUTING
NSF
NCSA
GREAT LAKES CONSORTIUM
FOR PETASCALE COMPUTATION
CRAY

Nvidia Nsight cuda development IDE

- Based on eclipse, but customized for CUDA
  - Handles kernels
  - A slightly simplified version of eclipse
  - Contains no parallel tools components (yet)
- Mac and Linux versions
- Windows Nsight is for visual studio – no demo today

Nsight demo – run remote or install ?

- Demo: Running locally from jyc or bw
  - Module load cudatoolkit ; nsight
- You can install your own version by downloading cuda5
- My linux laptop has a copy from bw obtained using Globus Online



Nsight Eclipse Edition | NVIDIA Developer Zone - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Nsight Eclipse Edition | NVIDIA Develope...
https://developer.nvidia.com/nsight-eclipse-edition
nvidia cuda nsight
Log In | Feedback | New Account
NVIDIA DEVELOPER ZONE
Search
DEVELOPER CENTERS
TECHNOLOGIES
TOOLS
RESOURCES
COMMUNITY
NVIDIA® Nsight™ Eclipse Edition
Learn More CUDA 5 Features and Benefits
GPU TECHNOLOGY CONFERENCE
MARCH 18-21, 2013 | SAN JOSE, CALIFORNIA
Everything about accelerated computing in one place. Learn more ›
Previous
Next

Nsight Eclipse Edition

For compute and graphics developers developers working in Visual Studio development environment, please see Nsight Visual Studio Edition

NVIDIA® Nsight™ Eclipse Edition is a full-featured IDE powered by the Eclipse platform that provides an all-in-one integrated environment to edit, build, debug and profile CUDA-C applications. Nsight Eclipse Edition supports a rich set of commercial and free plugins.

Available Now
As Part Of
CUDA 5.0
Just Install And Run 'Nsight'

Overview

- Nsight Source Code Editor
  - New project templates and integration with the CUDA SDK samples make getting started quick and easy
  - Project Builder with support for all CUDA capable GPUs with support for static and dynamic libraries

FEATURED ARTICLES

GTC 2013 - FOR DEVELOPERS

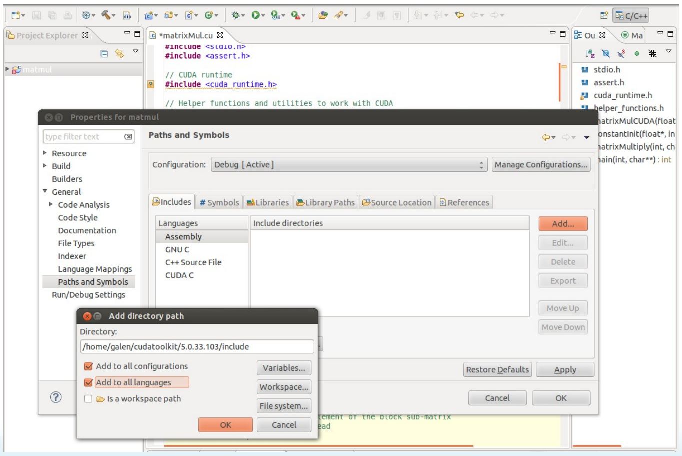
LATEST NEWS

CUDA 5 Performance Report Now Available

New MATHWorks Webinar: Accelerating Signal Processing And Communications Algorithms Using GPU Computing

Tegra TRM And Hardware Partner Updates

Nsight customizations when running local



```
#include <stdio.h>
#include <assert.h>

// CUDA runtime
#include <cuda_runtime.h>

// Helper functions and utilities to work with CUDA
```

Properties for matmul — Paths and Symbols

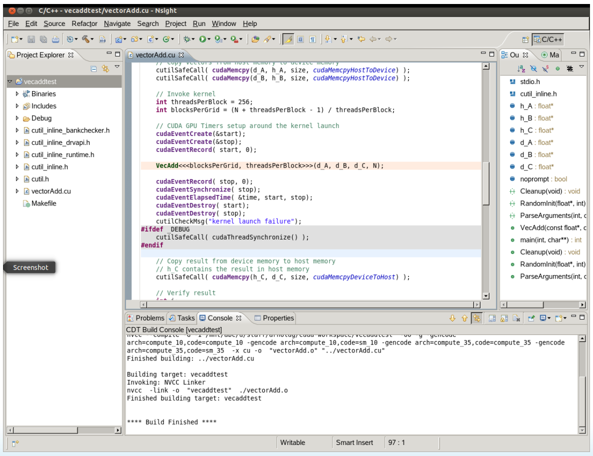
Configuration: Debug [Active]

Add directory path

Directory: /home/galen/cudatoolkit/5.0.33.103/include

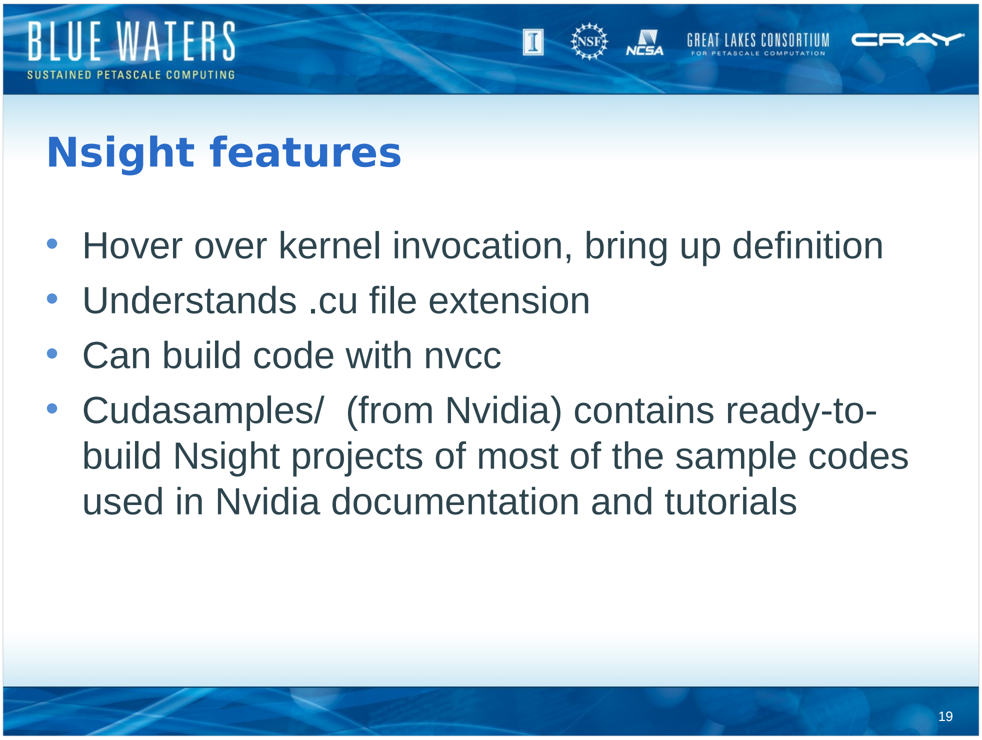
☑ Add to all configurations
☑ Add to all languages
☐ Is a workspace path

Nsight on bw or jyc



```
    cutilSafeCall( cudaMemcpy(d_A, h_A, size, cudaMemcpyHostToDevice) );
    cutilSafeCall( cudaMemcpy(d_B, h_B, size, cudaMemcpyHostToDevice) );

    // Invoke kernel
    int threadsPerBlock = 256;
    int blocksPerGrid = (N + threadsPerBlock - 1) / threadsPerBlock;

    // CUDA GPU Timers setup around the kernel launch
    cudaEventCreate(&start);
    cudaEventCreate(&stop);
    cudaEventRecord( start, 0);

    VecAdd<<<blocksPerGrid, threadsPerBlock>>>(d_A, d_B, d_C, N);

    cudaEventRecord( stop, 0);
    cudaEventSynchronize( stop);
    cudaEventElapsedTime( &time, start, stop);
    cudaEventDestroy( start);
    cudaEventDestroy( stop);
    cutilCheckMsg("kernel launch failure");
#ifdef _DEBUG
    cutilSafeCall( cudaThreadSynchronize() );
#endif

    // Copy result from device memory to host memory
    // h_C contains the result in host memory
    cutilSafeCall( cudaMemcpy(h_C, d_C, size, cudaMemcpyDeviceToHost) );

    // Verify result
```

```
arch=compute_10,code=compute_10 -gencode arch=compute_10,code=sm_10 -gencode arch=compute_35,code=compute_35 -gencode
arch=compute_35,code=sm_35  -x cu -o  "vectorAdd.o" "../vectorAdd.cu"
Finished building: ../vectorAdd.cu

Building target: vecaddtest
Invoking: NVCC Linker
nvcc  -link -o  "vecaddtest"  ./vectorAdd.o
Finished building target: vecaddtest


**** Build Finished ****
```



BLUE WATERS
SUSTAINED PETASCALE COMPUTING

GREAT LAKES CONSORTIUM FOR PETASCALE COMPUTATION

Nsight features

- Hover over kernel invocation, bring up definition
- Understands .cu file extension
- Can build code with nvcc
- Cudasamples/ (from Nvidia) contains ready-to-build Nsight projects of most of the sample codes used in Nvidia documentation and tutorials