



High performance tools to debug, profile, and analyze your applications

High-productivity development tools for science

Beau Paisley

bpaisley@allinea.com

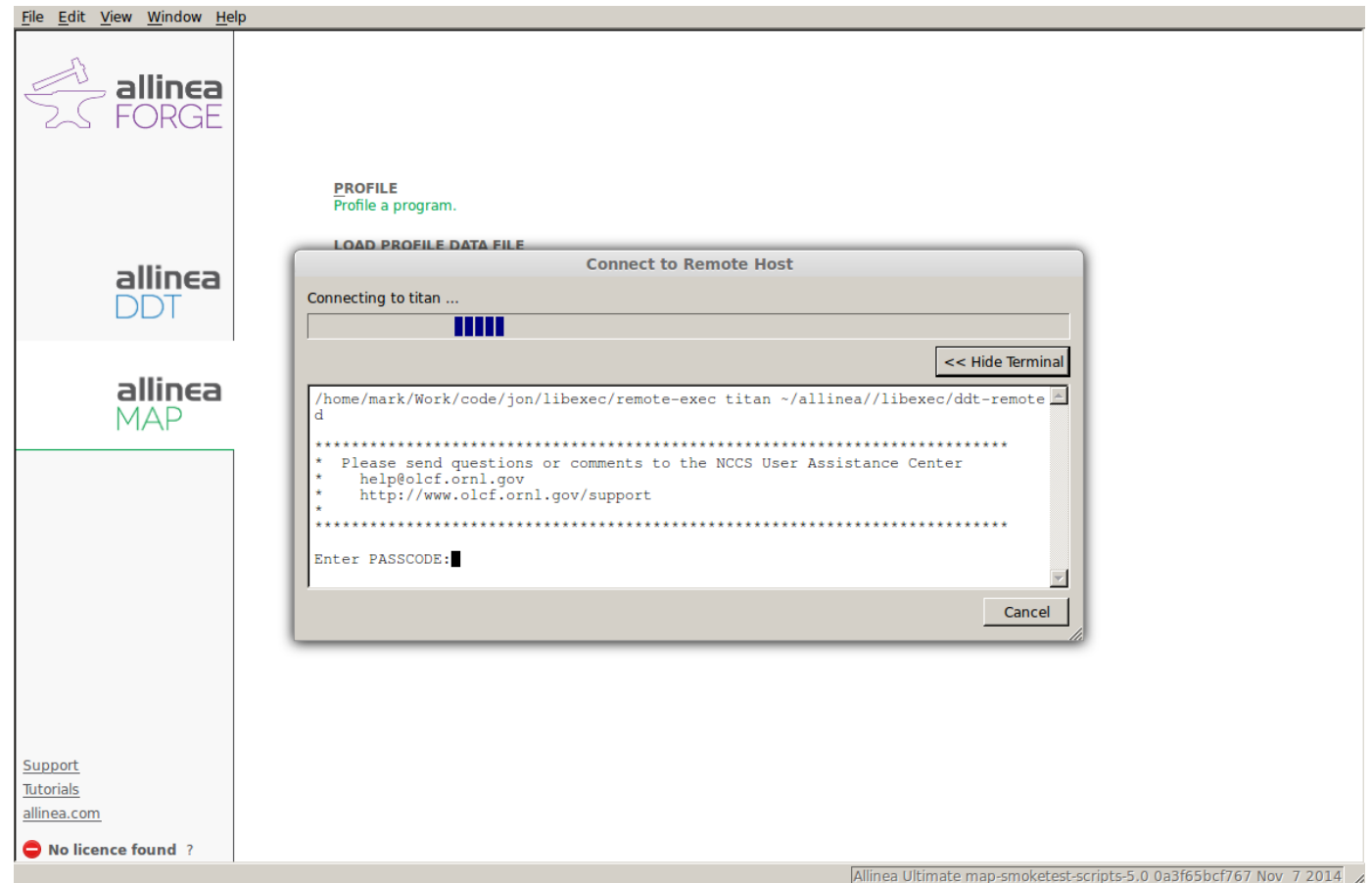


Industry Standard Tools

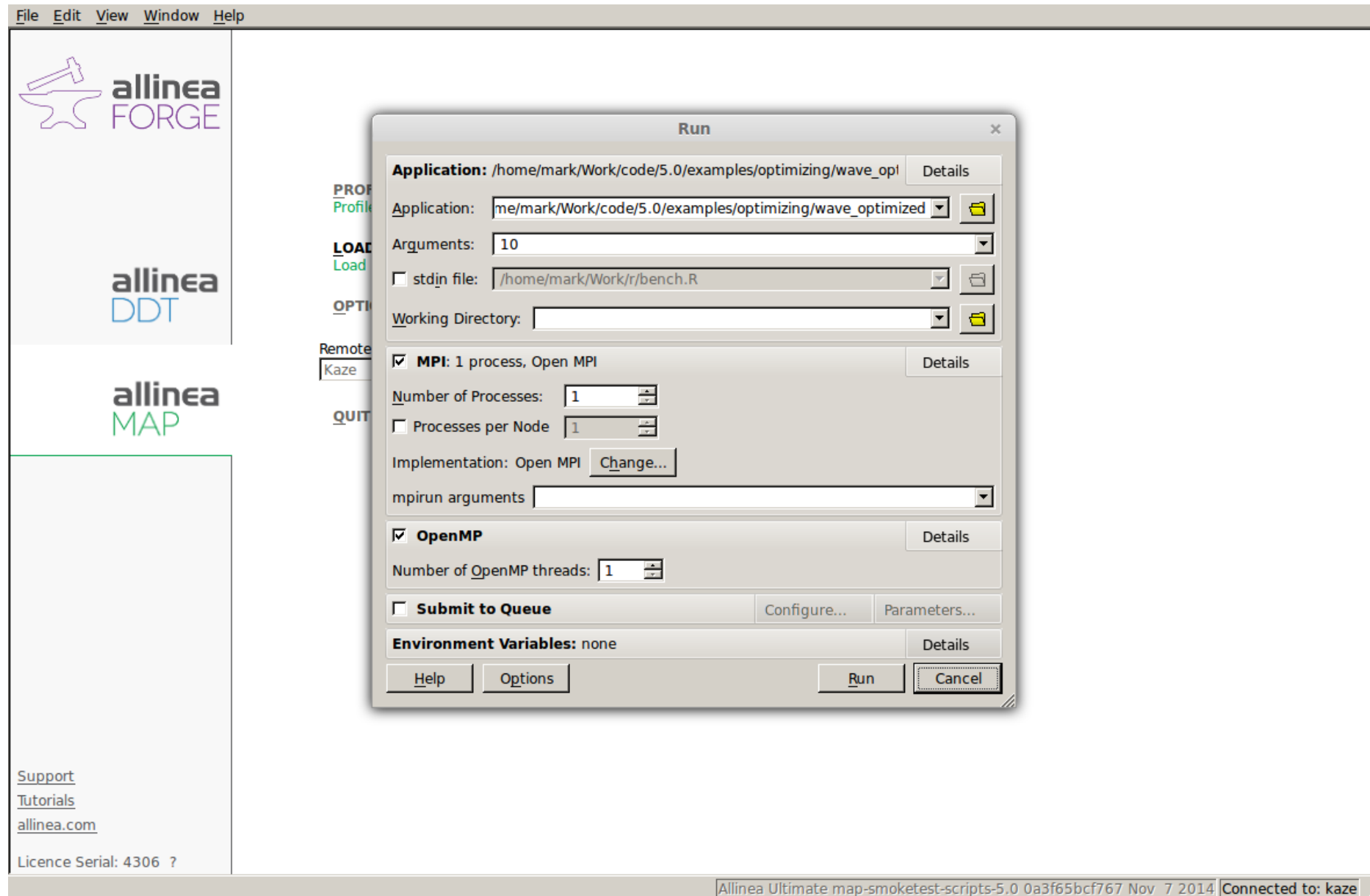


HPC means being able to work productively on remote machines

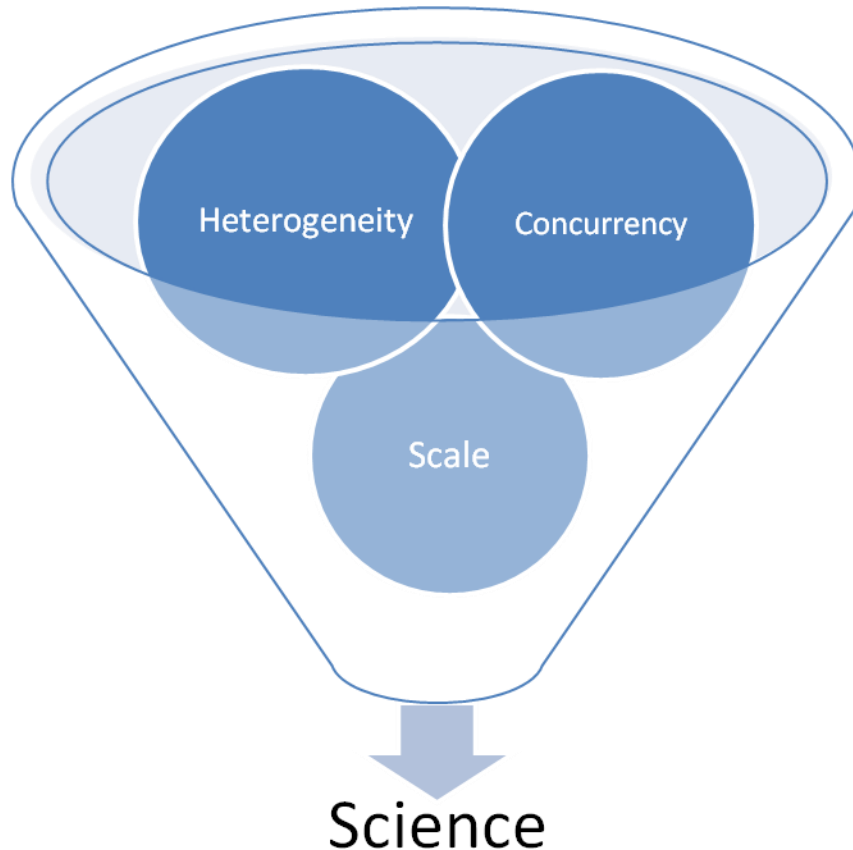
-  Linux
-  OS/X
-  Windows
-  Multiple hop SSH
-  RSA + Cryptocard
-  Uses server license



Submit to job queues or run interactively on any system



Today's Challenge



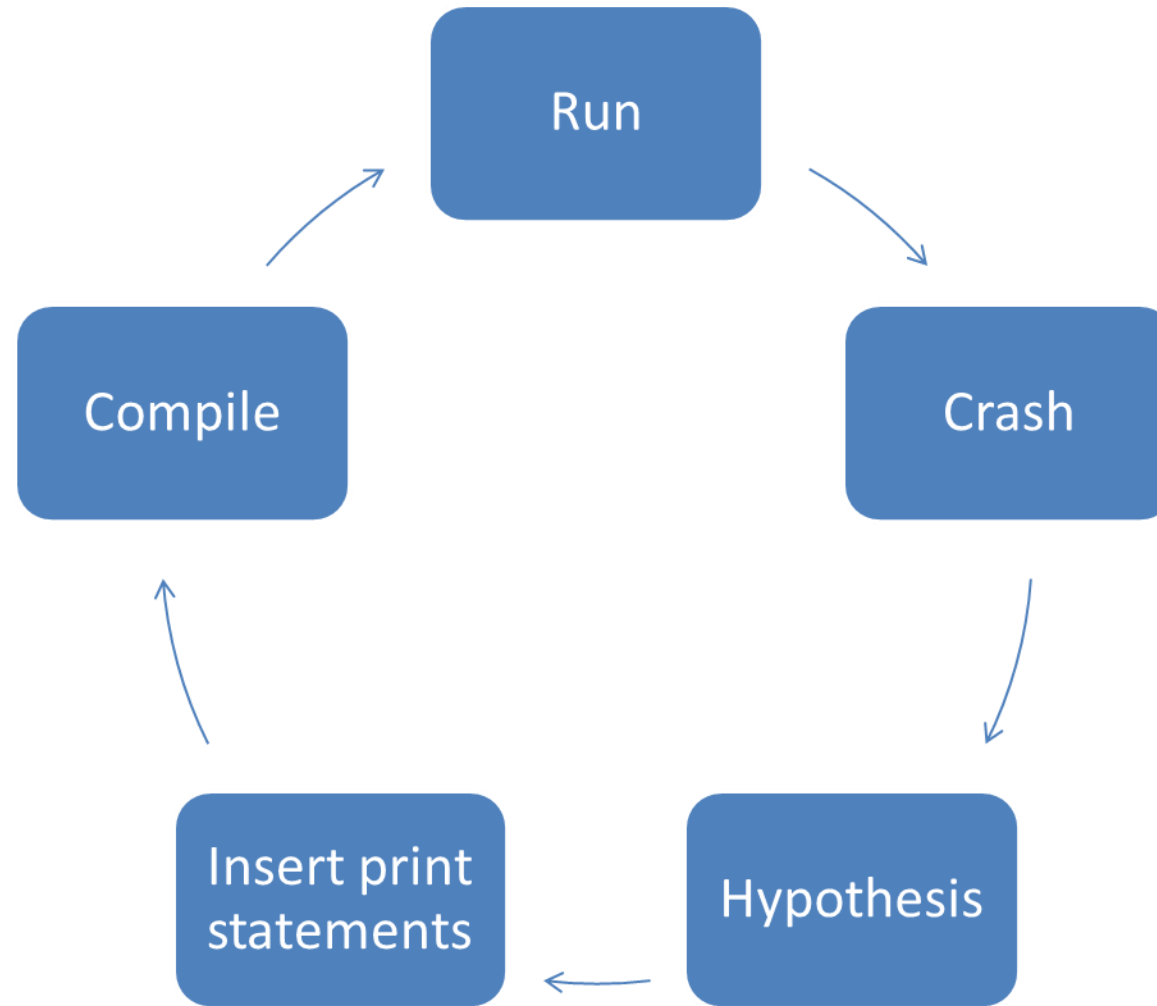
Q: What is the impact of current trends in HPC on your application?

Q: How can you make your science run well on the available system?

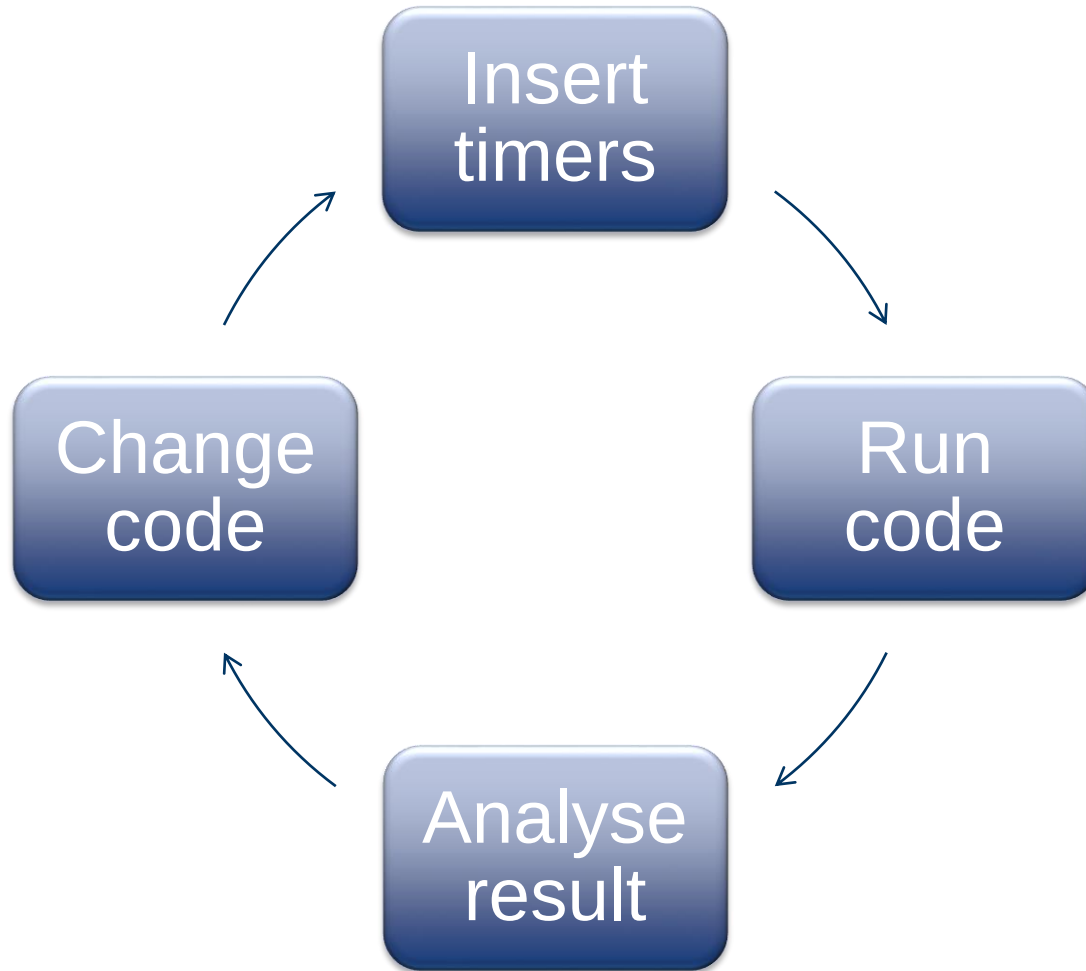
A: Development.

Development implies both fixing problems and optimizing the computation.

Debugging in Practice, ...



Optimization in Practice, ...



Simplified Code Optimization



Small data files



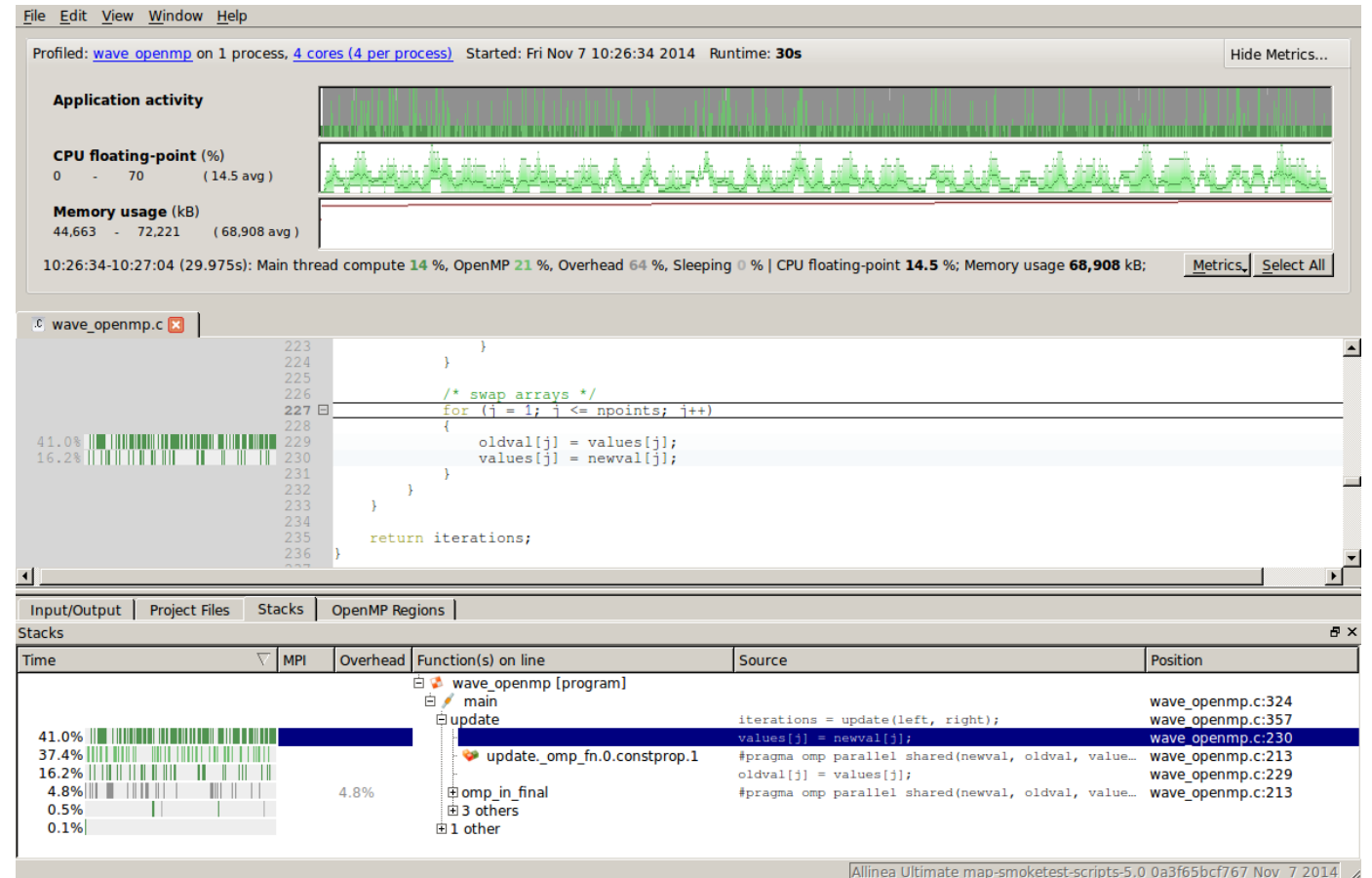
<5% slowdown



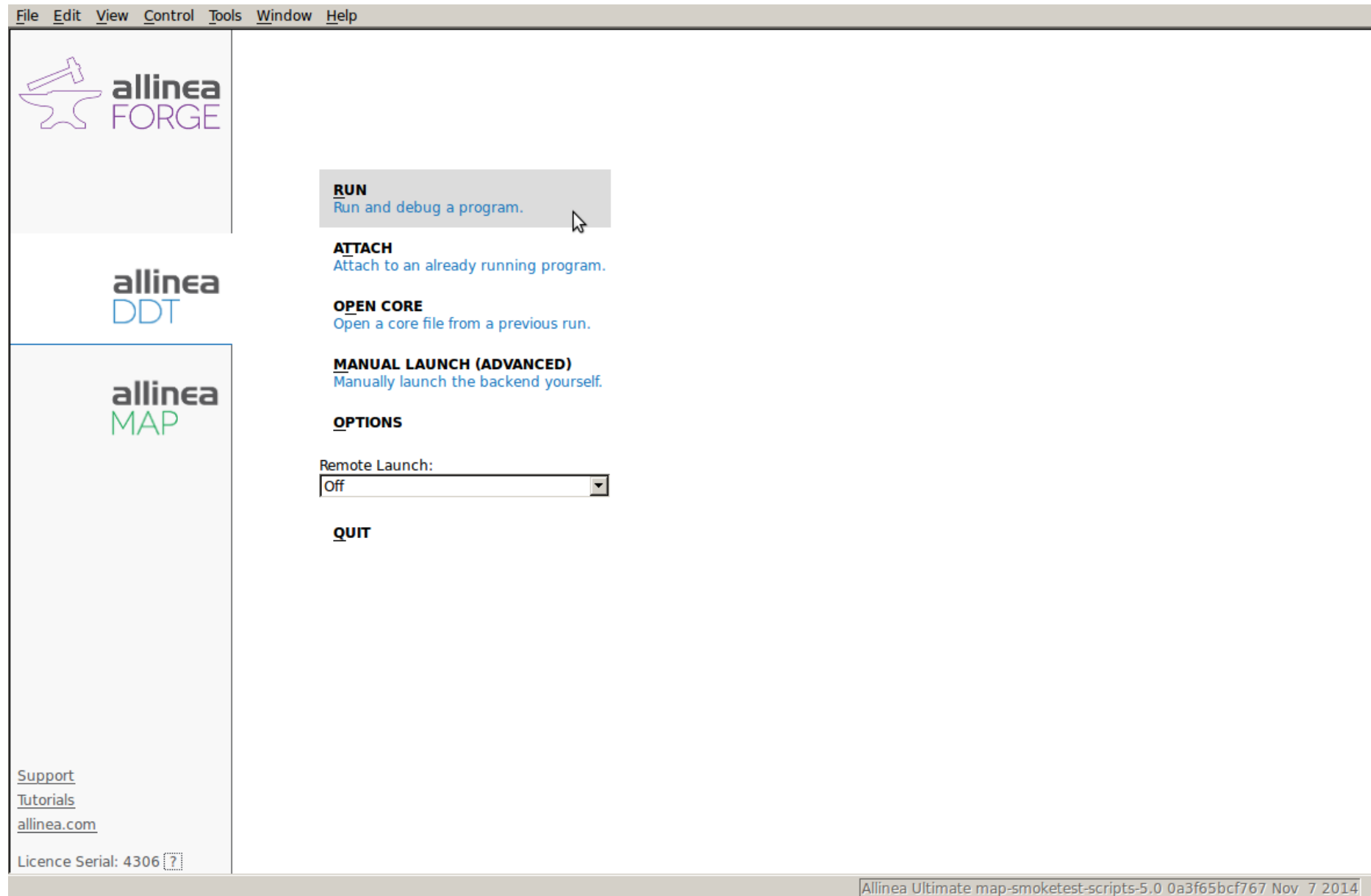
No instrumentation



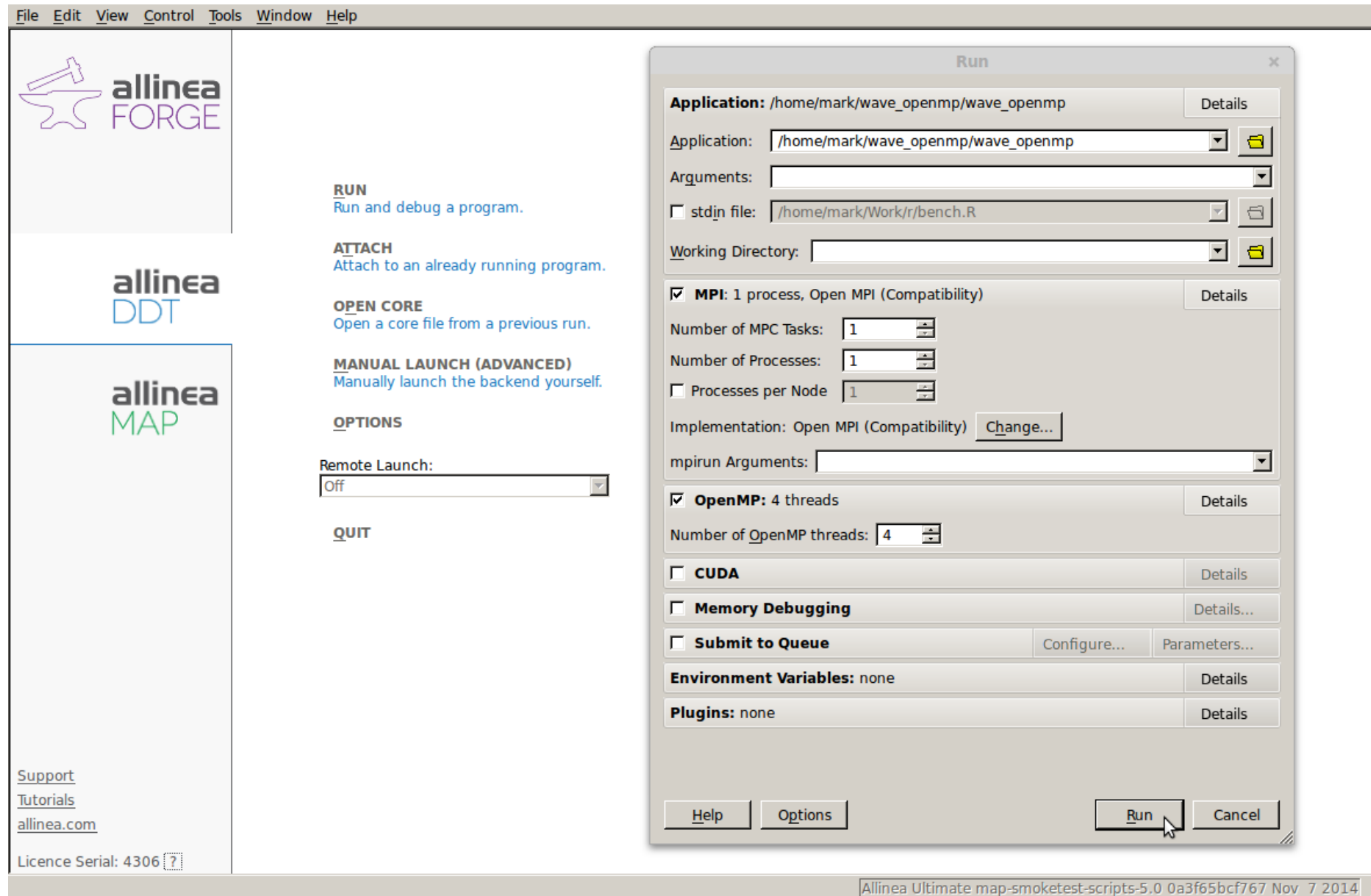
No recompilation



While still connected to the server we switch to the debugger



It's already configured to reproduce the profiling run



Our tools understand your version control system

The screenshot displays the Allinea DDT IDE interface. The 'View' menu is open, highlighting 'Version Control Information'. The main editor shows a C file named 'wave_omp.c' with OpenMP directives. The 'Locals' pane on the right shows the current line's variables. The 'Stacks' pane at the bottom shows the current function 'main (wave_omp.c:329)'. The 'Evaluate' pane at the bottom right shows the values of 'newval', 'oldval', and 'values' as '0x0'.

File Edit View Control Tools Window Help

Focus on Threads

Project Files Search (Ctrl+F)

App

External Code

Version Control Information

get_data(void) : void
init_line(void) : void
main(int argc, char **argv) : int
output_master(void) : void
output_workers(void) : void
reduce_print(const char *format, ...) : void
time_mpi_start : void
time_mpi_stop : void
update(int left, int right) : void

wave_omp.c

```
time_mpi_stop();  
  
/* update points along line */  
#pragma omp parallel shared(newval, oldval, values)  
{  
    #pragma omp for  
    for (j = 1; j <= npoints; j++)  
    {  
        /* global endpoints */  
        if ((first + j - 1 == 1) || (first + j - 1 == tpoin  
            newval[j] = 0.0;  
        else  
            do_math(j);  
    }  
  
    /* swap arrays */  
    oldval = values;  
    values = newval;  
}  
  
return iterations;
```

Locals Current Li... Current S...

Current Line(s)

Variable Name	Value
start	{tv_sec = 0, t

Type: none selected

Input/Output* Breakpoints Watchpoints Stacks Tracepoints Tracepoint Output Logbook Evaluate

Stacks

Function

main (wave_omp.c:329)

Expression	Value
newval	0x0
oldval	0x0
values	0x0

Ready

Most new bugs are in or around recently changed code

The screenshot displays the Allinea DDT debugger interface. The main window shows the source code for `wave_openmp.c`. A change set is highlighted on line 227, which contains the code `oldval = values;`. A tooltip for this change set is visible, providing details about the modification.

Change Set Details:

- changeset: 1:95f85f0e4dda
- tag: tip
- user: Mark O'Connor <mark@allinea.com>
- date: Fri Nov 07 11:37:23 2014 +0100
- summary: Swap arrays directly via their pointers instead of copying each element; this takes longer than the calc...

The interface also shows the **Locals** and **Current Line(s)** panels on the right, and the **Stacks** panel at the bottom. The **Stacks** panel shows the current call stack, including `main (wave_openmp.c:354)` and `update (wave_openmp.c:227)`.

Locals Panel:

Variable Name	Value
oldval	0x7ffff531c010
values	0x7ffff5abe010

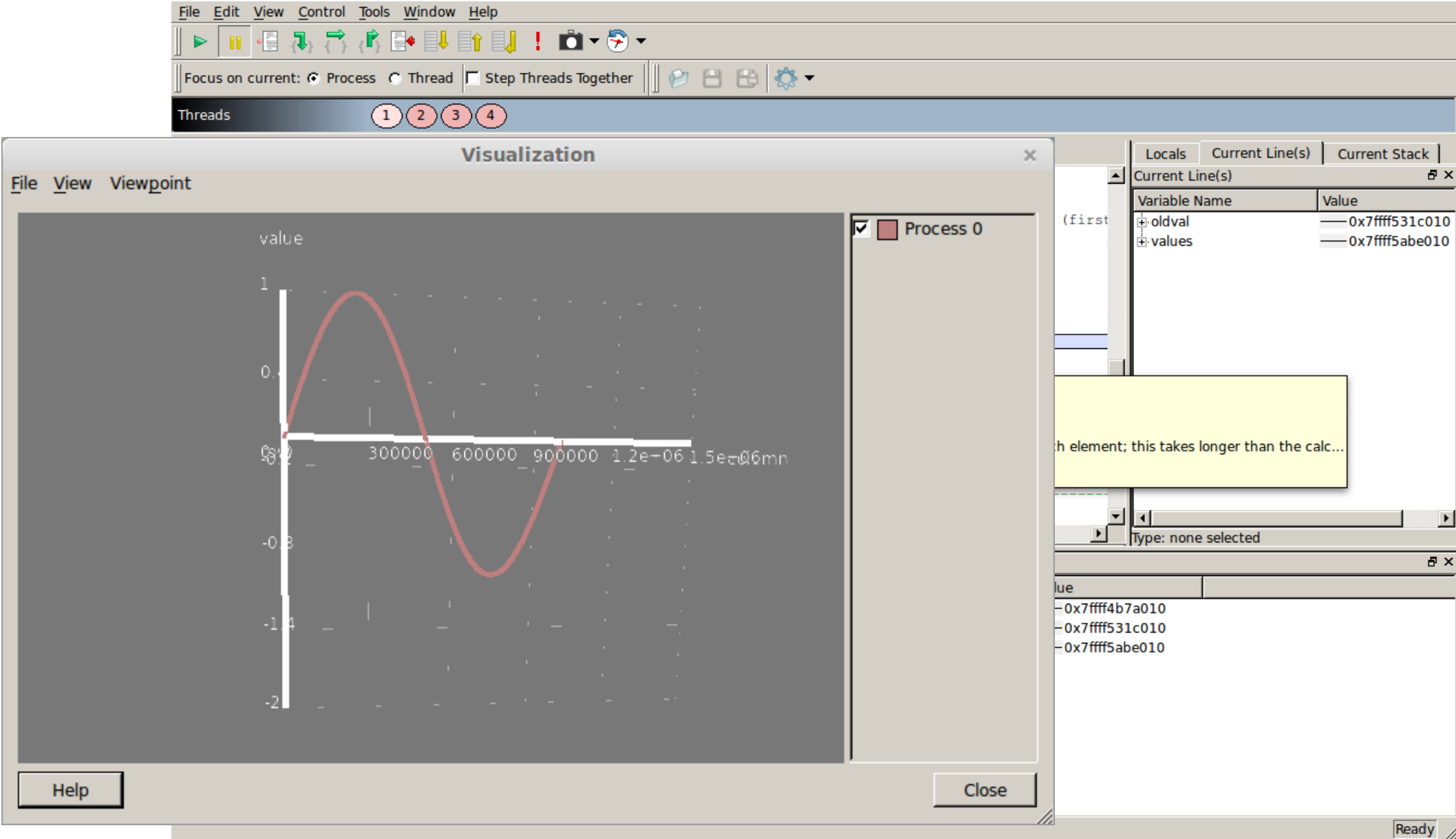
Stacks Panel:

Threads	Function
1	main (wave_openmp.c:354)
1	update (wave_openmp.c:227)
3	omp_in_final

The **Evaluate** panel at the bottom right shows the current state of variables:

Expression	Value
newval	0x7ffff4b7a010
oldval	0x7ffff531c010
values	0x7ffff5abe010

We can visualize multidimensional data across all processes



And generate statistical summaries of their contents

The image shows a software interface with two main windows. The 'Multi-Dimensional Array Viewer' window on the left displays statistical data for an array expression 'values[\$i]'. The 'Current Line(s)' window on the right shows code execution with variable values.

Multi-Dimensional Array Viewer

Array Expression: `values[$i]` [Evaluate] [Cancel]

Distributed Array Dimensions: `None` [How do I view distributed arrays?](#)

☐ Staggered Array [What does this do?](#)

Range of `$i`:

From: `0` To: `1000001` Display: `Rows`

☐ Only show if: [See Examples](#)

Data Table **Statistics**

Count:	1000002
Not shown:	0
Errors:	0
Aggregate:	0
Numerical:	1000002
Sum:	-1.26457e-08
Minimum:	-1
Maximum:	1
Range:	2
Mean:	-1.26456e-14
Variance:	0.499999
nan:	0
-nan:	0
inf:	0
-inf:	0
<0:	500000
=0:	3
>0:	499999

[Help] [Close]

Current Line(s)

Variable Name	Value
<code>oldval</code>	<code>0x7ffff531c010</code>
<code>values</code>	<code>0x7ffff5abe010</code>

Code Editor

```
for (j = 1; j <= npoints; j++)  
{  
    /* global endpoints */  
    if ((first + j - 1 == 1) || (first  
        newval[j] = 0.0;  
    else  
        do_math(j);  
}  
  
/* swap arrays */  
oldval = values;  
values = newval;  
}
```

85f0e4dda

Connor <mark@allinea.com>
07 11:37:23 2014 +0100
arrays directly via their pointers instead of copying each element; this takes longer than the calc...

output_master(void)

Type: none selected

Evaluate

Expression	Value
<code>newval</code>	<code>0x7ffff4b7a010</code>
<code>oldval</code>	<code>0x7ffff531c010</code>
<code>values</code>	<code>0x7ffff5abe010</code>

[Ready]

Variables are compared across all threads and processes automatically

The screenshot shows the Allinea DDT debugger interface. The main window displays the source code of a C program named `wave_omp.c`. A breakpoint is set at line 227, which is highlighted in red. The code is a multi-threaded program using OpenMP. The `update` function is currently executing, and the breakpoint is triggered. The `Threads` panel at the top shows four threads, with thread 1 selected. The `Stacks` panel at the bottom shows the call stack for the current thread, including `main` and `update`. The `Locals` panel on the right shows the current values of `oldval` and `values`.

Threads

1 2 3 4

Project Files

Search (Ctrl+K)

Application Code

- /
- Sources
 - wave_omp.c
- External Code

Source Code (wave_omp.c)

```
8 months ago 216 for (j = 1; j <= npoints; j++)
8 months ago 217 {
8 months ago 218     /* global endpoints */
8 months ago 219     if ((first + j - 1 == 1) || (first
8 months ago 220         newval[j] = 0.0;
8 months ago 221     else
8 months ago 222         do_math(j);
8 months ago 223     }
8 months ago 224 }
8 months ago 225
8 months ago 226 /* swap arrays */
8 months ago 227 oldval = values;
8 months ago 228 values = newval;
8 months ago 229 }
8 months ago 230 }
8 months ago 231 return iterations;
8 months ago 232 }
8 months ago 233
8 months ago 234 /*
8 months ago 235  * Master receives results from workers and print
8 months ago 236  *
8 months ago 237  */
8 months ago 238 void output_master(void)
8 months ago 239
```

Locals

Variable Name	Value
oldval	0x7ffff4b7a010
values	0x7ffff4b7a010

Stacks

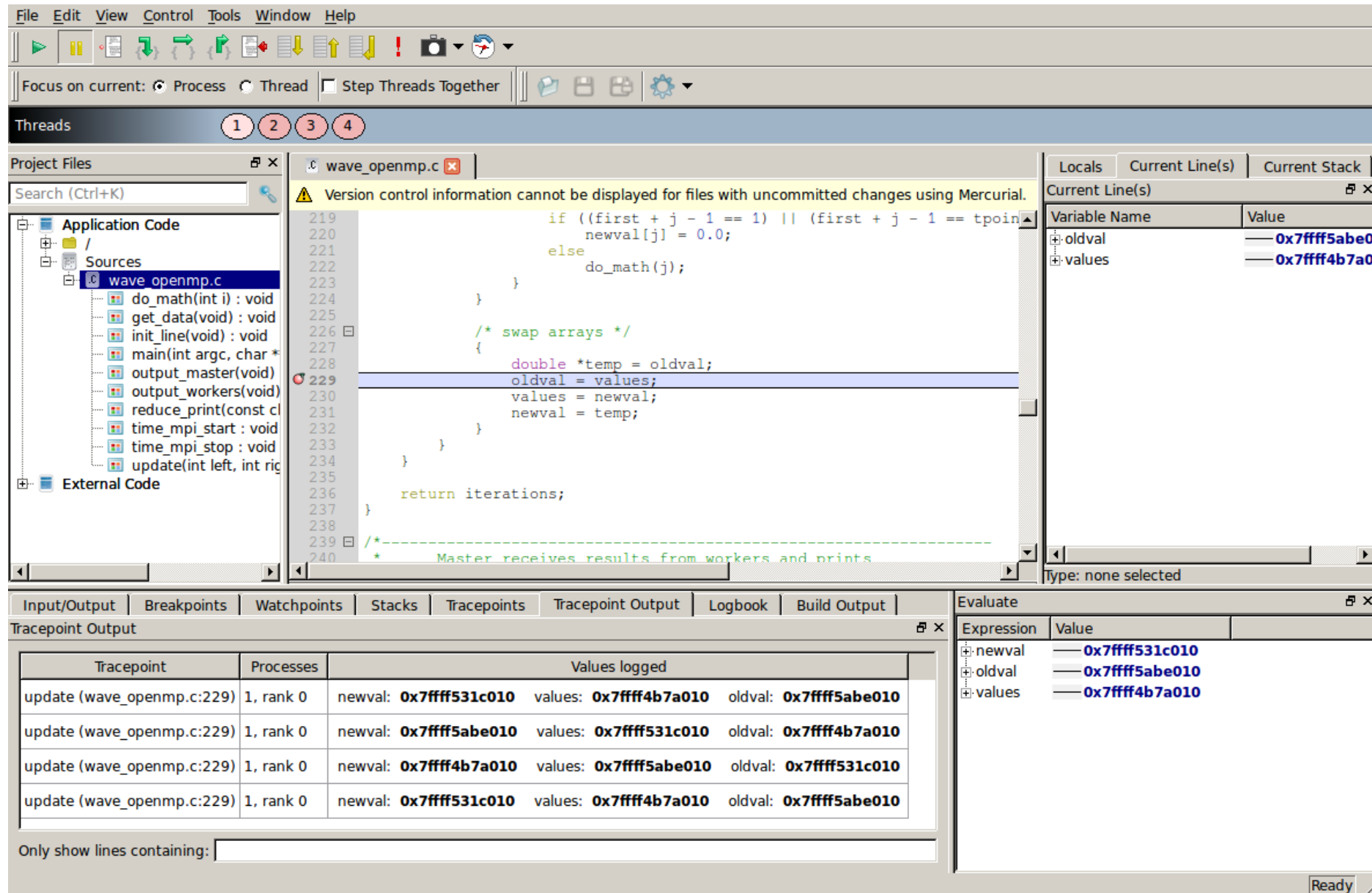
Threads	Function
1	main (wave_omp.c:354)
1	update (wave_omp.c:227)
3	omp_in_final

Evaluate

Expression	Value
newval	0x7ffff4b7a010
oldval	0x7ffff4b7a010
values	0x7ffff4b7a010

Ready

Verify our fix before committing it



The screenshot shows the Allinea DDT debugger interface. The main window displays the source code of `wave_openmp.c`. A breakpoint is set at line 229, which is highlighted. The right-hand side of the interface shows the 'Locals' and 'Current Line(s)' panels. The 'Locals' panel lists variables `oldval` and `values` with their current values. The bottom panel shows the 'Tracepoint Output' table, which contains four rows of data logged at the breakpoint.

Locals Panel:

Variable Name	Value
oldval	0x7fff5abe0
values	0x7fff4b7a0

Tracepoint Output Table:

Tracepoint	Processes	Values logged
update (wave_openmp.c:229)	1, rank 0	newval: 0x7fff531c010 values: 0x7fff4b7a010 oldval: 0x7fff5abe010
update (wave_openmp.c:229)	1, rank 0	newval: 0x7fff5abe010 values: 0x7fff531c010 oldval: 0x7fff4b7a010
update (wave_openmp.c:229)	1, rank 0	newval: 0x7fff4b7a010 values: 0x7fff5abe010 oldval: 0x7fff531c010
update (wave_openmp.c:229)	1, rank 0	newval: 0x7fff531c010 values: 0x7fff4b7a010 oldval: 0x7fff5abe010

A tracepoint shows the arrays pointers are swapping correctly now

The screenshot shows a debugger interface with a 'Commit Changes' dialog box open. The dialog has two sections: 'Modified Files:' and 'Commit Message:'. The 'Modified Files:' section lists 'M wave_openmp.c'. The 'Commit Message:' section contains the text 'Fixed array swapping, validation tests now pass again.' There are 'Commit' and 'Cancel' buttons at the bottom of the dialog.

In the background, the debugger's 'Tracepoint Output' window is visible, showing a table of tracepoint data. The table has columns for 'Tracepoint', 'Processes', and 'Values logged'. The 'Values logged' column contains three fields: 'newval', 'values', and 'oldval'. The tracepoint is 'update (wave_openmp.c:229)' and it is being executed by '1, rank 0'. The values are as follows:

Tracepoint	Processes	Values logged
update (wave_openmp.c:229)	1, rank 0	newval: 0x7fff4b7a010 values: 0x7fff5abe010 oldval: 0
update (wave_openmp.c:229)	1, rank 0	newval: 0x7fff531c010 values: 0x7fff4b7a010 oldval: 0
update (wave_openmp.c:229)	1, rank 0	newval: 0x7fff5abe010 values: 0x7fff531c010 oldval: 0
update (wave_openmp.c:229)	1, rank 0	newval: 0x7fff4b7a010 values: 0x7fff5abe010 oldval: 0

The 'Locals' window on the right shows the current line(s) and the values of local variables. The variables are 'oldval' and 'values', both with values 0x7fff53 and 0x7fff5a respectively.

The 'Evaluate' window at the bottom right shows the expression and value for the variables 'newval', 'oldval', and 'values'. The values are 0x7fff4b7a010, 0x7fff531c010, and 0x7fff5abe010 respectively.

Debug with the Scientific Method

The screenshot displays the Allinea DDT - Allinea Forge 5.0.1 [Trial Version] interface. The main window shows the source code of `cstartmpi.c` with the following visible lines:

```
84     t2 = malloc(sizeof(typeThree));
85
86     for(p=0;p<100;p++)
87         bigArray[p]=80000+p;
88
89     MPI_Init(&argc, &argv);
90     MPI_Comm_rank(MPI_COMM_WORLD, &my_rank);
91     MPI_Comm_size(MPI_COMM_WORLD, &n);
```

The interface includes several panels:

- Project Files:** Shows the project structure with `cstartmpi.c` and `func1: void`.
- Logbook:** A table of events with columns for Time, Ranks, and Message.
- Locals:** A table of local variables and their values.
- Evaluate:** A table for evaluating expressions.

Logbook Details:

Time	Ranks	Message
0:00	0-3	Launching program /home/bpaisley/demo/ddt/cstartmpi/cstartmpi.exe at Tue May 19 11:12:58 2015 Executable modified on Mon May 18 08:39:06 2015
0:03	0-3	Startup complete.
0:03	n/a	Select process group All
0:03	0-3	Add breakpoint for cstartmpi.c:164
0:03	n/a	Add Expression to Evaluate: argc
0:19	n/a	Hypothesis 1
0:27	0-3	Run to line 96 in cstartmpi.c
0:27	0-3	Process stopped in main (cstartmpi.c:96)

Locals:

Variable Name	Value
environ	0x7fffffd698
i	0
message	"\t"
my_rank	0

Evaluate:

Expression	Value
argc	1

The status bar at the bottom right indicates "Ready".

Debugging While you Sleep

report logbook

file:///home/bpaisley/demo/ddt/cstartmpi/report.html

debugging /home/bpaisley/demo/ddt/cstartmpi/cstartmpi.exe

Messages

Tracepoints

Memory Leaks

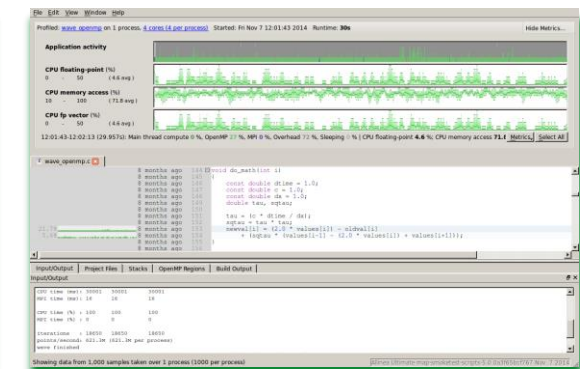
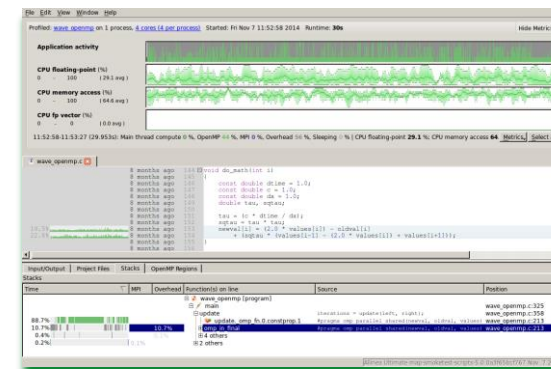
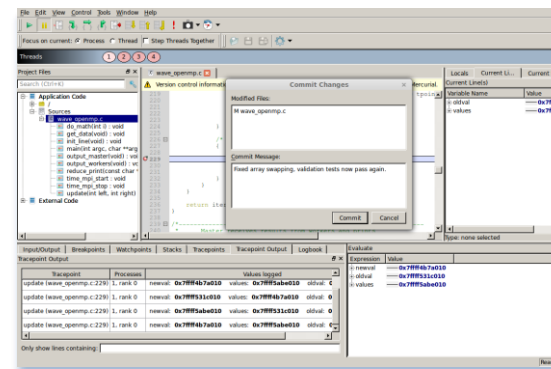
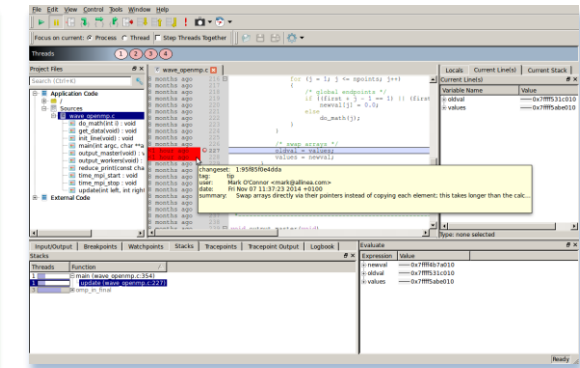
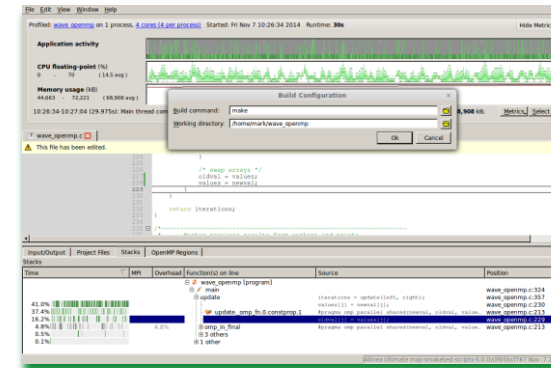
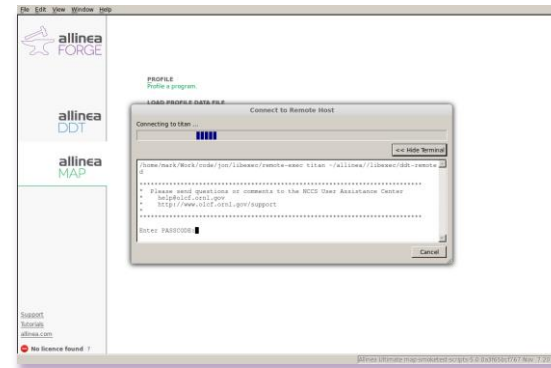
Output

Messages

[+] Expand All [-] Collapse All

#	Type	Time	Processes	Message												
1		0:00.000	0-3	Launching program /home/bpaisley/demo/ddt/cstartmpi/cstartmpi.exe at Wed May 6 11:34:30 2015 Executable modified on Mon Apr 27 13:22:13 2015												
2		0:01.649	0-3	Startup complete.												
3		0:01.655	n/a	Select process group All												
4		0:01.656	0-3	Add tracepoint for cstartmpi.c:109 Vars: x, y												
5		0:01.658	n/a	Add Expression to Evaluate: my_rank												
6		0:01.658	n/a	Add Expression to Evaluate: tables												
7				Additional Information ► Stacks ► Current Stack ▼ Locals <table><tr><th>Name</th><th>Value</th></tr><tr><td>argc</td><td>1</td></tr><tr><td>argv</td><td>0x7fffffff548</td></tr><tr><td>beingWatched</td><td>32767</td></tr><tr><td>bigArray</td><td></td></tr><tr><td>dest</td><td>0</td></tr></table>	Name	Value	argc	1	argv	0x7fffffff548	beingWatched	32767	bigArray		dest	0
Name	Value															
argc	1															
argv	0x7fffffff548															
beingWatched	32767															
bigArray																
dest	0															

A Productive HPC Workflow



allinea



High performance tools to debug, profile, and analyze your applications

<http://www.alcf.anl.gov/user-guides/allinea-ddt>

