Debugging Tools

Robert D. French
Adam Simpson
NCCS User Assistance
Memory Analysis and Debugging

ALLINEA DDT
DDT

• Graphical Parallel Debugging Tool
• Supports CPU, Memory, and GPU Debugging
  – Memory debugging similar to Valgrind
  – Catch out-of-bounds access
  – Race Detection
• DDT Remote Client (to avoid X11 tunneling)
Download Allinea DDT and Allinea MAP

Allinea DDT and Allinea MAP are part of one unified environment - you only need to install one binary package. The current release is version 4.2.

Windows and OS/X builds are remote clients only - they allow you to connect to a cluster and debug or profile on it but do not let you debug and profile programs running on Windows or OS/X.

Upgrading to a newer version

If you have a current support contract, all upgrades are free - just download the latest version and install it on top of or side-by-side with your current version.

Downloads
Download Allinea DDT and Allinea MAP

Allinea DDT and Allinea MAP are part of one unified environment - you only need to install one binary package. The current release is version 4.2.

Windows and OS/X builds are remote clients only, which allow you to connect to a cluster and debug or profile on it but do not let you debug and profile programs with Windows or OS/X.

Upgrading to a newer version

If you have a current support contract, all upgrades are free. Download the latest version and install it on top of or side-by-side with your current version.

Downloads
Remote client downloads for older versions

Download older Allinea tools remote clients for Windows and Mac OS X

Source code

Source code for the GPL 3, GPL 2, LGPL 2.1 and CPL licensed programs and libraries in the Allinea environment.
Remote client downloads for older versions

Download older Allinea tools remote clients for Windows and Mac OS/X

Source code

Source code for the GPL 3, GPL 2, LGPL 2.1 and CPL licensed programs and libraries in the Allinea environment.
You can download the latest full version of Allinea DDT and Allinea MAP from the downloads page.

These builds are for use as remote clients only - they allow you to connect to a cluster and debug or profile on it but do not let you debug and profile programs running on Windows or OS/X.

**Allinea Tools 4.1.1**

- Remote client for Mac OS X
- Remote client for Windows

**Allinea Tools 4.1**

- Remote client for Mac OS X
- Remote client for Windows

**Allinea Tools 4.0**

- Remote client for Mac OS X
- Remote client for Windows
You can download the latest full version of Allinea DDT and Allinea MAP from the downloads page.

These builds are for use as remote clients only - they allow you to connect to a cluster and debug or profile on it but do not let you debug and profile programs running on Windows or OS/X.

**Allinea Tools 4.1.1**

- Remote client for Mac OS X
- Remote client for Windows

**Allinea Tools 4.1**

- Remote client for Mac OS X
- Remote client for Windows

**Allinea Tools 4.0**

- Remote client for Mac OS X
- Remote client for Windows
DDT | Remote Launch

Options
Remote Launch:
- Off
- Configure...

Available Tools:
- Allinea DDT Remote Only
- Allinea MAP Remote Only
DDT | Remote Launch

Host Name: frenchrdf@titan.ccs.ornl.gov

How do I connect via a gateway (multi-hop)?

Installation Directory: /sw/xk6/ddt/4.1.1/sles11.1_binary/bin/

Script (optional):

Always look for source files locally

Test Remote Launch

Help OK Cancel
DDT | Remote Launch
DDT | Remote Launch

Licence Serial Number: 4929

Application:

Arguments:

Working Directory:

Number of Nodes: 1

Processes per Node: 16

Implementation: Cray XT/XE/XK/XC (MPI/shmem/UPC/CAF)

Queue Submission Parameters: Wall Clock Limit=01:00:00, Queue=

Environment Variables: none

Plugins: none

Support Expires 2016-07-01

Support | Tutorials | allinea.com
DDT | Remote Launch
DDT | Remote Launch

![DDT Remote Launch Interface](image-url)
DDT | Remote Launch
DDT | Remote Launch

Queue Submission Parameters

- Wall Clock Limit: 01:00:00
- Queue: batch
- Account: 

[OK] [Cancel]
DDT | Remote Launch

Queue Submission Parameters

Wall Clock Limit: 01:00:00
Queue: batch
Account: STF007

OK  Cancel
DDT | Remote Launch

![Remote Launch Screen](image-url)
DDT | Debugging
DDT | Debugging
DDT | Debugging
Create a group with these processes:

1,2,3 | 75% | Select All  | x2  | x0.5 | 1%

Group name: User Defined 1

[Buttons: Help, Create Group, Cancel]
DDT | Debugging
DDT | Debugging

The image shows a debugging interface with a focus on a specific line of code. The code snippet includes comments indicating the debugging process and variable values. The environment is set up for debugging with options for breakpoints, watchpoints, and stack traces.
Every process in your program has terminated – would you like to restart this session from the beginning?
Cray Performance Analysis Tool

CRAYPAT-LITE
Craypat-lite

• Provides Sample-based profiling
  – Checks periodically to see what function you are in

• To enable:
  – module load perftools-lite
  – Recompile
  – Launch as normal

• Generates reports & suggestions
Craypat-lite | Performance Statistics

CrayPat/X: Version 6.1.3 Revision 12145 (xf 12007) 11/18/13 21:56:10
Experiment:        lite sample_profile
Number of PEs (MPI ranks): 2
Numbers of PEs per Node: 1 PE on each of 2 Nodes
Numbers of Threads per PE: 1
Number of Cores per Socket: 16
Execution start time: Mon Jan 27 18:07:00 2014
System name and speed: chester-login2 2200 MHz

Wall Clock Time: 1.050018 secs
High Memory: 34.61 MBytes
MFLOPS (aggregate): 265.12 M/sec
I/O Read Rate: 30.87 MBytes/Sec
I/O Write Rate: 3.96 MBytes/Sec

Table 1: Profile by Function Group and Function (top 9 functions shown)

<table>
<thead>
<tr>
<th>Function Group</th>
<th>perfools_output.txt</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORMAL</td>
<td>9% : 10 : 1</td>
</tr>
</tbody>
</table>
### Craypat-lite | Performance Statistics

#### Table 1: Profile by Function Group and Function (top 9 functions shown)

<table>
<thead>
<tr>
<th>Samp%</th>
<th>Samp</th>
<th>Imb.</th>
<th>Imb.</th>
<th>Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>100.0%</td>
<td>92.0</td>
<td>--</td>
<td>--</td>
<td>Total</td>
</tr>
<tr>
<td>50.5%</td>
<td>46.5</td>
<td>--</td>
<td>--</td>
<td>MPI</td>
</tr>
<tr>
<td>49.5%</td>
<td>45.5</td>
<td>44.5</td>
<td>98.9%</td>
<td>MPI_Gatherv</td>
</tr>
<tr>
<td>1.1%</td>
<td>1.0</td>
<td>1.0</td>
<td>100.0%</td>
<td>MPI_Comm_create</td>
</tr>
<tr>
<td>41.3%</td>
<td>38.0</td>
<td>--</td>
<td>--</td>
<td>USER</td>
</tr>
<tr>
<td>19.6%</td>
<td>18.0</td>
<td>18.0</td>
<td>100.0%</td>
<td>hash_fluid</td>
</tr>
<tr>
<td>9.2%</td>
<td>8.5</td>
<td>8.5</td>
<td>100.0%</td>
<td>double_density_relaxation</td>
</tr>
<tr>
<td>7.6%</td>
<td>7.0</td>
<td>7.0</td>
<td>100.0%</td>
<td>viscosity_impluses</td>
</tr>
<tr>
<td>2.7%</td>
<td>2.5</td>
<td>2.5</td>
<td>100.0%</td>
<td>calculate_density</td>
</tr>
<tr>
<td>1.1%</td>
<td>1.0</td>
<td>1.0</td>
<td>100.0%</td>
<td>predict_positions</td>
</tr>
<tr>
<td>6.0%</td>
<td>5.5</td>
<td>5.5</td>
<td>100.0%</td>
<td>IETC</td>
</tr>
</tbody>
</table>
Craypat-lite | Performance Statistics

Observations and suggestions

MPI utilization:

No suggestions were made because each node has only one rank.

End Observations

Table 2: File Input Stats by Filename (top 10 files shown)

<table>
<thead>
<tr>
<th>Read Time</th>
<th>Read MBytes</th>
<th>Read Rate MBytes/sec</th>
<th>Reads</th>
<th>Bytes/Call</th>
<th>File Name [max10]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000226</td>
<td>0.006977</td>
<td>30.866392</td>
<td>106.0</td>
<td>69.02</td>
<td>Total</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>/proc/self/maps</td>
</tr>
</tbody>
</table>

Table 3: File Output Stats by Filename (top 10 files shown)

<table>
<thead>
<tr>
<th>Write Time</th>
<th>Write MBytes</th>
<th>Write Rate MBytes/sec</th>
<th>Writes</th>
<th>Bytes/Call</th>
<th>File Name [max10]</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORMAL</td>
<td>perfools_output.txt</td>
<td></td>
<td></td>
<td></td>
<td>77% : 80 : 1</td>
</tr>
</tbody>
</table>
Craypat-lite

• Provides Sample-based profiling
  – Checks periodically to see what function you are in

• To enable:
  – module load perftools-lite
  – Recompile
  – Launch as normal

• Generates reports & suggestions
  – pat_report
  – app2 (GUI)
Craypat-lite | Apprentice2 GUI

Profile

Function/Region Profile

49.5% = MPI_Gather
19.6% = hash_fluid
9.2% = doubl ... relaxation

Load Imbalance

44.5 = MPI_Gather
18.5 = hash_fluid
8.5 = doubl ... relaxation

CPU

Computation

49.5%

Programming_Model

50.5%

IO

0.0%

Memory Utilization

D1 cache hit ratio 99.8% hits
Process HiMem (MBytes) 34.609
LD + ST per TLB miss 17071.76 refs/miss

Data Movement

No data collected.
Craypat-lite | Apprentice2 GUI
nvprof

• GPU Kernel profiler from NVIDIA
• Produces sample-based profiling
  – Checks periodically to see which function is executing
• To load, simply `module load cudatoolkit`
  – Then `aprun nvprof application`
nvprof | Generating Profile Reports

```
[frenchrd@chester-login2 ~]$ qsub -I -X -A STF007 -lwalltime=2:00:00 -lnodes=2
qsub: waiting for job 173861 to start
qsub: job 173861 ready

[frenchrd@chester-login2 ~]$ module swap PrgEnv-pgi PrgEnv-gnu
[frenchrd@chester-login2 ~]$ module load cudatoolkit
[frenchrd@chester-login2 ~]$ cd $MEMBERWORK/stf007
[frenchrd@chester-login2 /lustre/atlas/scratch/frenchrd/stf007]$ aprun -n16 -N8 nvprof --metrics achieved_occupancy -o gol_occupancy_metrics.%h.%p ./GOL_shared_cu

==8159== NVPROF is profiling process 8159, command: ./GOL_shared_cu
==8158== NVPROF is profiling process 8158, command: ./GOL_shared_cu
==8152== ==8155== NVPROF is profiling process 8152, command: ./GOL_shared_cu
NVPROF is profiling process 8155, command: ./GOL_shared_cu
```
nvprof | Generating Profile Reports

```bash
---8180--- Generated result file: /lustre/atlas1/stf007/scratch/frenchrd/gol_occupancy_metrics.nid00073.8180
---8183--- Generated result file: /lustre/atlas1/stf007/scratch/frenchrd/gol_occupancy_metrics.nid00073.8183
---8169--- Generated result file: /lustre/atlas1/stf007/scratch/frenchrd/gol_occupancy_metrics.nid00073.8169
Total Alive: 45224
---8159--- Generated result file: /lustre/atlas1/stf007/scratch/frenchrd/gol_occupancy_metrics.nid00086.8159
Total Alive: 45224
Application 535024 resources: utime ~12s, stime ~43s, Rss ~88556, inblocks ~23430, outblocks ~15384
[frenchrd@chester-login2 /lustre/atlas/scratch/frenchrd/stf007]$ nvvp
[frenchrd@chester-login2 /lustre/atlas/scratch/frenchrd/stf007]$
```
nvprof | NVIDIA Visual Profiler (nvvp)
nvprof | NVIDIA Visual Profiler (nvvp)
nvprof | NVIDIA Visual Profiler (nvvp)
nvprof | NVIDIA Visual Profiler (nvvp)
nvprof | NVIDIA Visual Profiler (nvvp)
nvprof | NVIDIA Visual Profiler (nvvp)
nvprof | NVIDIA Visual Profiler (nvvp)

The guided analysis system walks you through the various analysis stages to help you understand the optimization opportunities in your application. Once you become familiar with the optimization process, you can explore the individual analysis stages in an unguided mode. When optimizing your application, it is important to fully utilize the compute and data movement capabilities of the GPU. To do this you should look at your application's overall GPU usage as well as the performance of individual kernels.
nvprof | Command Line Reports

[fr<eunchrd@che<ster-login2 /lustre/atlas/scratch/frenchrd/stf007]$ aprun -n1 nvprof --query-metrics

Could't parse executable

Available Metrics:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>l1_cache_global_hit_rate</td>
<td>Hit rate in L1 cache for global loads</td>
</tr>
<tr>
<td>l1_cache_local_hit_rate</td>
<td>Hit rate in L1 cache for local loads and stores</td>
</tr>
<tr>
<td>sm_efficiency</td>
<td>The percentage of time at least one warp is active on a multiprocessor</td>
</tr>
<tr>
<td>ipc</td>
<td>Instructions executed per cycle</td>
</tr>
<tr>
<td>achieved_occupancy</td>
<td>Ratio of the average active warps per active cycle to the maximum number of warps supported on a multiprocessor</td>
</tr>
<tr>
<td>gld_requested_throughput</td>
<td>Requested global memory load throughput</td>
</tr>
</tbody>
</table>
nvprof | Command Line Reports

```
[frenchrd@chester-login2 /lustre/atlas/scratch/frenchrd/stf007]$ aprun -n1 nvprof --metrics achieved_occupancy ./GOL_shared_cu
==8240== NVPROF is profiling process 8240, command: ./GOL_shared_cu
==8240== Profiling application: ./GOL_shared_cu
==8240== Profiling result:
==8240== Metric result:
Invocations | Metric Name | Metric Description | Min  | Max  | Avg
Device "Tesla K20X (0)"
  Kernel: GOL(int, int*, int*)
    1024 | achieved_occupancy | Achieved Occupancy | 0.807132 | 0.831816 | 0.813505
  Kernel: ghostCols(int, int*)
    1024 | achieved_occupancy | Achieved Occupancy | 0.035258 | 0.035897 | 0.035639
  Kernel: ghostRows(int, int*)
    1024 | achieved_occupancy | Achieved Occupancy | 0.034606 | 0.035680 | 0.034992
Total Alive: 45224
Application 535025 resources: utime ~2s, stime ~19s, Rss ~88552, inblocks ~4663, outblocks ~7567
[frenchrd@chester-login2 /lustre/atlas/scratch/frenchrd/stf007]$  
```
OLCF Debugging Tools

QUESTIONS?