Debugging Best Practices using Allinea DDT
Agenda

• Allinea DDT: debugging best practices
  – Configuring Allinea DDT
  – Overview of MPI debugging
  – Overview of GPU debugging
HPC's current challenge

• GPUs – a rival to traditional processors
  – AMD, Intel, ARM
  – NVIDIA, OpenCL, CUDA

• A big challenge for HPC developers
  – Data transfer
  – Several memory levels
  – Grid/block layout and thread scheduling
  – Synchronization

• New languages, compilers, potential standards
Allinea DDT
What is basic debugging?

• Developers need to see data
  – Too many variables to trawl them manually
  – Too many tasks or thread to display them at the same time

• Intelligent data management inside the debugger
  – Automatic monitoring of the data
  – Subtle highlights differences
  – Sparklines and smart display

• Even more detailed analysis if needed
  – Cross process comparison
  – Historical values of variables
Allinea DDT
In a Nutshell

• Graphical debugger designed for:
  – Multithreaded code
    ▪ Single address space
  – Multiprocess code
    ▪ Interdependent or independent processes
  – Parallel code
    ▪ Multi-node software
  – Any mix of the above
• Strong feature set and language support
  – Memory debugging
  – Data analysis
  – C, C++, Fortran, UPC, CUDA
• Managing concurrency
  – Emphasizing differences
  – Collective control
Allinea DDT

Gather, Sort and Display information

• User and administrator friendly

• Flow control

• Data monitoring

• Many environments
Allinea DDT

Gather, Sort and Display information

• User and administrator friendly
  – Get started easily
  – Fast, reliable, simple and intuitive GUI interface
  – Offline debugging

• Flow control

• Data monitoring

• Many environments
User and administrator friendly
Getting started

- Quick creation of runs
- Well integrated in workload schedulers
  - Loadleveler, LSF, SGE, PBS,...

![User interface screenshot of job submission settings and run parameters.](screen.png)
User and administrator friendly
Intuitive GUI interface

- Intelligent GUI that adapts to the environment
  - From workstation to large scale clusters
User and administrator friendly
Intuitive GUI interface

- Intelligent GUI that adapts to the environment
  - Even in CUDA environments
- Using a workload scheduler
  - Machines are available when the scheduler decides (by night ?)
  - Can be tricky to get a big cluster exactly when the developer wants it

- Offline debugging : printf replacement
  - Tracepoints and offline debugging
  - Job runs without debugger interface and record variables

- Worlds first scalable batch debugger
  - Set tracepoints, breakpoints, and run !
  - Memory debugging errors, crashes
  - Reports in HTML or plain text
Allinea DDT

Gather, Sort and Display information

• User and administrator friendly

• Flow control
  – Static analysis
  – Control progress at scale
  – Understand deadlocks
  – Start investigation

• Data monitoring

• Many environments
Flow control
Static analysis

• Fix those errors before they bite!
• Static analysis
  – Integrated with cppcheck
  – Also includes ftncheck
Flow control
Controlling progress at scale

- Bulk control is essential for multicore debugging
  - Group processes together
  - Play, step, reach breakpoints... Based on groups
  - Change interleaving order by stepping/playing selectively
Flow control
Resolving MPI issues

- Find cycles or blockage in message queue display
  - Parallel stacks
  - Variables
- More details than examining variables and processes alone

- We can see messages
  - MPI standard exists for debugging message queues
- Integrates with MPI correctness tools
  - Check correctness of the messages
  - Intel MPI Checker; Marmot

[Image of code snippets and debugging interface]
Flow control
Understanding what happens

- Application crashes
  - Threads/processes can be anywhere
  - Impossible to scroll through them individually

- Finding where processes crashed is essential
  - Allinea DDT merges stacks from processors and threads into a tree
  - Common faults patterns instantly evident
    - Divergence, deadlocks...
  - Information scalable without overload
Allinea DDT

Gather, Sort and Display information

• User and administrator friendly

• Flow control

• Data monitoring
  – Monitor variables
  – Detect memory errors
  – Check the calculation data

• Many environments
Data monitoring
Monitor variables

- Developers need to see data
  - Too many variables to trawl them manually
  - Too many tasks or thread to display them at the same time

- Intelligent data management inside the debugger
  - Automatic monitoring of the data
  - Subtle highlights differences
  - Sparklines and smart display

- Even more detailed analysis if needed
  - Cross process comparison
  - Historical values of variables
Data monitoring
Smart displays : tracepoints

• “printf” is still mostly used but serious drawbacks:
  – Need to recompile the code
  – Information randomly printed on screen (depending on interleaving)
  – One line per process

• Scalable and advanced printf :
  – First step to reconcile printf and GUI
  – Information sorted by steps
  – Merged by groups of processes
  – No information overload
  – Possibility to filter the printed values
  – Save output for offline analysis
Random errors are sneaky
- Can’t fix a bug that doesn’t repeat
- Often caused by memory issues
Memory debugging can force the bug
- Better to happen every time, than only during product demos

Data monitoring
Memory debugging:
- Places agent between memory library and user process
- Communicates problems to the debuggers
- Monitors usage: detect memory leaks
- Automatically protects ends of arrays
- Trigger instant stop on touching memory
- Many classes of errors can be checked
- Also has CUDA support
Data monitoring
Searching haystacks

- Arrays are the building blocks of HPC
  - Largest jobs accumulate TB of data
  - Usually 2GB or 4GB per core

- Integrated visualization tool
  - Search data across all tasks or threads
  - Data displayed on a picture as requested by the user
  - Export at runtime
Allinea DDT

Gather, Sort and Display information

- User and administrator friendly
- Flow control
- Data monitoring
- Many environments
  - Debugging at scale
  - ARM, Blue Gene, CUDA 4.0
Flow control
Debugging at scale

- Allinea DDT: designed architecture to scale
  - Tree network communicates with daemons
  - Logarithmic performance
  - Allinea partnership with largest users
    - US DoE Oak Ridge National Laboratories
    - Argonne National Lab, CEA (France)
- High performance debugging
  - Over 220,000 cores debugged simultaneously
  - Step all and display stacks in ~1/10s
- Usability
  - Scalable interface and features
    - Memory debugging
    - Array filtering
    - Data comparison, etc.
• **Need for performance**: creation of new hardware systems
  – Hardware complexity deported to software complexity
  – Tools particularly not adapted (initially released by hardware vendors)

• **Problems**:
  – Slow
  – Unreadable
  – Unmanageable
  – ...

But huge potential!
Many environments

Successful rework

• Built on NVIDIA’s low level efforts
  – cuda-gdb, compiler

• Execution model is unusual
  – GUI work required to support 32 thread units (warps) in blocks and grids

• Mixed GPU/CPU in one interface
  – Interaction with CPUs
  – Easy to switch between contexts (stacks, threads, data...)
  – Support multiple nodes
Many environments
Directives support

- Wide range of partnerships
  - Support the environments you use for hybrid development
  - You swiftly benefit from the latest updates
  - Read our latest white papers available with PGI and CAPS
OpenACC : New parallel programming standard

Fully supported within Allinea DDT

You benefit from our partnerships
Summary

• You can use an intelligent debugger to fix bugs quickly
  – Other methods have limited success and issues at scale
  – Intelligence makes debugging easier and faster for you

• Allinea DDT supports the environments that you use for hybrid development

• Our technical team is available to help you
  – Technical support
  – Organize other webinars
  – Any other need...
Thank you

Your contacts:
- US Technical Contact: Ian Lumb  ilumb@allinea.com
- Technical Support team: support@allinea.com