

allinea



Leaders in parallel software development tools

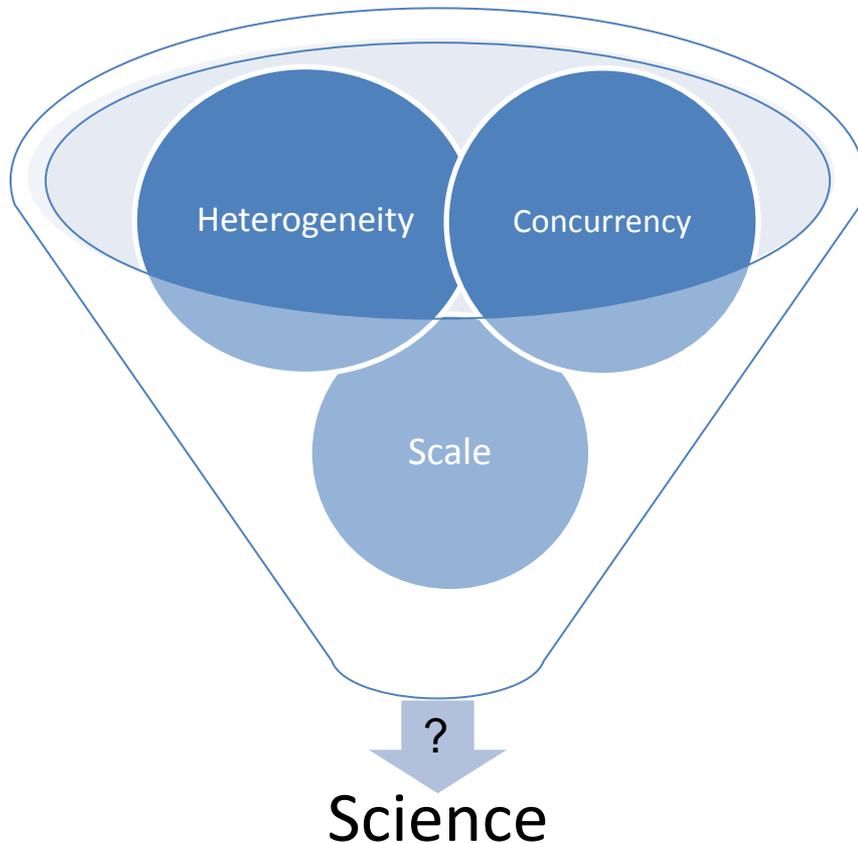
Getting Results from Applications at Scale on Mira Or “what to do when `write(*,*)` doesn’t cut the mustard”

David Lecomber,
Allinea Software
david@allinea.com

Agenda

- **The challenge today**
- **What's wrong with (the) Status Quo?**
- **What can tools do for me?**
- **Examples – live demos**

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.

Scalable science needs development tools

HPC is beyond the turning point for developers

- Print-style debugging cannot cope
- Systems too complex to understand performance
- Many existing tools fail to handle today's systems
- HPC experts are overloaded

Scalable systems need scalable tools

- Tools enable software to fully exploit the hardware
- Scale does not have to be hard
- Scale does not have to be slow

Allinea provides the solution

- Allinea DDT and Allinea MAP
- Proven scalable, Super-Petascale-capable tools
- We understand what HPC developers need

Collaboration – National Labs



Partnership to develop Petascale debugger with NVIDIA and Cray support for Titan



Partnership on Full Scale debugging on IBM BlueGene P & Q



Partnership with CEA French Atomic Energy Authority on scalable programming, CUDA and Alinea MAP



European partnership to develop techniques and solutions which address the exascale challenges



Partnership with BSC through its utilization of ARM technology to develop energy efficient HPC systems

Full speed debugging on Mira

- Scalable systems need *scalable* tools
 - Allinea and ANL collaborate to enable scalable Mira debugging
 - Millisecond responses at super-petascale
- Scalable systems need *stable* tools
 - Production quality at full scale
 - All debugging tasks/capabilities supported
- Ready to use at the extreme: **today!**
 - Regular Blue Waters and Titan sessions at the extreme
- **Don't print it, debug it!**

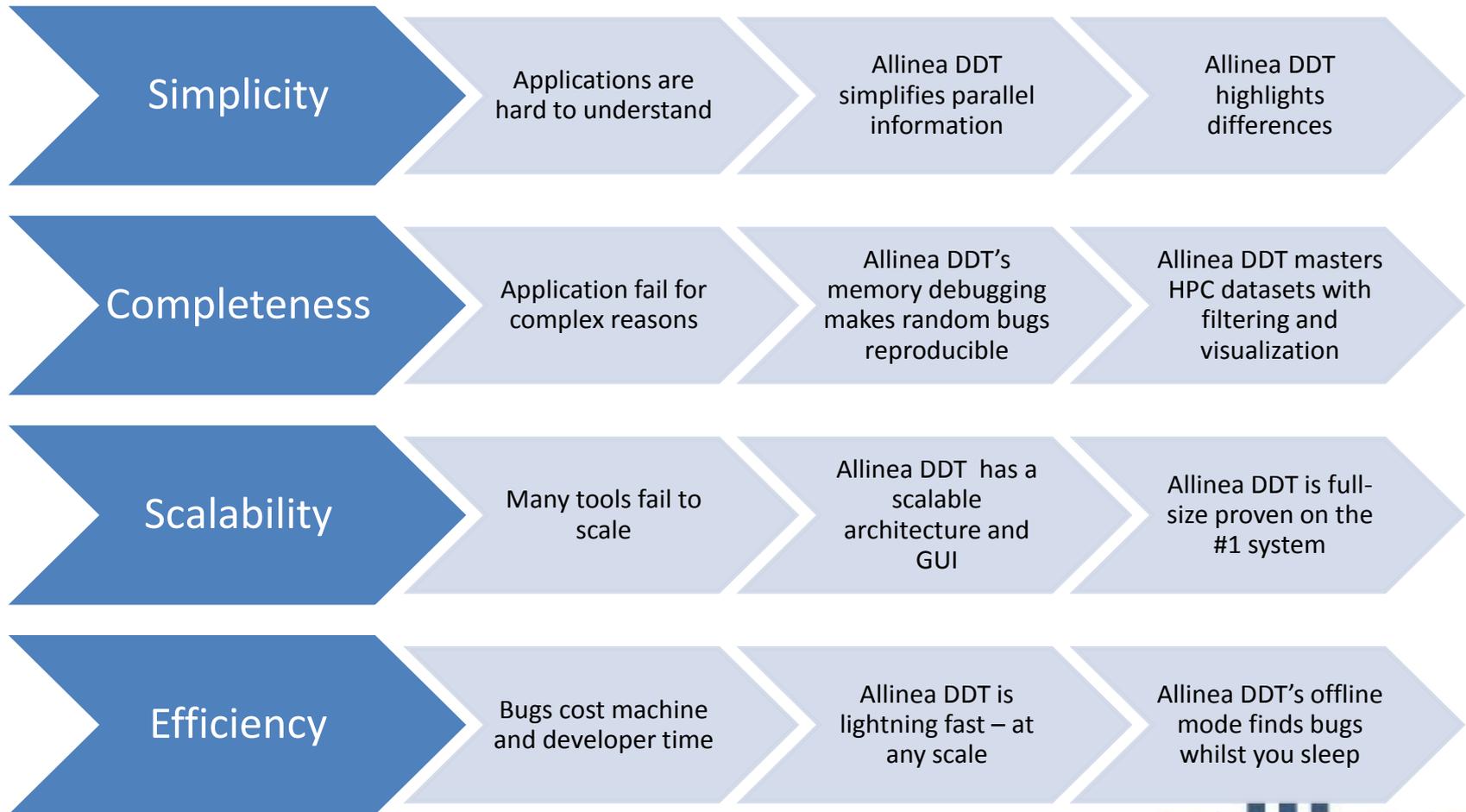


The Allinea Environment

- **At last: a modern integrated environment for the HPC developer**
- **Supporting the lifecycle of application development and improvement**
 - **Allinea DDT** : Productively debug code
 - **Allinea MAP** : Enhance application performance
- **Designed for productivity**
 - Consistent integrated easy to use tools
 - Enables effective use of HPC resources and expertise

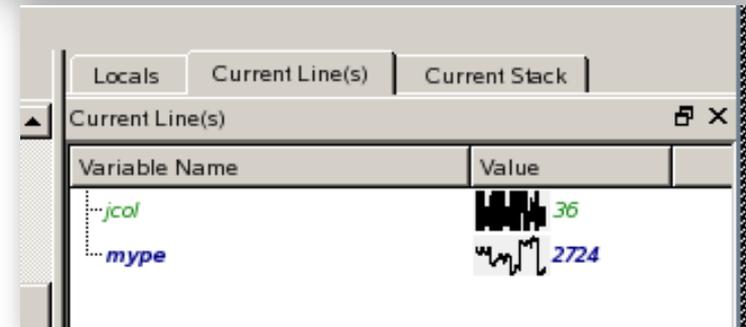
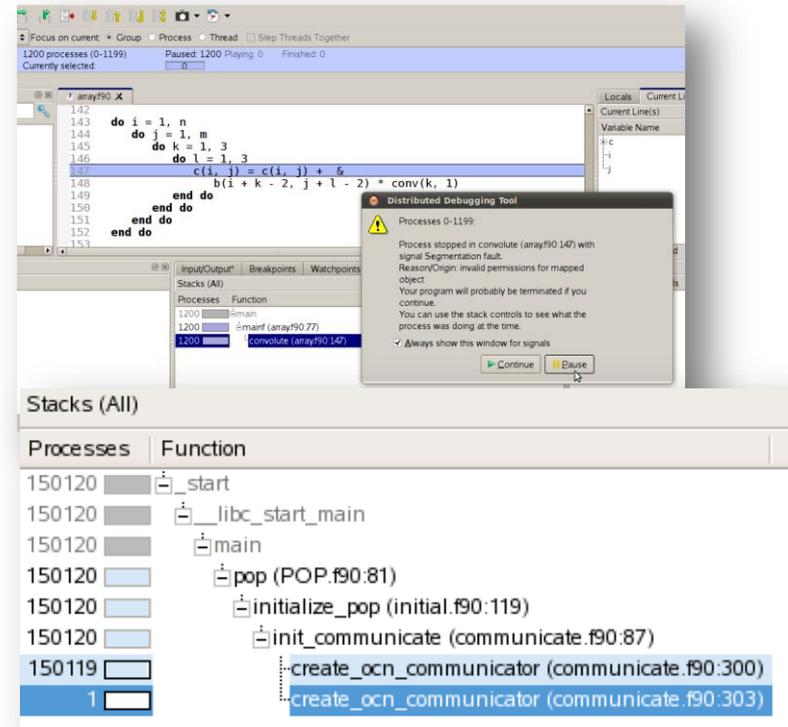


Allinea DDT – Debugging for HPC



Allinea DDT: Scalable debugging by design

- **Where did it happen?**
 - Allinea DDT leaps to source automatically
 - Merges stacks from processes and threads
- **How did it happen?**
 - Some faults evident instantly from source
- **Why did it happen?**
 - Real-time data comparison and consolidation
 - Unique “Smart Highlighting” – colouring differences and changes
 - Sparklines comparing data across processes
- **Force crashes to happen?**
 - Memory debugging makes many random bugs appear every time



Machine access tough?

- Use offline mode: debug while you sleep
 - Set breakpoints, tracepoints from command line
 - Memory debugging
 - Record variables, stacks – crashes and breakpoints
- Submit and forget
 - Post-mortem analysis
 - HTML/plain text
- Example:

`ddt -offline foobar.html -n 4 myfusioncode`

Stack for process 9

Process stopped in sched_yield (syscall-template.S:82) with signal SIGSEGV (Segmentation fault). Reason/Origin: kill, sigsend or raise
Your program will probably be terminated if you continue.
You can use the stack controls to see what the process was doing at the time.

Stacks

Processes	Threads	Function
1-15	15	applu (lu.f:118)
1-15	15	ssor (ssor.f:131)
1-15	15	blts (blts.f:55)
1-3,5-7,9-11,13,15	11	exchange_1 (exchange_1.f:37)
1-3,5-7,9-11,13,15	11	pmpl_recv
1-3,5-7,9-11,13,15	11	PMPI_recv
1-3,5-7,9-11,13,15	11	nca_pal_obl_recv
1-3,5-7,9-11,13,15	11	opal_progress
1-3,5-7,9-11,13,15	11	sched_yield (syscall-template.S:82)
4,8,12,14	4	exchange_1 (exchange_1.f:55)
4,8,12,14	4	pmpl_recv
4,8,12,14	4	PMPI_recv
4,8,12,14	4	nca_pal_obl_recv
4,8,12,14	4	opal_progress
4,8,12,14	4	sched_yield (syscall-template.S:82)

Stack for process 1

Every process in your program has terminated.

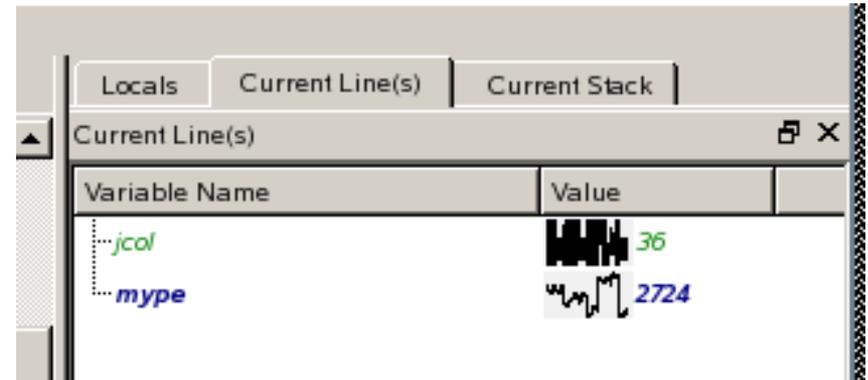
Messages Tracepoints Output

Tracepoints

#	Time	Tracepoint	Processes	Values
1	00:13.451	subdomain.f:96	0	jend: — 0 ny: — 9
2	00:13.453	ssor.f:177	0-15	delunm(5): — 0.25
3	00:13.455	ssor.f:177	0-15	delunm(5): — 0.25

Key features at scale

- Top 5 features at scale
 - Parallel stack view
 - Ideal for divergence or deadlock
 - Memory debugging
 - Rare bugs happen more frequently at scale
 - Parallel array searching
 - Data is too large to examine manually
 - Process control with step, play, and breakpoints
 - Still essential
 - Offline debugging
 - Access to machine may be hard – try offline debugging instead



What can I do next?

Learn about Allinea DDT



Ask a local expert



Contact Allinea:
support@allinea.com



Introductory tours:
<http://www.allinea.com/webinars>

How do I start on Mira?



Add "+ddt" to your .soft file



Compile your application with "-g"
flag



"ddt"