Linaro Forge

Debugging and Optimization Tools for HPC

HPC Development Solutions from Linaro

Best in class commercially supported tools for HPC



Performance Engineering for any architecture, at any scale

Supported Platforms



Debugging with DDT

An interoperable toolkit for debugging



- The de-facto standard for HPC development
 - Most widely-used debugger in HPC
 - Fully supported by Linaro on Intel, AMD, Arm, Nvidia, AMD GPUs, etc.

State-of-the art debugging capabilities

- Powerful and in-depth error detection mechanisms (including memory debugging)
- Available at any scale (from serial to exascale applications)

Easy to use by everyone

- Unique capabilities to simplify remote interactive sessions
- Innovative approach to present quintessential information to users



Linaro DDT Debugger Highlights



Multi-dimensional Array Viewer

What does your data look like at runtime?

View arrays

- On a single process
- Or distributed on many ranks

Use metavariables to browse the array

- Example: \$i and \$j
- Metavariables are unrelated to the variables in your program
- The bounds to view can be specified
- Visualise draws a 3D representation of the array

Data can also be filtered

• "Only show if": \$value>0 for example \$value being a specific element of the array



Debug Example - Fortran code

	Linaro DDT - Linaro Forge 23.0	- σ X
e <u>E</u> dit <u>V</u> iew <u>C</u> ontrol <u>T</u> ools <u>W</u> indow	Help	
> 🔲 🗄 🞝 📅 🖪 🖾 📑 📑		
Irrent Group: All - Focus on cur	rrrent: Group O Process O Thread Step Threads Together	
0 1 2		
ot		
aves 1 2 3	3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19	
eate Group		
oject Files Fortran Modules	I hello,f ×	Locals Current Line(s) Current Stack Raw Command
oject Files	ALLOCATE (domain (my_rank*100000))	- Locals Ø®
arch (Ctrl+K)	$domain (i) = 2^{+i} + 1$	Name Value =
Application Code	53 END DO	• mpi_ergvs_null
Headers	55 message="Hello From Me !"	mpi_bottom — 0
 Sources 	56	mp_in_place — 0
I hello.f	57 PRINT *, "My rank is ", my_rank, "!"	mp[status_ign
<pre> forge_constants FUNC1() : INTEGER </pre>	C 59 CALL SUB1()	 mpi_statuses mpi_unweighted
hellof77	60	 mpi_weights_e
show_consts	61 Deingwatched=1 62 tag=0	dest — 0
SUBI External Code	63	+ domain
	64 IF (my_rank.ne.0) THEN 0.65 PRINT * (Greatings from process " my rank "!"	
	66 PRINT * "Sending message from ", my_rank, "!"	message 'Hello from my process'
	67 dest=0	my_rank 0
	69 c.ierr)	my_size — 20
	70 beingwatched=beingwatched=1	stat 0
	0 72 message="Hello from my process"	tag 0
	73 - DO source=1, (my_size=1)	ompi release v
	74 PRINT *, "waiting for message from ", source	ompi major ve — 3
	76 c, stat, ierr)	ompi_comm_ty — 6
	77 PRINT *, "Message recieved: ", message	omp_comm_ty — 7 ompi comm_ty — 0
	79 END DO	ompi_comm_ty — 5
	80 END IF	ompi_comm_ty — 4
	81 82 beingwatched=12	ompi_comm_ty — 1
	83 CALL MPI Finalize (ierr)	ompi_comm_ty — 9
	84 beingwatched=0	ompi_comm_ty 10
	86 END	omp_comm_y = 2 ompi_comm_tv = 11
	87	ompi_comm_ty — 8
		T
nput/Output* Breakpoints Watchpoin	ints Stacks (All) Tracepoints Tracepoint Output Logbook	Evaluate
acks (All)		Bar Value Value
cesses Threads Function	* (56)	my fank / 0
1 hellof77 (hello.f:		*
40 progress_engine	e.	
		Type: integer(kind=4)
		Ready
		L. L

Linaro Performance tools

Characterize and understand the performance of HPC application runs



Commercially supported

by Linaro

- Gather a rich set of data
 - Analyses metric around CPU, memory, IO, hardware counters, etc.
 - Possibility for users to add their own metrics



Build a culture of application performance & efficiency awareness

- Analyses data and reports the information that matters to users
- Provides simple guidance to help improve workloads' efficiency



Relevant advice to avoid pitfalls

Adds value to typical users' workflows

- Define application behaviour and performance expectations
- Integrate outputs to various systems for validation (eg. continuous integration)
- Can be automated completely (no user intervention)

MAP Capabilities

MAP is a sampling based scalable profiler

- Built on same framework as DDT
- Parallel support for MPI, OpenMP, CUDA
- Designed for C/C++/Fortran

Designed for 'hot-spot' analysis

- Stack traces
- Augmented with performance metrics

Adaptive sampling rate

- Throws data away 1,000 samples per process
- Low overhead, scalable and small file size



Linaro MAP Source Code Profiler Highlights



MAP Example - ROCm AMD GPU

<u>F</u> ile	<u>E</u> dit	<u>V</u> iew	<u>M</u> etrics	<u>W</u> indow	<u>H</u> elp										
Profile	d: <u>mix</u>	ed-cpu	<mark>i-gpu</mark> on 3	processes	s, 1 node	, <u>3 cores</u>	s (1 per proc	ess) Sampled	from: Mon Fe	b 28 21:22:2	4 2022 for 6	5.1s	lide Metr	rics	
Ma	in thr	ead a	tivity												
GP	U utili	zatior	1	100											
	67.7	%			-	-	-			- · ·					
GP	U mer	nory ı	isage	100										_	
	4.1 9	6													
GP	U mer	nory ı	ıtilizatior	n ¹⁰⁰										_	
	6.7 %	6													
21:22	:24-21	1:22:30	(6.131s):	Main thre	ad comp	ute 31.2	2 %, MPI 7.3	%, Accelerator	61.4 %	-		<u>Zoc</u>	om 🔍 i	₹ 0	
m m	ixed-c	pu-gpu	срр 🗙]											
					10 # 11 #	define define	BLOCK_SIZ DURATION	E 32 5.0						-	
					13 •	_qloba	1 void M	MatrixMulHIP	(float *C,	float *A,	float *B	, int w	A, int	hA,	
					14 { 15 16	flo	at temp =	0;							
4					17	int	i = block	Idx.y * bloc	ckDim.y + t	threadIdx.	y;			-	
				(
Input	/Outp	ut F	roject File	s Main	Thread S	Stacks	Functions	GPU Kernels						99	
GPU K	emeis			A G	PI L Korne	Jc.		Sourco						만폰	
28.1	.%			- 0	mixe 🌮 Ma	d-cpu-g trixMulH	pu [program] IIP	GPU: line-leve	el information	n is not availa	ble for ROC	m kernel	s		
25.4	1%				😻 Ma	trixMulH	IIPShared								
Showi	ng dat	a from	900 sam	ples taken	over 3 p	rocesses	s (300 per pr	ocess)		Arm	Forge 22.0	🤹 Main	Thread \	/iew	

Profile

- Ran for 6s, taking 300 samples per process
- Able to bring up metadata of the profile
- Mixed CPU [green] / GPU [purple] application
- CPU time waiting for GPU Kernels [purple]
- GPU Kernels graph indicating Kernel activity

GUI information

- GUI is consistent across platforms
- Zoom into main thread activity
- Ranked by highest contributors to app time

Performance Reports Example - Python

Report information

- Summary of application performance
- Actionable insight
- Could integrate into CI/CD

Generating a report

- Able to generate from a MAP file
- Generated through the command-line
- Text output or HTML report

	Command:	/ορι/ργιπο
	Resources:	1 node (1
Linaro	Memory:	15 GiB pe
Performance	Tasks:	1 process
Reports	Machine:	thoth
	Architecture:	x86_64
	CPU Family:	alderlake

/opt/python/3.8.5-shared/bin/python3 python-profiling.py 1 node (12 physical, 16 logical cores per node)
15 GiB per node
1 process thoth
x86_64
x86_64
alderlake
Tue May 30 14:44:01 2023
29 seconds
/opt/python/3.8.5-shared/bin



Linaro Forge

Summary: python-profiling.py is Python Interpreter-bound in this configuration



Time spent running application code. High values are usually good. This is **low**; consider improving MPI or I/O performance first

Time spent in MPI calls. High values are usually bad. This is **very low**; this code may benefit from a higher process count

Time spent in filesystem I/O. High values are usually bad. This is **negligible**; there's no need to investigate I/O performance

Time spent in the Python interpreter. Calling out to precompiled libraries may be more efficient.

This is average; consider moving more logic to compiled libraries

This application run was Python Interpreter-bound.

As very little time is spent in MPI calls, this code may also benefit from running at larger scales.

Running on Frontier

Forge is already on Frontier module load forge

DDT offline debugging ddt --offline srun -A <project_id> -t 00:05:00 -p <partition> -N 2 -n 4 --ntasks-per-node=2 ./app

MAP offline profile map --profile srun -A <project_id> -t 00:05:00 -p <partition> -N 2 -n 4 --ntasks-per-node=2 ./app

Performance report perf-report srun -A <project_id> -t 00:05:00 -p <partition> -N 2 -n 4 --ntasks-per-node=2 ./app

For remote connection

https://docs.linaroforge.com/latest/html/forge/forge/connecting_to_a_remote_system/connecting_remotely.html

Thank you

Go to <u>www.linaroforge.com</u> rudy.shand@linaro.org