

International HPC Summer School 2019: Performance analysis and optimization

Vampir, TAU, Extra-P, Darshan

VI-HPS Team Ilya Zhukov – Jülich Supercomputing Centre



Vampir Event Trace Visualizer



- Offline trace visualization for Score-P's OTF2 trace files
- Visualization of MPI, OpenMP
 - and application events:
 - All diagrams highly customizable (through context menus)
 - Large variety of displays for ANY part of the trace
- http://www.vampir.eu
- Advantage:
 - Detailed view of dynamic application behavior
- Disadvantage:
 - Requires event traces (huge amount of data)
 - Completely manual analysis

Vampir Displays



Vampir: Timeline Diagram

- Functions organized into groups
- Coloring by group
- Message lines can be colored by tag or size
- Information about states, messages, collective and I/O operations available through clicking on the representation



Vampir: Process and Counter Timelines

- Process timeline show call stack nesting
- Counter timelines
 for hardware and
 software counters



Vampir: Execution Statistics

- Aggregated profiling information: execution time, number of calls, inclusive/exclusive
- Available for all / any group (activity) or all routines (symbols)





V VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

¥ File ∨iew Help View Chart Filter

Vampir: Process Summary

¥ Fil ¥ Fil

- Execution statistics over all processes for comparison
- Clustering mode available for large process counts

CS	5							1	: 🚧 🌕	5 🏭 👪 4	🗿 🍪 🤅	9 💡 🤉	1	
									0 s					Pr
								0	s	3.s (ō,s	9 s	13	2 s
								Pros 0	solve_em_	module	.driver_ M	PI_Wait		mod.
								Pros l	MPI_Bcast	S	olve_em_		module_	drive
								Pros 2	MPI_Bcast		solve	em	m	dule
								Pros 3	MPI_Bcast	_	solve_em_		module_	drive
								Pros 4	MPI_Bcast	S	olve_em_		module	driver
								Pros 5	MPI_Bcast	S	olve_em_		module_	drive
								Pros 6	MPI_Bcast	sc	live_em_	l man du	nodule	. driver
								Pro s P	MPI_Bcast	5	lve_em_	mod	ulo dri	
								Pro e 9	MPL Boast	50	we_em_	mod	nodule	driver
								Dr 10	MPL Boast	30	olve em		module	drive
								Pr 11	MPL Boast		olve em		module	drive
								Pr12	MPL Boast	50	ove em		module	drive
								Pr13	MPI Bcast	sc	lve em	_	module	drive
								Pr14	MPI Bcast		solve em		module	driv
								Pr15	MPI Bcast		solve em	mo	duleriv	er 👖
								Pr16	MPI Bcast		solve_em	n n	nodule	driver
						•		Pr17	MPI_Bcast		solve_en	n	modu	led
	Van	npir - [Trace View - /hor	me/dolescha/trac	files/feature	-traces/wrf-p64	-io-mem-rusa	age/wrf.1h.ot	fl	- x	1	olve_em_		module	drive
e \	/iew Help								_ 8 ×	1	solve_em_		module	driv
cł	nart Filter									s	olve_em_		module_	drive
- L.	- III 🌰 💳	- 9 22 - 100 - 120 - 120 - 120									solve_em		module	dri∖
Ð	ध 🏭 🗢 🖂) 📰 📰 🔛 🚳 🕓									s	olve_em_		mod
	0			Process Summ	ary						so	ve_em_	mo	duler
c	us)s 3	s 6s 9	s 12 s	15 s	18 s	21 s 2	24 s 27	's 3	80 s		solve_em_	mo	oduled	river_
ī	MPI_Bcast	solve_em_	moduledriv	MPI Wait	moder_in	nodld_ mo	p open n	n			solve_em_		module	driv
Ш	MPI_Bcast	solve_em	moduleriver	MPI_Wait	m in the second s	oduver_mo	dld_ mop_	openm		1	solve_em	-	module	driv
Ĩ	MPI_Bcast	solve_em_	moduledriver_	MPI_Wait	moder_ mod	l ldmop_	open <mark>1</mark> m.	·_			solve_em_	-	module	driv
	solve_em_	moduledriver_ MPI_V	Vait mo	rmodld_	mop_ open	m	extose_ <mark>reac</mark>							
	MPI_Bcast	solve_em_	moduledrive	r_ MPI_Wait	moder_ mod	ldmopc	ipen <mark>m</mark>							
Ш	MPI_Bcast	solve_em_	moduleriver_	MPI_Wait	noder_t	modld_ mo.	p_openr	n						
	MPI_Bcast	solve_em_	modulerive	MP_Wait	mod…er_ <mark>`_ n</mark>	nodld_ mo	p open n	יייר						
Ш	MPI_Bcast	solve_em_	moduledrive	rMPIWait	noder	modldr	nop_ open	.m						
Ц	MPI_Bcast	solve_em_	moduler	ver_ <mark>MPI_Wa</mark>	it L moder	modld_	mop_ open	m						
Ц	MPI_Bcast	solve_em_	modulerive	er_ MPI_Wait	moder_	modldm	op_lopen_l	m						
Ц	MPI_Bcast	solve_em_	moduleriv	er_ MPI_Wait	L modu	ver_modld	oper	<u>m</u>						
	MPI_Bcast	solve	e_emmoo	luleriver_	IPI_Wait	moder_1	modld_ mo.	p_ m						
Щ	MPI_Bcast	scive_em_	moduleriver_	MPI_Wait r	noduvermod.	a_ m c	pen m							
╓	MPI_Bcast	solve_em_	modulerivi	MPI_Wait	moder_1m	oaia_ mo	p_open_1m.							
ш	MPI_Bcast	solve_em_	moduleri	ver_ wini_Wat	noduver_1	modid_ mo	p_ open 1 m	···_						
	: :	:	· · ·	:	1	:			:					

Process Summary	
3 s 6 s 9 s 12 s 15 s 18 s 21 s 24 s 27	s 30 s
moduledriver_MPI_Waitmoder_modld_mop_openmextose_read	
solve_emmodule_idriverMPI_WaitmodermodIdmopopenm	
solve_emmoduledriverMPI_Waitmodermodldmopm	
solve_emmoduledriverMPI_Waitmodermodldmopopenm	
solve_em <mark>moduledriver_</mark> MPI_Vaitmoder_modldmopopenm	
scive_em <mark>moduledriver_</mark> MPI_Vait <mark>naoder_</mark> _modldmopopenm	
solve_emmoduledriver_MPI_Vaitmodermodldmopopenm	
solve_em <mark>moduleriverMPI_Waitmoder_</mark> modldmopopenm	
solve_em <mark>moduledriverMPI_Waitmoder_</mark> modldmopcpenm	
solve_em <mark>moduledriverMPI_Waitmoder_</mark> _modldmopopenm	
solve_emmoduledriver_MPI_Waitmodermodldmopopenm	
scive_emmoduledriver_MPI_Waitmodermodldmopdpenm	
solve_emmoduledriver_MPI_Waitmoder_modlcmopopenm	
solve_emmodule_jdriver_MPI_Waitmodler_modlet_modlet_modlet_modlet_modlet_modlet	
solve_emmoduledrver_MPI_waitmodermodldmop_ openm	
solve_emmodulemvermin_watemodermodtomopen_m	
sove en nouse and an and a deixer and a so as a de so as as	
solve_emmodulemermedarcmodemoddmop_open_m	
solve em module driver MPI Wait module module module mon popen m	
solve em module driver MPI Wait mod er mod Id mo n open m.	
solve em module udriver MPI Wait moduler moduld moup open mul	
solve em moduledriver MPI Wait moduler moduld mop m.	
solve em moduleriver MPI Wait moder modld mop m	
solve em moduledriver MPI Wait moder modid mop open m	
solve em moduledriver MPE Wait moder modld mop cpen m	
solve_emmoduledriver_MP} Waitmodermodldmopopenm	
solve_emmoduledriver_MP Waitmoder_modld_mopopen_m	

Vampir - [Trace View - /home/dolescha/tracefiles/feature-traces/wrf-p64-io-mem-rusage/wrf.1h.otf]

1

Vampir - [Trace View - /home/dolescha/tracefiles/feature-traces/wrf-p64-io-mem-rusage/wrf.1h.otf]

Vampir: Communication Statistics

- Byte and message count, min/max/avg message length and min/max/avg bandwidth for each process pair
- Message length statistics



Vampir: CUDA Example

- Detailed information on kernel execution and memory transfers
- All statistics and displays also available for CUDA events



TAU

Very portable tool set for

instrumentation, measurement and analysisof parallel multi-threaded applicationshttp://tau.uoregon.edu/

- Supports
 - Various profiling modes and tracing
 - Various forms of code instrumentation
 - C, C++, Fortran, Java, Python
 - MPI, multi-threading (OpenMP, Pthreads, ...)
 - Accelerators



TAU: Instrumentation

- Flexible instrumentation mechanisms at multiple levels
 - Source code
 - manual
 - automatic
 - C, C++, F77/90/95 (Program Database Toolkit (PDT))
 - OpenMP (directive rewriting with Opari)
 - Object code
 - pre-instrumented libraries (e.g., MPI using PMPI)
 - statically-linked and dynamically-loaded (e.g., Python)
 - Executable code
 - dynamic instrumentation (pre-execution) (DynInst)
 - virtual machine instrumentation (e.g., Java using JVMPI)
- Support for performance mapping
- Support for object-oriented and generic programming

TAU: Basic Profile View



TAU: Callgraph Profile View



VI-HPS

V VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

TAU: 3D Profile View



Extra-P

- **Goal**: identification of parts of the program which scaling behavior is unintentionally poor (much worse than expected) by means of automatic performance-modeling
- Supports Linux (x86x86_64/IA64/PPC/Power), Mac OS X (x86_64)
- Accepts input files in the Cube format and processes them into a condensed Cube format containing functions for each metric and call path
- <u>http://www.scalasca.org</u>
- Open Source: BSD 3-Clause License

Interactive exploration of performance models in Extra-P



Darshan

- I/O characterization tool logging parallel application file access
- Summary report provides quick overview of performance issues
- Works on unmodified, optimized executables
- Shows counts of file access operations, times for key operations, histograms of accesses, etc.
- Supports POSIX, MPI-IO, HDF5, PnetCDF, ...
 - Doesn't support mpif90 on BlueGene systems (use mpif77)
- Binary log file written at exit post-processed into PDF report
- http://www.mcs.anl.gov/research/projects/darshan/
- Open Source: installed on many HPC systems

Example Darshan report extract

