

Center for Information Services and High Performance Computing (ZIH)

# Introducing OTF / Vampir / VampirTrace

Zellescher Weg 12 Willers-Bau A115 Tel. +49 351 - 463 - 34049

Robert Henschel (Robert.Henschel@zih.tu-dresden.de)



## Dresden

- Infineon/Qimonda, AMD, etc.
- Public and private funded research institutes
- Evolving field: Molecular Biology and Bioinformatics
- TU Dresden
  - Center for Information Services and High Performance Computing (ZIH)







#### **HPC** Complex



#### **Tools Overview**

#### Tracing:

- Collection of all events of a process / program
- Sorted by time stamp
- VampirTrace
- Trace File Format:
  - Fast and efficient sequential and parallel access
  - Platform independent
  - OTF
- Trace Visualization
  - Parallel/distributed server
  - Lightweight client on local workstation
  - VampirServer and Client





Vamp<mark>ir+Vampi</mark>rtrace 3.0

#### Vampir+Vampirtrace 4.0 / Intel Trace Analyzer 4.0+Intel Trace Collector 4.0

Vampir+Vampirtrace 5.0 Vampir Server+Client 1.0 Intel Trace Analyzer 6.0 Intel Trace Collector 6.0





## **Tracing: OTF Trace Format**

- Open source trace file format
  - Available from the homepage of TU Dresden, ZIH
  - Google for *tu dresden* and *otf*
- Includes powerful *libotf* for use in custom applications
- Actively developed
  - In cooperation with the University of Oregon and Lawrence Livermore National Laboratory





## Tracing: VampirTrace

- Open source trace library
  - Available from the homepage of TU Dresden, ZIH
  - Google for tu dresden and vampirtrace
- Record events from applications
  - Function enter/leave
  - Process creation
  - MPI and OpenMP events
  - Hardware performance counters (PAPI)
- Collect event properties
  - Time stamp
  - Location (process / thread / MPI)
  - MPI specifics like message size etc.





- Parallel/distributed server
  - Runs in (part of) production environment
  - No need to transfer huge traces
  - Parallel I/O
- Lightweight client on local workstation
  - Receive visual content only
  - Already adapted to display resolution
  - Moderate network load
  - Scales to traces >40 GB







۲ 🗖 🔍	Vampir NG - Timeli	ine	)				
	ns-nt1-it	3-proc32-smal	ll_grid.vpt (36.47	77 ms = 1.071 s	= 1.035 s)		
	0,200 s	0.400 s	0.600 s	0.800 s	1,000 s	A	
Process 0 💻	• 1077 0 <mark>; 1</mark> 4		🖬 1064 🔤 🖬 😽	2072 FLUSHTR	ACE	🗧 Flush 🔚	
Process 1	MPB_BCAGT HMPI_A	LLTOALL		<b>115</b>		MPI	
Process 2	MP <del>Q_DCAGT_MMPI_A</del>	LLTOALL		115		MG	
Process 3	MP <del>Q_DCAGT_MMPI_A</del>	<del>llto</del> all	M · · · · · · · · · · · · · · · · · · ·	115		NS	
Process 4	MP <del>Q_DCAGT_MMPI_A</del>	<del>llto</del> all		115			
Process 5	MP <del>M_DCAST MMPI_A</del>	<del>llto</del> all		115			
Process 6	MP <del>Q_DCAGT MMPI_A</del>	<del>llto</del> all	M • M	115			
Process 7	MP <del>Q_DCAGT MMPI_A</del>	<del>llto</del> all		115			
Process 8	MP <del>Q_DCAGT_HMPI_A</del>	<del>llto</del> all	M · · · · · · · · · · · · · · · · · · ·	115			
Process 9	MP <del>Q_DCAGT_MMPI_A</del>	<del>llto</del> all		115			
Process 10	MP <del>Q_DCAGT_MMPI_A</del>	<del>llto</del> all	M · · · · · · · · · · · · · · · · · · ·	115			
Process 11	MP <del>Q_DCAGT_HMPI_A</del>	<del>llto</del> all	M · · · · · · · · · · · · · · · · · · ·	<mark>-115</mark>			
Process 12	MP <del>Q_DCAGT_MMPI_A</del>	<del>llto</del> all	M • M	115			
Process 13	MP <del>Q_DCAGT_HMPI_A</del>	<del>llto</del> all	M • M	115			
Process 14	MPB_BCAGT MIPI_A	<del>llto</del> all		115			
Process 15	MP <del>D_DCAGT_HMPI_A</del>	<del>llto</del> all	M · · · · · · · · · · · · · · · · · · ·	115			
Process 16	MP <del>D_DCAGT MMPI_A</del>	<del>llto</del> all		115			
Process 17	MPB_BCAGT MPI_A	<del>LLTO</del> ALL		115			
Process 18	MP <del>Q_DCAST MIPI_A</del>	<del>llto</del> all	H - HH - O	115			
Process 19	MPB_BCAST MPI_A	<del>LLTO</del> ALL	H · H	115			
		Di	splayed 20 from 3	2 bans			

#### Global Timeline with Thumbnail View







Zoomed







**Robert Henschel** 

**High Performance Computing** 



Process Timeline with MFLOP Counter





👻 🦳 🛋 Vampir N	IG - Summary Chart		👻 🔎 🛋 Vampir NG - Summary Chart
ns-nt1-it3-proc32-s	mall_grid.vpt (0.655 s - 0.	722 s = 67,161 ms)	ns-nt1-it3-proc32-small_grid.vpt (0.655 s - 0.722 s = 67.161 ms)
0,000s 0,50	00 s 1.000 s 1.500	s 2,000 s	0,000s 0,500 s 1,000 s 1,500 s Sum CellGeometryA 0,370 s MPI ISEND 0,172 s
			MPI_WAITALL 0.171 s MPI_ALLREDUCE 0.161 s MPI_BCAST 0.149 s Comals [xxx] 0.140 s
MPI	0.797 s		FLUSHTRACE 0.110 s MPI_BARRIER 89.790 ms WallDistance 779.398 ms ShasR1 [xxx] 75.416 ms CellGeometruB 72.006 ms
MG	0.777 s		ComputeUs   69,974 ms     Flu0d [xxx]   63,173 ms     ComeSL [xxx]   62,154 ms     MainProgram   55,163 ms     MPI_IRECV   52,602 ms
NS 0,465 s			ShasR0 [xxx] 37.412 ms   unit_vector 34.256 ms   Corues [xxx] 19.348 ms   ShasI0 [xxx] 16.973 ms   AdvectiveFlux 13.727 ms
Flush 0,110 s			EvalTimeStep 12.303 ms Smadel [xxx] 11.228 ms DiffusiveFlux 10.416 ms cross 9.936 ms DoStatistics 7.467 ms
NS_custom 0,268 ms		V	Comals [xxx] 7.021 ms Conve1 [xxx] 8.640 ms Priebc [xxx] 6.397 ms
	Displayed 6 from 6 bars	All: Exclusive Times	Displayed 30 from 82 bars pols: Exclusive Times

**Grouped / Comprehensive Function Statistics** 







**Zoomed Message Statistics** 





000 1	/ampir NG - Timeline - < 2 >			Х
	sppm-ompi-128x8.vpt (45,195 s	- 46.365 s = 1.170 s)		
	45.500 s	46.000 s		
process 48	238 <b>123</b> \$mp do \$\$mp do	251123	📕 MPI 🦢	
thread 48:1	238 !\$omp_do	224 226	OMP	
thread 48:2	238 !\$omp_do	224 226	USR	Δ
thread 48:3	238 !\$omp_do	224226	UMP-SYNC	· · · · · · · · · · · · · · · · · · ·
thread 48:4	238 !\$omp_do	224226		
thread 48:5	238 !\$omp_do	<mark>224 2</mark> 26		
thread 48:6	238 !\$omp do	224 226		·····
thread 48:7	238 !\$omp_do	224 226		i in the second s
process 56	238 <b>123</b> \$omp do \$\$omp do			
thread 56:1	238 !\$omp_do	223 226		
thread 56:2	238 !\$omp_do	221 226		
thread 56:3	238 !\$omp_do	221 226		· · · · · · · · · · · · · · · · · · ·
thread 56:4	238 !\$omp_do	221 226		
thread 56:5	238 !\$omp_do	221 226		
thread 56:6	238 !\$omp_do	221 226		
thread 56:7		<b>221</b> 226		
process 64	O Vampir NG - Identified Colle X			
thread 64:1	Participants: Process(s) 32-39	224 226		
thread 64:2	Occurtion + Itom bounion	224 226		
thread 64:3	operation : :>omp Darrier	224 226		
thread 64:4	Communicator: 5			
thread 64:5	Interval : 46.173 s - 46.300 s	224 226		
thread 64:5	Dunction + 0.127 c	224 226		
thread 64:/	Duración : V+127 S	224 226	V	
<				
	Close	024 bars		

#### **OpenMP Barrier Synchronization**





We are currently evaluating how to integrate Vampir and Eclipse

- What features of the Vampir GUI may be useful within Eclipse
- What additional opportunities are created by having trace information and application source code within the same IDE
- Eclipse integration to better support application developers that are using performance analysis tools
  - Easy creation of trace files
  - Configuration of filters and function groups
  - Matching trace file information and source code location





- Displaying OTF trace file information in Eclipse
- Eclipse plug-in that displays OTF information and statistics
- JNI code to access *libotf* from Java applications
- Launching the Vampir GUI from within Eclipse







Test other views of Vampir inside Eclipse

Problems Properties 🗢 OTF Timelin	ne 🛿	i i " 🗖
main	printMessage	doWork





VT\_ macro awareness in Eclipse editor

- Like MPI/OpenMP in PLDT
- For example:
  - Check for matching VT\_USER\_START / VT\_USER\_END blocks
  - Create custom tracing blocks in the source code from selected areas
- Provide a wizard that will help the user to create a build target for tracing
- Provide a wizard to edit the launch configuration for a tracing run
  - Automatically set the VampirTrace specific environment variables
- Provide support to automatically build filters and groups
- Display information from OTF trace files in Eclipse (summary)
- Trace file management for projects
- Design a scaled down version of some Vampir displays for Eclipse
- Link those displays with the source code view of Eclipse





## Questions?





**Robert Henschel** 

Center for Information Services & High Performance Computing