

#### **Performance Analysis with Vampir**



### **Event Trace Visualization with Vampir**

 Visualization of dynamic runtime behaviour at any level of detail along with statistics and performance metrics
 Alternative and supplement to automatic analysis

#### Typical questions that Vampir helps to answer

- What happens in my application execution during a given time in a given process or thread?
- How do the communication patterns of my application execute on a real system?
- Are there any imbalances in computation, I/O or memory usage and how do they affect the parallel execution of my application?

#### Timeline charts

 Application activities and communication along a time axis



#### Summary charts

 Quantitative results for the currently selected time interval



### Visualization Modes (1) Directly on front end or local machine

% vampir



# Visualization Modes (2)

On local machine with remote VampirServer



### **Main Performance Charts of Vampir**

#### **Timeline Charts**



- Master Timeline
- Process Timeline
- Summary Timeline
- Performance Radar
- Counter Data Timeline
- I/O Timeline

- all threads' activities
- single thread's activities
- all threads' function call statistics
- all threads' performance metrics
- single threads' performance metrics
- all threads' I/O activities

#### Summary Charts



Process Summary
 Communication Matrix View
 Call Tree



### Vampir Case Study: Analyzing Load Imbalance in COSMO-SPECS





			14	<b>3</b>	6 📑		ii 🖄	5	3 🍪	*	0 s				578.101 s			578 s
							Timelin	ne						××		Function Sum	mary	
		0 s	50 s	100 s	150 s	200 s	250 s	300 s	350 s	400 s	450 s	500 s	550 s		All Processes, Accumulated	d Exclusive Tim	e per Function Gro	up
Drago		<b>M</b>								-					20,000 s 15,000 s	10,000 s	5,000 s	0 s
Proces	s 0		i internet			mm								×	23,104.034 s			MP
Proces	c 2						mm								18,/83./41 s	44 577 400		MPI
Proces	53	đ		min		mm	min									11,577.183	1550 646 s	MP_UTIL Application
Proces	is 4	Ő		00000		mm											1,290,593 s	COUPLE
Proces	s 5	O															993.146 s	METEO
Proces	is 6	O		DDDDDD													4.366	VT_API
Proces	is 7	O	COCCO	DIDIDI	CCCCCCCC	1000000	100000	0000000										
Proces	is 8	01	<b>COCCO</b>	DDDDD	CCCCCCCCC		0000000											
Proces	is 9	01	00000	DDDDDD	DDDDDDDD		DEBER					8888888				Context Vie	w	
Proces	is 10	0		DDDDDD	CCCCCCCCC	000000		00000000				8888888			<b>=</b> ×	i Trace		7 +
Proces	is 11	0	<b>DDDDD</b>	DDDDDD	<b>CCCCCCCC</b>	DDDDDD	100000	DDDDDDDD		899999		8888888			Property	Value		
Proces	is 12	O		DDDDDD	<b>CCCCCCCC</b>	<b>DDDDDD</b>	0000000	0000000							File	/Licoro/mouro		mair Tutorial
Proces	is 13	Q	$\mathbf{m}$	DDDDDD	<b>DECECCE</b>	<b>DEDEE</b> E	CCCCCCCC	DEBERE	<b>DDDDDD</b>	888888		9909090	00000000		File	/Users/mawe	5 11 0	inpir-rutonai
Proces	s 14	0	mm	COCOCO	<b>CCCCCCCC</b>		<b>COCCECC</b>	<b>DEBESSE</b>	0000000	000000	00000000				Creator	VampirTrace	5.11.2	
Proces	is 15	0	ונכנכנ	DDDDDD	DDDDDDDD	DDDDDD	<u></u>	0000000	0000000			000000			Version	1.11 alpha		
Proces	is 16	0	ננננננ	DDDDD	<b>BEEEEE</b>		000000	0000000	0000000		000000	000000	0000000		Number of Processes	100		
Proces	is 17	0		DDDDDD	<b>BDDDDDD</b>	000000	<b>BBBBB</b> B		0000000		8888888				Timer Resolution	50 ns		
Proces	is 18	0	000000	<b>DDDDDD</b>	DDDDDDDD	<b>DDDDDD</b>	DDDDDD			888888	9999999	8008081			Trace Times			
Proces	is 19	0	פכככי	DDDDDD	DDDDDDDD	DDDDDD	0000000	88888888							Start	Thu Nov 31	2:36:57 2011 (1320	32021776169
Proces	is 20	0	nnn	mm	<b>CCCCCCCC</b>	<b>DDDDDD</b>	0000000	0000000							Stop	Thu Nov 31	2:46:30 2011 (1320	3207909206
Proces	is 21	0						0000000				000000			Flansed	00:09:33 (57	(1020	
Proces	is 22														VampirTrace Environment	00.00.00 (0)	0.00000)	
Proces	is 23															70.005		
Proces	s 24														VI_MODE	TRACE		
Proces	5 25														VT_BUFFER_SIZE	200M		
Proces	5 20														VT_SYNC_FLUSH	no		
Proces	5 2/												mmmm		VT_SYNC_FLUSH_LEVEL	. 80		
Proces	s 20														VT_MAX_FLUSHES	1		
Proces	s 20		mm	mm				mm							VT_METRICS	PAPI_FP_OPS	6	
FIDCES	3 30				1000										VT RUSAGE	<not set=""></not>		
					170	8	1000	2		2			1		noonor			

- Weather forecast code
  COSMO-SPECS
- Run with 100 processes
- COSMO: weather model (METEO group)
- SPECS: microphysics for accurate cloud calculation (MP and MP\_UTIL group)
- Coupling of both models done in COUPLE group

1000	Section.		K HAL		11 III III		<b>1</b>	3 40	119	•			578.101 s
	0.0	50 0	100 a	150 a	200 a	Timelin	e 200 e	250 a	400 a	450 a	500 a	550 a	Function Summary
	US	50 5	100 5	150 \$	200 s	250 \$	300 S	350 S	400 S	450 S	500 s	550 S	All Processes, Accumulated Exclusive Time per Function Group
ess 0													20,000 s 10,000 s 10,000 s 0,000 s 0 s
ess 4													18 783 741 s (32 78%)
ess o ess 12													11 577 183 s (20 2%)
ess 16	-												1 550 646 s (2 71%)
ess 20													1 290 593 s (2 25%)
ess 24	-												993 146 s (1 73%)
ess 28	- 22												4 366 s (0.01%) VT
ess 32													
ess 36													
555 4U													
ess 44	-					21 <b>   </b>	<b></b>						Increasing MPI time
ess 52	-												increasing in i time
ess 56	-		H										7
ess 60	-	16.34											
ess 64	-												
ess 68												/	Function Summary
ess 72		i in i i	Sul du			m Et						/	All Processes, Accumulated Exclusive Time per Function
255 70	-										/		20,000 s 10,000 s 0 s
ess 84	-										/		23,104.034 s (40.32%) organize_sbm
ess 88	-		s of the state								7		13,312.738 s (23.23%) MPI_Recv
ess 92	-						E. K						8,009.115 s (13.98%) sbm_exchange_mete
ess 96	-												5,000.233 s (8.73%) MPI_Wait
										7 /			1,845.621 s (3.22%) sbm_advection_pd_h
ocesses	, Exclu	sive Time	per Functi	on Group									1,722.447 s (3.01%) sbm_remove_mete
100	%												1,550.646 s (2.71%) SBM_PREPARE
1000					MP_UTIL								1,290.593 s (2.25%) met_set
80	%												746.765 s (1.3%)
60	%								MPI				246.356 s (0.43%) org_runge_kutta
													229.582 s (0.4%) MPI_Isend
40	%					-						- 18 L	179.277 s (0.31%) MPI_Allreduce
20	%				N	1P							55.964 s (0.1%) MPI_Init
		-	-		_	_	-						4.366 s (0.01%) sync time
	0/	a land the second second second		Contraction of the local division of the loc	Concession of the local division of the loca	and the owner where the party is not	Contraction of the local division of the loc					_	4.212 s (0.01%) MPI_Bcast
0	/0			:	:								

- Compared to METEO, MP and MP\_UTIL are very compute intensive, however this is due to more complex calculations and no performance issue
   Problem: >32% of time
- spent in MPI
- MPI runtime share increases throughout the application run



- Zoom into the first three iterations
- MP/MP\_UTIL perform four sub-steps in one iteration
- Low MPI time share
- Everything is balanced and looks okay



- Zoom into the last three iterations
- Very high MPI time share
  (>50%)
- Large load imbalance caused by MP functions around Process 54 and Process 64



- **PAPI\_FP\_OPS** counter showing higher FLOPs rates on processes causing the imbalance
- Reason for imbalance: Static grid used for distribution of processes.
   Depending on the weather, expensive cloud computations (MP group) may be only necessary on some processes



- Process Summary helps finding outliers
- Groups processes by their behavior (similar call/duration profile)
- Number of expected groups is variable
- In this case 4 yields the best results



## Vampir Showcase: Analyzing Multilayer File I/O Applications





### **Multilayer File I/O Application**



- IO bandwidth benchmark
  b\_eff\_io
- Measures achievable I/O bandwidth of parallel MPI-I/O applications
- Shared Resource Timeline offers a per file and per thread view on File I/O operations

### **Multilayer File I/O Application**



 Visualization of I/O on multiple layers (MPI & POSIX)

Example: behavior of MPI\_File\_open

- Internally uses POSIX
  open for opening the actual file on disk
- Multiple consecutive calls to open and close on master rank



## Vampir Showcase: Analyzing CUDA Applications





#### **CUDA Application**



- Material science code LSMS
- CUDA is utilized for heavy computations
- CUDA streams are child's of the owning Process
- Allows an in-depth analysis of host-device communication

#### **CUDA Application**



- Communication Matrix best for analyzing the general communication pattern
- Expectation: balanced communication, represented by a symmetric matrix
- Problem: communication
  with stream 7 is different

#### **CUDA Application**



- Shared Resource Timeline offers a per device view on kernel executions
- Best suited for analyzing multi GPU per node scenarios
- Allows a dedicated analysis of kernel execution patterns
- Yields insights of the actual hardware usage

## VI-HPS

VIRTUAL INSTITUTE – HIGH PRODUCTIVITY SUPERCOMPUTING

## Summary

#### Vampir

- Interactive trace visualization and analysis of:
  - MPI, OpenMP, CUDA applications
  - File I/O
  - Hardware performance counters
  - (Collective) communication
- Intuitive browsing and zooming
- Available for Linux, Windows, and macOS
- VampirServer
  - Scalable to large trace data sizes (20 TiByte)
  - Scalable to high parallelism (200,000 processes)

