Intel Trace Analyzer / Collector Introduction and Tutorial

Ullrich Becker-Lemgau Intel GmbH, Solutions & Software Group ullrich.becker-lemgau@intel.com

> October 18, 2004 Istanbul Technical University Istanbul, Turkey





Intel Trace Analyzer / Collector

scalable performance analysis and visualization for parallel MPI programs



Agenda

- Introduction to Intel Trace Analyzer (ITA)
 - concept, GUI, examples traces, BKMs
- Introduction to Intel Trace Collector (ITC)
 - concept, compiling, linking
 - default tracing
 - recording names of functions or regions
 - recording source-code-locations
 - recording performance counter data
 - selective tracing, filtering
 - handling structured tracefiles
- Using ITC and ITA with a simple application.



Visualization and Analysis of MPI Programs

ITA is the visualization and analysis tool for ITC







Introduction to Intel Trace Analyzer / Collector

- What is the Intel Trace Analyzer/Collector?
 - It is a tool that focuses on instrumentation for two parallel programming paradigms for High Performance Computing (HPC), namely:
 - OpenMP
 - MPI
- ITA/ITC provides the following features:
 - Ease of Use basic how-to's, to apply to your development
 - Integration from default performance analysis to mastering the workbench, e.g. instrumenting functions, extracting hardware monitors, detailing the runtime environment
 - Scalability works with 10,000+ Processors, how to manage more or less performance data



Intel Trace Analyzer/Collector Features

- Offline trace analysis for MPI (and others ...)
- Traces generated by Intel Trace Collector tool (`ld ... -IVT -Ipmpi -Impi`)
- Scalability in time and processor—space (STF)
- High-performance graphics, excellent zooming and filtering
- Display and analysis of MPI and application events:
 - execution of MPI routines, p2p, and collectives
 - MPI–2 I/O operations
 - performance counter displays (events, PAPI, ...)
 - execution of application subroutines (optional, gnu-func.)
- Easy customization



Intel Trace Analyzer/Collector Components and Interaction

- Compiler
 - MPI, ...
 - ITC API
- Intel Trace Collector
 - Data collection
 - Run-time configuration
- Intel Trace Analyzer
 - MPI analysis







Intel Trace Analyzer Main Window

Intel Trace Analyzer main window:

🔜 Intel(R) Trace Analyzer 4.0									
File	<u>G</u> lobal Displays	Process Displays	Filter <u>s</u>	P <u>r</u> eferences	Extras	<u>H</u> elp			
redblad	redblack_sndrcv.stf								

- Tracefile loading can be interrupted at any time
- Tracefile loading can be resumed
- Tracefile can be loaded starting at a specified time offset
- Tracefile can be re—written (re—grouped symbols)



Intel Trace Analyzer Displays

- Global displays show all selected processes
 - Summary Chart: aggregated profiling information
 - Timeline: detailed application execution over time axis
 - Communication statistics: message statistics for each process pair
 - Activity Chart: presents per-process profiling information
 - Global Comm. Statistics: collective operations statistics
 - I/O Statistics: MPI I/O operation statistics
 - Calling Tree: draws global or local dynamic calling trees
- Process displays show a single process per window
 - Activity Chart
 - Timeline
 - Calling Tree



Summary Chart





- Aggregated profiling information
 - execution time
 - number of calls
- Inclusive or exclusive of called routines
- Look at all/any category or all states
- Values can be exported/imported
- Tracefiles can be compared

Timeline Display



- Now displays MPI collective and I/O operations
- To zoom, draw rectangle with the mouse
- Also used to select sub-intervals for statistics





Timeline Display (Message Ir message line



© Copyright 2002-2003 Intel Corporation

Communication Statistics

🐵 🕈			Intel(R)	Trace Analyz	er 4.0 - Messa	ge Statistics			• 🗆 ×	1						
			psolid.x.	stf (Sum. Leng	th, 1.0 us - 1:	48,832 = 1:48.	832)									
	Process 0	Process 1	Process 2	Process 3	Process 4	Process 5	Process 6	Process 7								
Process 0-		17,859 M	11,391 M	12,967 M	24,485 M	12.064 M	11.937 M	12,302 M	25.000 M 24.000 M 23.000 M	Intel/F) Trace Analyz	or 4.0 - Mossa	no Statistics			• n x)
Process 1-	7.713 M		8,266 M	20.547 K	14,316 M	1,758 K	51.25 K	264	22.000 M 21.000 M 20.000 M	psolid.x Process 2	stf (Max. Leng Process 3	th, 1.0 us - 1: Process 4	48.832 = 1:48. Process 5	832) Process 6	Process 7	1.000 M
Process 2-	111.379 K	7.658 M		11,834 M	14,009 M	276	1.77 K	5.156 K	19.000 M 18.000 M 17.000 M	1.0 M	1.0 M	1.0 M	1.0 M	1.0 H	1.0 M	0.938 M 0.875 M
Process 3	41.551 K	27.711 K	11,579 M		25.554 M	57.781 K	5₊246 K	11.633 K	16,000 M 15,000 M 14,000 M	82,902 K	172 184,176 K	915,918 K 896,207 K	172 172	192 172	172 172	0,812 M 0,750 M 0,688 M
Process 4-	42.055 K	5,227 K	5,227 K	12,111 M		7.817 M	5 , 977 K	10,113 K	13,000 M 12,000 M 11,000 M 10,000 M	150,285 K		953.488 K	192	172	172	0.625 M 0.562 M
Process 5-	40,234 K	1.895 K	404	171.066 K	20,635 M		10,788 M	10,918 K	9.000 M 8.000 M 7.000 M	172	91.055 K		79,293 K	172	172	0.500 M 0.438 M 0.375 M
Process 6-	41.109 K	188.93 K	1.898 K	5.285 K	13.892 M	15.544 M		6.08 M	6.000 M 5.000 M 4.000 M	172	240	904.645 K		71,559 K	172	0.312 M 0.250 M
Process 7.	40.84 K	1.0 K	5,137 K	11.523 K	14.105 M	10.023 K	7,408 M		3.000 M 2.000 M 1.000 M	172	172	902.09 K	238,176 К 172	 91.762 К	39,281 к	0,188 M 0,125 M 0.062 M

- Message statistics for each process pair:
 - Byte and message count
 - min/max/avg message length
 - min/max/avg bandwidth
- Filter for message tags or communicators





Intel Trace Collector

Tracing of MPI and Application Events

> Intel Trace Collector is an event-based tracing library for program analysis







Counter Displays

 Intel Trace Analyzer Counter Timeline Display, e.g. using PAPI





The Structured Tracefile Format – Concept

- Support *realistic* applications and configurations
- Improve trace file loading and navigation
- Subdivide trace into *frames*
- Compute statistics for each frame
- Stripe data across multiple files
- Index file ties everything together





The Structured Tracefile Format – Implementation

- Components of a structured trace
 - one declaration, frame, index and statistics file
 - one data file per process (group)
 - one anchor file per data file
- Single-STF
 format packs all components
 into one file







Intel Trace Analyzer / Collector Tutorial





Installing Intel Trace Analyzer/Collector

- create directories 'ITA' and 'ITC'
- unpack tar-files in directories
- run install in both directories
- add '~/ITA/bin' to PATH
- set PAL_LICENSEFILE





Intel Trace Analyzer – MPI Performance Analysis

- Invoke Intel Trace Analyzer: 'traceanalyzer'
- Select File → Open tracefile 'redblack_icomm.stf'





Intel Trace Analyzer – Looking at Frame Information

 e.g. Click on Frame and select Context Menu⇒Frame Info



© Copyright 2002-2003 Intel Corporation



Intel Trace Analyzer – Looking at Frame Information

Zoom to see more detail







Intel Trace Analyzer – Precomputed STF Statistics

- Context Menu⇒Open Thumbnail
- Click on Frame





Intel Trace Analyzer – Precomputed STF Statistics

Context Menu⇒Display⇒Symbol Statistics
 ⇒Min + Avg + Max

🔹 🕈 Trace A	Analyze	r 4.0 -	Thum	bnail I 🗖	o x
redblack_icom	n₊stf (S	ymbol 🛛	Statis	tics / Fr	rame #2)
	0 1	000	2000	3000	
MPI_Recv				3200	
MPI_Isend				3200	
MPI_Waitall		1600	i	i	
MPI_Allreduce	800	-			
Exchangeblack	800				
Exchangered	800				
MPI_Wtime	46	ł			
Red-Black	β	i -	i.		
Initialisation	β	i i	i	i	
User_Code	β	1			
MPI_Bcast	8	-			
MPI_Comm_split	в	-	-		
MPI_Finalize	B	i i			
MPI_Comm_size	β	i	í		
MPI_Comm_rank	β	-			$\overline{\mathbf{\nabla}}$
\triangleleft					

🔹 🕈 Trace A	nalyzer 4.0 - Thumbn	ail I 🗉 🗶 🗙
redblack_icomm	.stf (Symbol Statistic	s / Frame #2)
0.0)s 0,188 s 0,375 s	s 🔼
User_Code	0,259 s 0,43	88 s
MPI_Bcast	0,275 s	
MPI_Comm_split	0,257 s	
Red-Black	40.692 ¢s ¦	
MPI_Allreduce	0.828 ms ¦	
Initialisation	0.579 ms ¦	
MPI_Recv	0.505 ms	
Exchangered	0₊28 ms	
Exchangeblack	P.22 ms	
MPI_Isend	0,2 ms ¦	
MPI_Finalize	0.135 ms ¦	
MPI_Comm_rank	0.125 ms	
MPI_Waitall	0.118 ms	
MPI_Wtime	36.693 us	
MPI_Comm_size	25.987 us	
\leq		





Intel Trace Analyzer – Precomputed STF Statistics

 Context Menu⇒Display⇒Sent Message Statistics⇒Count







Intel Trace Analyzer – Loading Event Data

Select frames by

- Context Menu⇒Select/Deselect Frames and draw rectangle (toggles selection status)
- Context Menu⇒Select All Frames
- Load by Context Menu⇒Load Selected Frames
- First "regular" Intel Trace Analyzer display pops up
- Configure with Preferences⇒Displays⇒Startup with





Intel Trace Analyzer – Loading Event Data

Configure the Summary Chart









Intel Trace Analyzer – Process Grouping and Filtering

- Select Filters Processes
- Look at list of defined groups





High-Performance C

Intel Trace Analyzer – Routine Profiles and Statistics

- Setup Summary Chart
 - Global Displays⇒
 Summary Chart
 - Context Menu⇒
 Select⇒All Symbols
 - Context Menu⇒
 Options⇒Per Process
- Selected group: All Master Threads

💠 🕂 l) Trace	Analyzer 4.0 - Summary 🔹 🗖 🗙
.ack_icomm.stf	(Times, 13,333 ns-0,698 s) (Per Prod
Sum	0.599 s
User_Code	0,282 s
MPI_Comm_split	0.113 s
MPI_Bcast	58.677 ms
Red-Black	_ 41.829 փs
MPI_Allreduce	28,544 ms
MPI_Isend	18.905 ms
Exchangered	17.089 m\$
Exchangeblack	16.62 ms
MPI_Recv	14.428 ms
MPI_Waitall	6.396 ms
Initialisation	0.367 ms
MPI_Comm_rank	0.117 ms
MPI_Finalize	0.102 ms
MPI_Wtime	62,363 us
MPI_Comm_size	25.98 us
	0.2 s 0.4 s 0.6 s





- Setup Process Profile
 - Global Displays⇒Process Profile







Context Menu ⇒
 Select ⇒ MPI







Context Menu⇒
 Sort by⇒Value Up

÷ +	Intel(R)	Trace Analyzer	4.0 - Process	Profile		×
	redblack_ico	omm.stf (Average,	13,333 ns-0,69	98 s = 0,698 s)		
0.0 s	0+1 ms	0,2 ms	0,3 ms	0.4 ms	0.5 ms	
					Activity: MPI	Ţ
						ſ





- Zoom to identify
- min/max load







Intel Trace Analyzer – Message Statistics

Setup Message Statistics

- Global Displays⇒Message Statistics
- Context Menu⇒Display⇒Avg. Rate







Intel Trace Analyzer – Message Statistics

Zoom to read min/max values

4 +		Inte	l(R) Trace A	nalyzer 4.0	Message S	itatistics			• 🗆 X	1
		redblack,	_icomm.stf (Avg. Rate, 1	3.333 ns - 0	.698 s = 0.69	98 s)			
	Process 0	Process 1	Process 2	Process 3	Process 4	Process 5	Process 6			
Process 0 -		1,321 e6							8,40 e6 8,10 e6 7,80 e6	
Process 1 -	8.472 e6		1,537 e6						7.50 e6 7.20 e6 6.90 e6	
Process 2 -		4,708 e6		1.772 e6					6.60 e6 6.30 e6 6.00 e6	
Process 3 -			3₊53 e6		2,222 e6				5.70 e6 5.40 e6 5.10 e6	
Process 4 -				2,458 e6		1,575 e6			4.80 e6 4.50 e6 4.20 e6	
Process 5 -					4.411 e6		1.737 e6		3.90 e6 3.60 e6 3.30 e6	
Process 6 -						3.263 e6			3.00 e6 2.70 e6 2.40 e6	
Process 7 -							4.806 e6	∇	2.10 e6 1.80 e6 1.50 e6	





Intel Trace Analyzer – Message Statistics

Select All Nodes

😔 +	Intel(R) Trac	ce Analyzer 4.0 - I	Message Statistic	S	• 🗆 X
	redblack_icomm.st	tf (Avg. Rate, 13.	333 ns - 0,698 s	= 0.698 s)	
tros	tV44.pacific.lini	gov tros	tV61.pacific.lini	+90V	
					4.80 e6
frost044 pacific llpl oov-	3 093 e6	1 535 e6			4.60 e6
11 00000000 pacin 10+11111+300	0,000 00	1,000 00			4.40 e6
					4.20 e6
					4.00 e6
		2.836 e6			3,80 e6
frost045.pacific.llnl.gov	4.103 e6		1,46 e6		3.60 e6
					3.40 e6
					3,20 e6
					3.00 e6
frostOF1 pacific llpl cou		4,393 e6	2.752 e6	1 42 66	2,80 e6
n oscoor.pacinic.rini.gov				1,42 60	2,60 e6
					2,40 e6
					2,20 e6
					2,00 e6
frost063.pacific.llnl.gov	.llnl.gov		4,928 e6	3.017 e6	1.80 e6
					1,60 e6





Intel Trace Analyzer – Timeline Displays

- Open two Timelines by
 - Global Displays⇒Timeline
 - Global Displays⇒Summary Timeline
 - Zoom



Intel Trace Analyzer – Activity Timeline

4 ÷		In	tel(R) Trace Analyz	er 4.0 - Timeli	ine		• • ×
		redbla	ck_icomm.stf (13.33	3 ns - 0,698 s	:= 0,698 s)		
	0,1	s 0,2 s	0. <u>3</u> s	0.4 s	0.5 s	0.6 s	
Process 0	User_Code		MPI_Comm_s	split			MPI
Process 1	User_Code		MPI_Comm_:	split			Application
Process 2	User_Code				<u>241</u>		Lalculation
Process 3	User_Code		MPI_Comm_s	split			
Process 4	User_Code		248 MPI_Comm_:	split			
Process 5	User_Code		248 MPI_Comm_s	split			
Process 6			User_Code		24	1	
Process /			Useri_Code				
Process 8	User_Code				248		
Process 9			UserLuode				
Process 10	User_Lode		MPI_Beast	1			
Process 11	User_Code		248 MP1_Comm_:	split			
Process 12			User_Code		24)		
Process 13	User_Lode		248MP1_Comm_:	split			
Process 14	User_Lode	1	248 MP1_LOMM_:	split	0.4		
Process 15			User_Lode		24	<u>1</u>	
Process 15			User_Lode				
Process 1/	User_Lode		MPI_Beast		24	1	
Process 18	User_Lode		MP1_BCast	1.11	24	<mark>⊥</mark>	
Process 19	User_Lode		248 MPI_LOMM_:	split			
Process 20	User_Lode		MP1_LOMM_:	split	24	4	
Process 21	User_Lode	1			<u>24</u>	<u>⊥</u>	
Process 22			User_Lode	1.1.1			
Process 23	User_Lode		248 MP1_LOMM_S	split			
Process 24	User_Lode		248091_LOMM_3	sp11t			
Process 25	user_Lode		248 "P1_LOMM_:	split			
Process 26	University of the second						
Process 2/	User_Lode		MDI Deset				
Process 28	User_Lode		MDI Deet				
Process 29	User_code		MF1_bCast				
Process 30			User_Code				
Process 51	1	I	user_code	1			

Intel Trace Analyzer – Summary Timeline







Intel Trace Analyzer – Summary Timeline

Select All Nodes





*Other brands and names are the property of their respective owners © Copyright 2002-2003 Intel Corporation



Intel Trace Analyzer – Zooming and Linked Displays

🤣 +			Intel(R) Trace Analyzer 4	.0 - Timeline			• 🗆 ×
		n	edblack_icomm.stf (13.333 ns	: - 0,698 s = 0,698	s)		
	0,1	.s 0,2	2s 0 <u>,3</u> s	0.4 s 0.5	s	0.6 s	
Process 0	User_Code		MPI_Comm_spli	t			M PI
Process 1	User_Code		MPI_Comm_spli	t			Application
Process 2	User_Code				241	1	Calculation
Process 3	User_Code		MPI_Comm_spli	t			
Process 4	User_Code		248 MPI_Comm_spli	t			
Process 5	User_Code		248 MPI_Comm_spli	t			
Process 6			User_Dode		241		
Process 7			User_Code				
Process 8	User_Code				248	1	
Process 9			User_Code				
Process 10	User_Code		MPI_Bcast				
Process 11	User_Code		248 MPI_Comm_spli	t			
Process 12			User_Code		241		
Process 13	User_Code		248MPI_Comm_spli	t			
Process 14	User_Code		248 MPI_Comm_spli	t			
Process 15			User_Code		241		
Process 16			Us¢r_Code				
Process 17	User_Code		MPI_Bcast		241	1	
Process 18	User_Code		MPI_Bcast		241	1	
Process 19	User_Code		248 MPI_Comm_spli	t			
Process 20	User_Code		MPI_Comm_spli	t			
Process 21	User_Code				241		
Process 22			User_Code				
Process 23	User_Code		248 MPI_Comm_spli	t		1	
Process 24	User_Code		248MPI_Comm_spli	t			
Process 25	User_Code		248 MPI_Comm_spli	t			
Process 26			User_Code				
Process 27	User_Code		MPI_Bcast				
Process 28	User_Code		MPI_Bcast				
Process 29	User_Code		MPI_Bcast				
Process 30			User_Code				
Process 31			i User_Code				
\leq							



Intel Trace Analyzer – Zooming and Linked Displays

- To zoom on a display, press the left mouse button at the start of the region to be magnified
- While holding the mouse button, drag the mouse to the end of the desired magnification region, which will be shown by Intel Trace Analyzer with a rectangular rubber-band, and then release the left mouse button
- Zooming can be done in an unlimited depth
- The reverse action, Undo Zoom, works hierarchical and can be invoked with the hotkey U (caseinsensitive) or from the context menu





Intel Trace Analyzer – Zooming and Linked Displays

• The result of zooming from the previous panel display



Product Key Features and Benefits

Event-based tracing tool Accurate and detailed information of serial and parallel program runs

Low overhead structured trace file format (STF) Designed from the ground up for scalability and compact data representation. It can be written in parallel and allows random access to parts of a trace.

Provides a convenient way to graphically analyze runtime event traces produced by MPI and other applications Enables the user to quickly focus at the appropriate level of detail.





Product Key Features and Benefits

Variety of graphical displays Presents important aspects of the application runtime behavior in detailed and aggregate views.

Timeline views and parallelism display The timeline display visualizes the concurrent behavior of parallel applications and statistics can be calculated on demand for certain time intervals and specific processes.

Communication statistics Communication metrics for an arbitrary time interval and the message-length distribution.





Product Key Features and Benefits

Execution statistics

Display subroutine execution metrics for an arbitrary time interval and shows call-tree information.

Easy to use API

Provide the user with a flexible and easy to use interface to instrument the user code.

Thread-safe

Allows tracing of multithreaded applications.





Tutorial – Task0: setup environment

- objective: getting started
- places and paths for compile, link, execute, license keys, doc, etc.





Tutorial – Task1: start Intel Trace Analyzer

- objective: basic use of Intel Trace Analyzer
- this directory contains
 jacobic.stf.single.gz
 which is an example trace file.





Tutorial – Task2: basic tracefile generation

objective: basic use of Intel Trace Collector, how to

generate a tracefile.

this directory contains





Tutorial – Task3: basic API use to instrument function

objective: basic use of Intel Trace Collector API, how to instrument a function

this directory contains





Tutorial – Task4: instrument function with source code location

objective: basic use of Intel Trace Collector API, how to instrument a function with source code location

this directory contains





Tutorial – Task5: record custom performance counter

objective: basic use of Intel Trace Collector API, how to instrument a custom performance counter

this directory contains





Tutorial – Task6: selective tracing and filtering

objective: basic use of Intel Trace Collector API, switch recording off and on, basic use of a VT_CONFIG configuration file

this directory contains





Tutorial – Task7: convert data to single trace file

objective: learn about stftool, use single file trace archives to simplify transfer of storage

