

Analysis report examination with CUBE

Marc Schlütter Jülich Supercomputing Centre





CUBE

Parallel program analysis report exploration tools

- Libraries for XML report reading & writing
- Algebra utilities for report processing
- GUI for interactive analysis exploration
 - requires Qt4

Originally developed as part of Scalasca toolset

Now available as a separate component

- Can be installed independently of Score-P, e.g., on laptop or desktop
- Latest release: CUBE 4.3.0 (January 2015)

CUBE 4.3.1 (April 2015)

Analysis presentation and exploration

- Representation of values (severity matrix) on three hierarchical axes
 - Performance property (metric)
 - Call path (program location)
 - System location (process/thread)
- Three coupled tree browsers
- CUBE displays severities
 - As value: for precise comparison
 - As colour: for easy identification of hotspots
 - Inclusive value when closed & exclusive value when expanded
 - Customizable via display modes





WIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Analysis presentation



Analysis report exploration (opening view)

Absolute	~	Absolute	~	Absolute	~
Metric tree		💽 Call tree 🔲 Flat view		🔚 System tree 🚺 Box Plot	
 1.63e9 Visits 767.48 Time 0.00 Minimum Inclus 48.58 Maximum Incl 5.27e8 bytes_sent 5.27e8 bytes_receive 	sive Time usive Time ed		Ξ	I.63e9 generic cluster I.64e9 gener	₹
	>	K	>		>

VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Metric selection



Expanding the system tree



WIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Expanding the call tree



Inclusive vs. Exclusive values

- Inclusive
 - Information of all sub-elements aggregated into single value
- Exclusive
 - Information cannot be subdivided further



VICTOR CONFIDENCE AND A CONFIDENCE AND A

Selecting a call path



VIAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Source-code view via context menu



Shows the source code of the clicked item

Source-code view

	/home/geimer/Proje	cts/Tests/NPB3.3-MZ-MF	PI/BT-MZ/solve_subs.f	×
subroutine binvcrhs(l	ns,c,r)			
 2				
]]]				
implicit none				= =
double precision pivot, dimension lhs(5,5) double precision c(5,5	, coeff, lhs), r(5)			
 ; 				
pivot = 1.00d0/lhs(1,1) lhs(1,2) = lhs(1,2)*pivot $lhs(1,3) = lhs(1,3)*pivotlhs(1,4) = lhs(1,4)*pivot$) ot ot			
lhs(1,5) = lhs(1,5)*pivotc(1,1) = c(1,1)*pivotc(1,2) = c(1,2)*pivotc(1,3) = c(1,3)*pivotc(1,4) = c(1,4)*pivot	ot			
Read only	Save	Save as	Font	Close

VI-HPS

WIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Flat profile view



VI-HPS

VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Box plot view



VICTOR COMPUTING

Alternative display modes



Important display modes

- Absolute
 - Absolute value shown in seconds/bytes/counts
- Selection percent
 - Value shown as percentage w.r.t. the selected node "on the left" (metric/call path)
- Peer percent (system tree only)
 - Value shown as percentage relative to the maximum peer value

WIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Multiple selection



Derived metrics in Cube

- Value of the derived metric is not stored, but calculated on-the-fly
- One defines an CubePL expression, e.g.:

metric::time(i)/metric::visits(e)

- Types of derived metrics:
 - **Prederived**: evaluation of the CubePL expression is done before the aggregation
 - **Postderived**: evaluation of the CubePL expression is performed after the aggregation
- Examples:
 - "Average execution time" Postderived metric with an expression:

metric::time(i)/metric::visits(e)

"Number of FLOP per second" Postderived metric with an expression:

metric::FLOP()/metric::time()

VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Derived metrics in Cube GUI



 \checkmark

VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Example derived metric FLOPS based on PAPI_FP_OPS and time

	Cub	e-4.3.1: scorep_8x4_sum/profile.cubex (on froggy1)	_ _ x		
	<u>File D</u> isplay <u>P</u> lugins <u>H</u> elp				
	Restore Setting Save Settings				
Edit metric FLOPS (on froggyl)	Absolute	Absolute	Absolute		
Select metric from collection : please select	Metric tree	Call tree Flat view	System tree Barplot Heatmap Image: Bos ≤ Image: Im		
Derived metric type : Postderived metric	■ 1148.49 Time (sec)		□ - node frog6 □ - MPI Bank 0		
Display name : FLOPS	■ 41.57 Maximum Inclusive Time (■ 1.17e9 Master thread		
Unique name : flops	□ 0 bytes put (bytes)	■ 7.39e5 zone setup	9.43e8 OMP thread 1		
Data type : DOUBLE	🗆 🗆 0 bytes get (bytes)	■ ■ 9.31e5 map zones	-■ 9.47e8 OMP thread 2		
Unit of measurement :	■ 5.75e12 PAPI_TOT_INS (#)	■ 9.39e4 zone_starts_	9.47e8 OMP thread 3		
URL :	■ 2.69e12 PAPI_TOT_CYC (#)	-■ 6.16e5 set_constants_	🖻 🗆 - MPI Rank 1		
Description :	■ 2.12e12 PAPI_FP_OPS (#)	🗉 🖻 5.91e8 initialize_	■ 1.17e9 Master thread		
	■ 3.12e9 bytes_sent (bytes)	□ □ 0.00 exact_rhs_	-■ 9.87e8 OMP thread 1		
	3.12e9 bytes_received (bytes)	🖻 🖬 145.62 !\$omp parallel @exac	■ 9.68e8 OMP thread 2		
	■ 1.84e9 FLOPS	⊕ ■ 2.54e4 !\$omp do @exact_r	9.72e8 OMP thread 3		
		□ 9.65e8 !\$omp do @exact_r	□ P P Rank 2		
		■ 9.62e8 !\$omp do @exact_r	■ 1.10e9 Master thread		
		■ ■ 8.14e8 !\$omp do @exact_r	8.9768 OMP thread 1		
		■ 1.21e5 !\$omp do @exact_r	8.77e8 OMP thread 2		
metric::PAPI_FP_OPS()/metric::time()		□ 0.00 !\$0mp implicit barrier			
		0.23e4 excn_qpc_	□ - MPI Kalik 3		
		■ 1.9469 dul_			
		\square 2.19e5 MFI_balliel \square \square 1.92e9 < bt iter>> (200 iters	9.00e8 OMP thread 2		
		□ □ 1.9269 < 1.9269 < 1.0260 (200 ftera □ □ □ 0.000 ftera □ 0.0000 ftera □ 0.000 ftera □ 0.0000 ftera □ 0	9.04e0 OMP thread 2		
		□ 1 05e5 MPL Reduce			
Edit <u>m</u> etric <u>C</u> ancel	7 7		All (32 elements)		
Share this metric with SCALASCA group	0.00 1.84e9 (100.00%) 1.84e	9 0.00 9.65e8 (-0.00%) -12858016489314434.00	0.00179769313486231570814527423731704356798070		
	Selected "!\$omp do @exact_rhs.f:46"		0		

VICTOR COMPUTING

Context-sensitive help



CUBE algebra utilities

Extracting solver sub-tree from analysis report

% cube_cut -r '<<ITERATION>>' scorep_bt-mz_B_8x8_sum/profile.cubex
Writing cut.cubex... done.

Calculating difference of two reports

% cube_diff scorep_bt-mz_B_8x8_sum/profile.cubex cut.cubex
Writing diff.cubex... done.

- Additional utilities for merging, calculating mean, etc.
- Default output of cube_utility is a new report utility.cubex
- Further utilities for report scoring & statistics
- Run utility with "-h" (or no arguments) for brief usage info

Loop Unrolling

- Show time dependent behavior by unrolling iterations
- Preparations:
 - Mark loops by using Score-P user instrumentation in your source code

SCOREP_USER_REGION_BEGIN(scorep_bt_loop, "<<bt_iter>>", SCOREP_USER_REGION_TYPE_DYNAMIC

Result in the CUBE profile:

- Iterations shown as separate call trees
- >Useful for checking results for specific iterations

or

- Select your user instrumented region and mark it as loop
- Choose hide iterations
- >View the Barplot statistics or the (thread x iterations) Heatmap

VICTOR COMPUTING

Loop Unrolling - Barplot



VIRTUAL INSTITUTE - HIGH PRODUCTIVITY SUPERCOMPUTING

Loop Unrolling – Heatmap



Further information

CUBE

- Parallel program analysis report exploration tools
 - Libraries for XML report reading & writing
 - Algebra utilities for report processing
 - GUI for interactive analysis exploration
- Available under New BSD open-source license
- Documentation & sources:
 - http://www.scalasca.org
- User guide also part of installation:
 - `cube-config --cube-dir`/share/doc/CubeGuide.pdf
- Contact:
 - mailto: scalasca@fz-juelich.de

