# *Scalasca* support for Intel Xeon Phi

**Brian Wylie** & Wolfgang Frings
Jülich Supercomputing Centre
Forschungszentrum Jülich, Germany

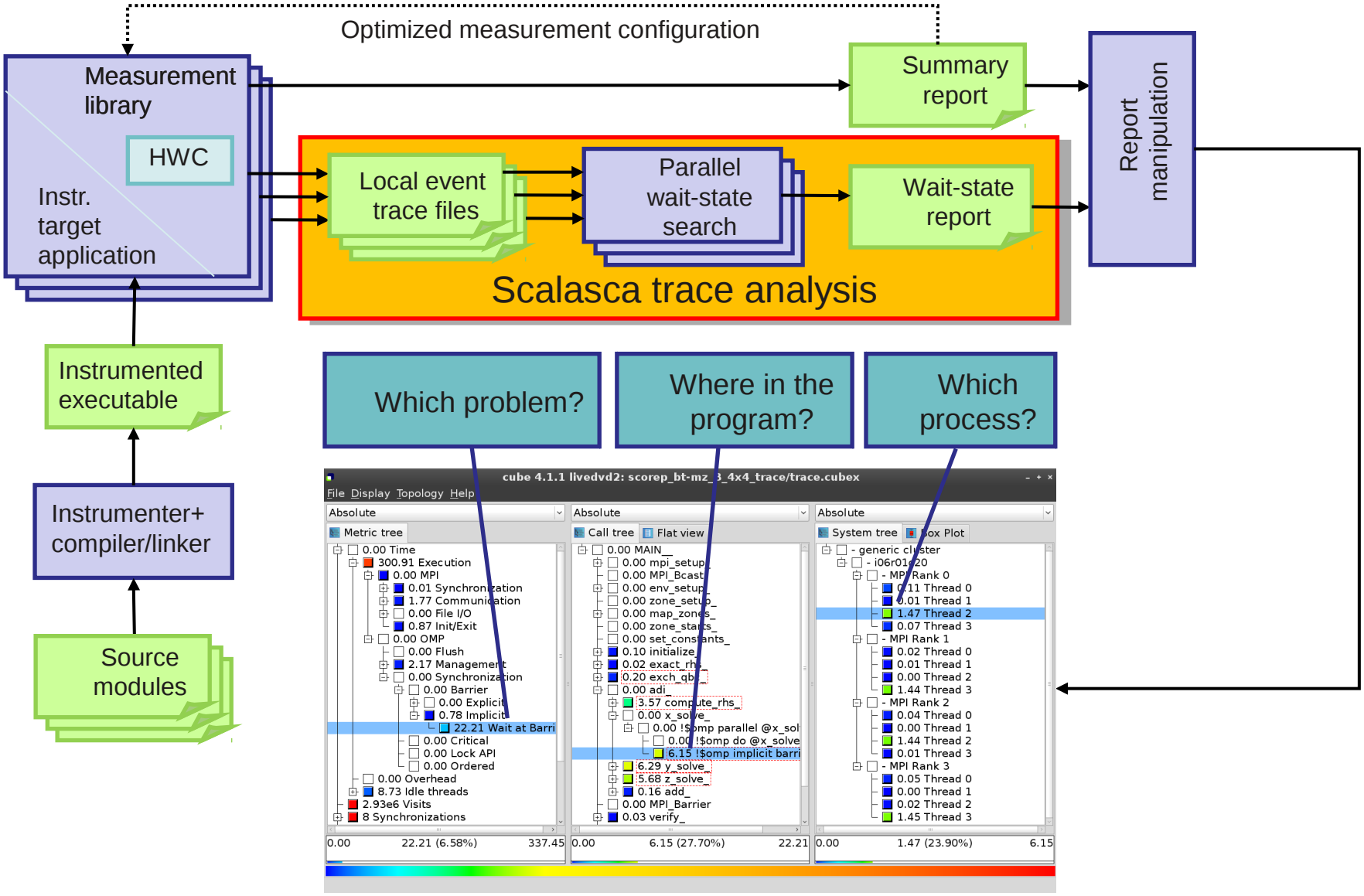
# Overview

- *Scalasca* performance analysis toolset
  - support for MPI & OpenMP parallel applications
  - runtime summarization & automatic trace analysis
- Intel Xeon Phi (MIC architecture)
  - parallel programming/execution models
- Case study
  - TACC Stampede
  - symmetric BT-MZ execution on 4 compute nodes
- Conclusion

# Scalasca

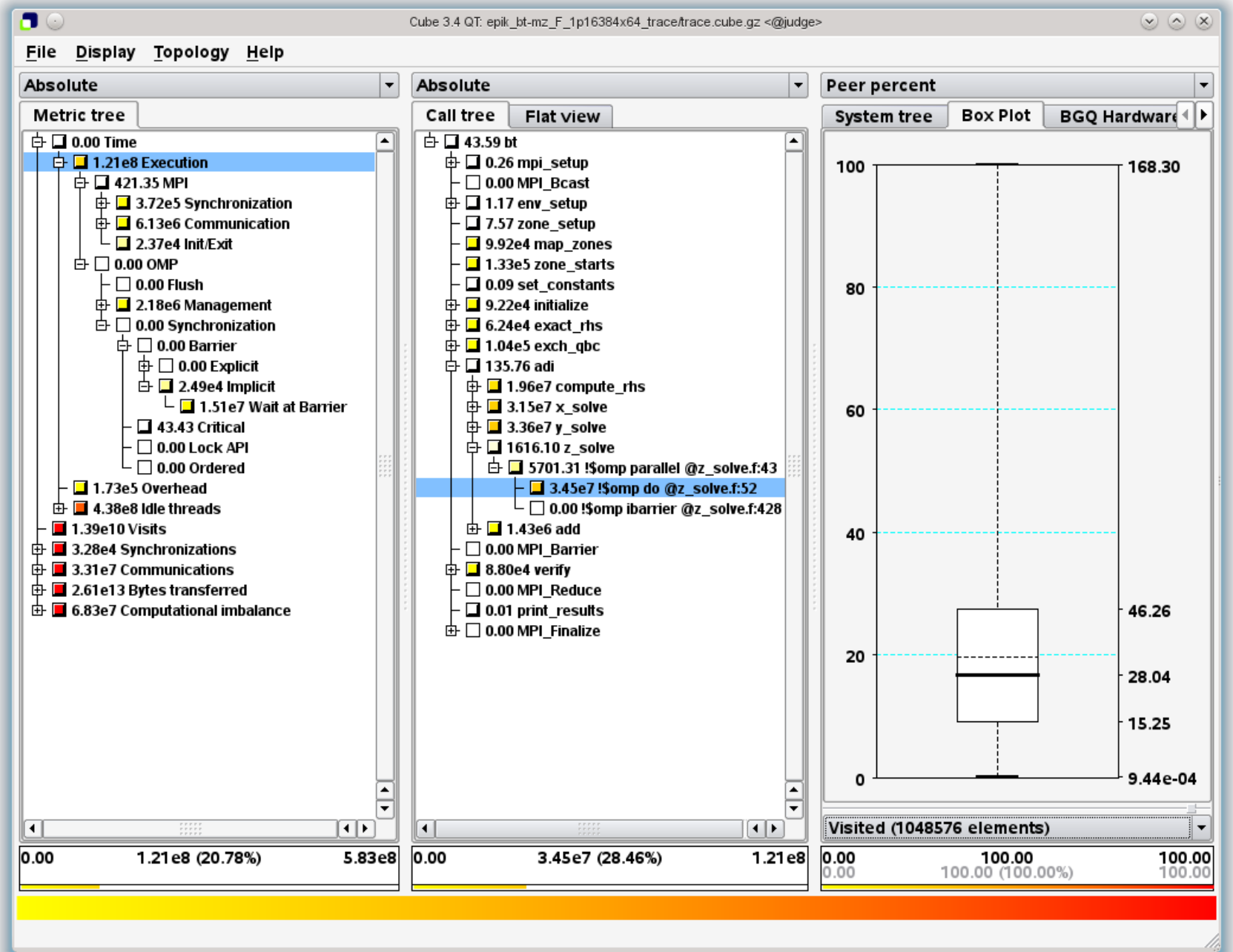
- Scalable performance analysis toolset for most popular parallel programming paradigms
  - supports MPI, OpenMP and hybrid OpenMP+MPI
- Specifically targeting large-scale parallel applications
  - such as those running on Blue Gene or Cray systems with thousands of processes or millions of threads
- Integrated instrumentation, measurement & analyses
  - runtime summarization (callpath profiling) and/or automatic event trace analysis
- Developed by JSC and GRS, available as open-source

# Scalasca workflow



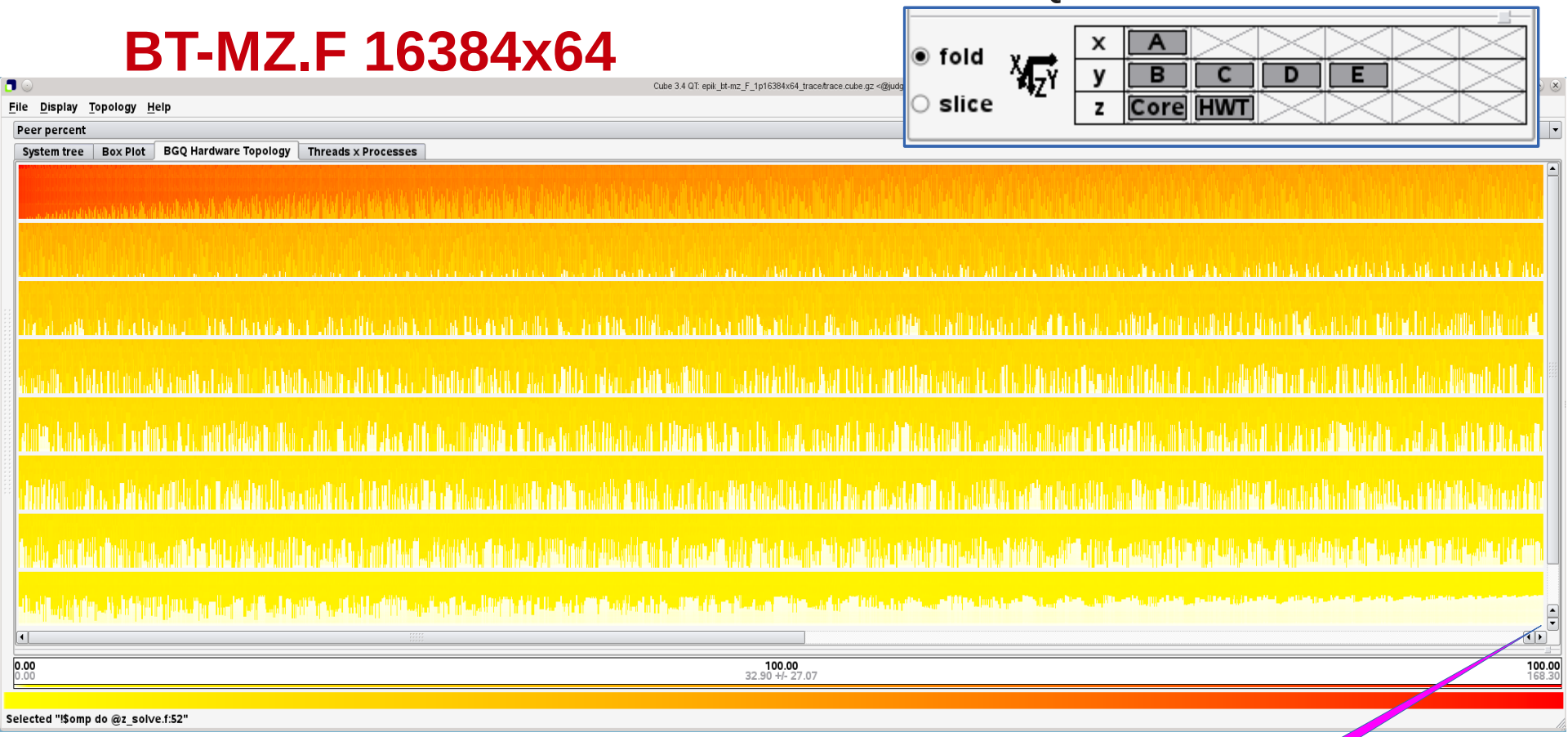
# BGQ BT-MZ.F execution imbalance

scalasca



# BT-MZ.F 16384x64

IBM BGQ 8x16x8x8x2 torus network



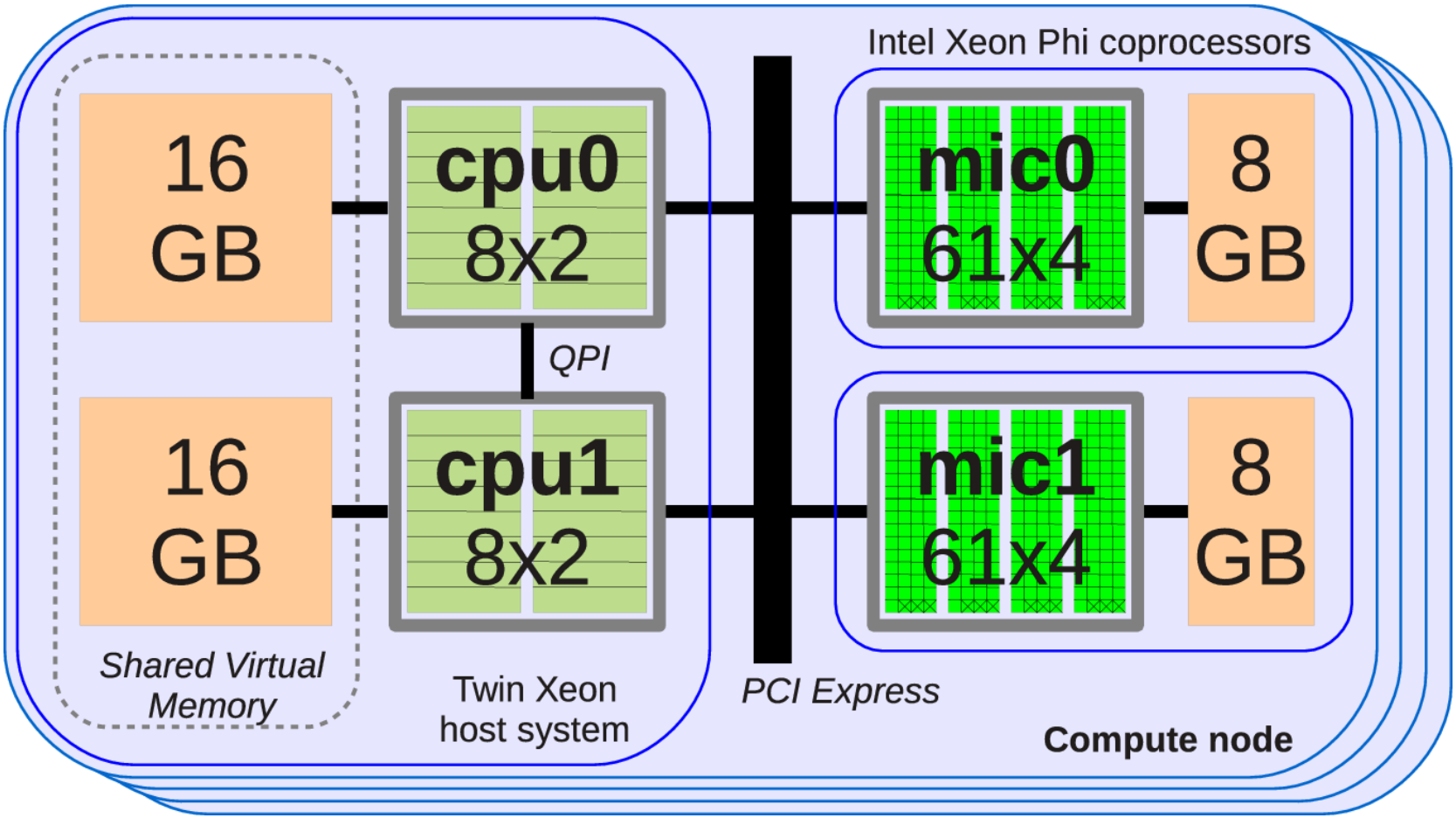

- 16384 MPI processes with 64 OpenMP threads
- 512 s benchmark execution (<3% dilation)
- 2623 s extra for 312GB trace collection+analysis

# SIONlib

Portable native parallel file I/O library & utilities

- Scalable massively parallel I/O to task-local files
- Manages single or multiple physical files on disk
  - reduces meta-data server contention, optimizes bandwidth available by matching blocksizes/alignment
- POSIX I/O compatible sequential & parallel API
- Tuned for common parallel filesystems (GPFS, Lustre)
- Convenient for application I/O, checkpointing
  - can be used by Scalasca tracing
- Developed by JSC, available as open-source
  - http://www.fz-juelich.de/jsc/sionlib/

# Intel Xeon Phi compute nodes



# Intel Xeon Phi (MIC)

| Host | Device |
|---|---|
| Xeon E5-2680 (processor) | Xeon Phi SE10P (coprocessor) |
| 2x sockets<br>8x 2.7 GHz cores, each 2-way HT<br>x86_64 (cpu)<br>32 GB shared virtual memory | 2x PCI Express cards<br>**61x** 1.1 GHz cores, each **4-way HT**<br>mic (native)<br>8 GB local memory per card |
| Intel Composer XE 2013 (13.1)<br>• normal compile & link<br>Intel MPI 4.1 | N/A<br>• **-mmic** cross-compile & link<br>Intel MPI 4.1 |

*MIC* = Many Integrated Core architecture (aka *manycore*)
*Xeon Phi* = Intel's first product (aka *Knights Corner*)

# MIC parallel programming models

| Host | | Device | |
|---|---|---|---|
| **MPI** | host offload to device | **MPI** | device offload to host |
| + **OpenMP**<br>+ OpenCL<br>+ pthreads | | + **OpenMP**<br>+ OpenCL<br>+ pthreads | |
| + F2008<br>+ TBB [C/C++]<br>+ Cilk Plus [C++] | | + F2008<br>+ TBB [C/C++]<br>+ Cilk Plus [C++] | |

***symmetric*** = host MPI[+MT] + device MPI[+MT]
*offload* = automatic or compiler-assisted kernel execution on device or host
*LEO* = language extension for offload (pragmas/directives similar to OpenMP)

# Scalasca usage (basic)

| Host | Device |
|---|---|
| *skin* mpiifort -O -openmp *.f | *skin* mpiifort -O -openmp **-mmic** *.f |
| *scan* mpiexec -n 2 a.out.cpu<br><br>*scan* mpiexec.hydra \<br>-host node0 -n 1 a.out.cpu \<br>: -host node1 -n 1 a.out.cpu | ssh mic0 \<br>-c "*scan* mpiexec -n 61 a.out.mic"<br>*scan* mpiexec.hydra \<br>-host mic0 -n 30 a.out.mic \<br>: -host mic1 -n 31 a.out.mic |
| *square* epik_a_2x16_sum | *square* epik_a_mic61x4_sum |

Symmetric execution:
*scan* mpiexec.hydra -host node0 -n 2 a.out.cpu : -host mic0 -n 61 a.out.mic
*square* epik_a_2x16+mic61x4_sum

# Scalasca case study configuration

TACC *Stampede* Dell PowerEdge C8220

- 6,400 compute nodes (2x Xeon + [012]x Xeon Phi)
- 4 compute node partition with 1x MIC for experiment

NPB3.3-MZ-MPI class D: bt-mz_D.68[.cpu|.mic]

SLURM_TASKS_PER_NODE=2(x4)
MIC_PPN=15
MIC_OMP_NUM_THREADS=OMP_NUMTHREADS=16
***scan -t*** ibrun.symm -c bt-mz_D.68.cpu -m bt-mz_D.68.mic

➝ epik_bt-mz_D_2p8x16+mic15p60x16_trace

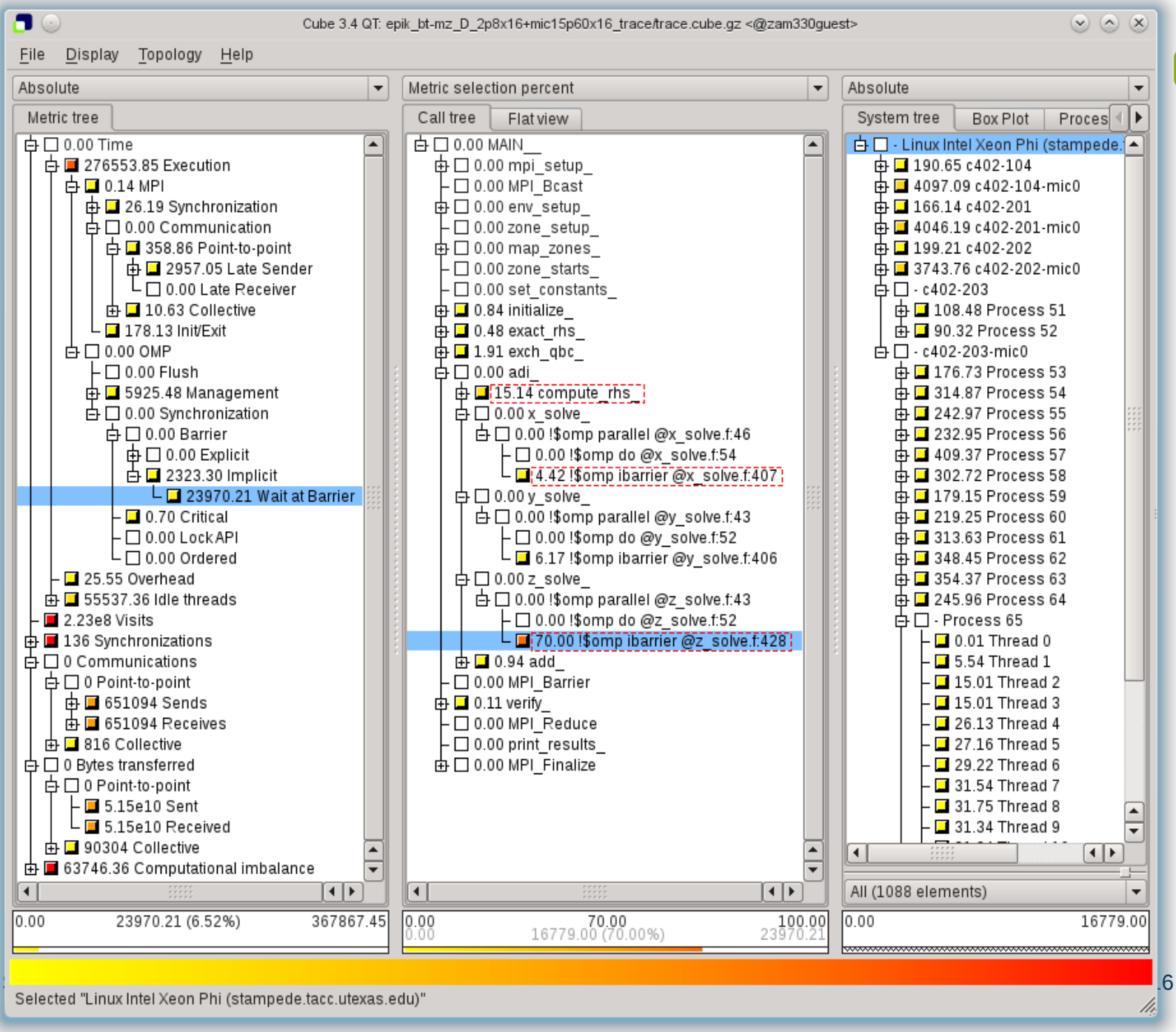
# Scalasca usage (stampede ibrun.symm)

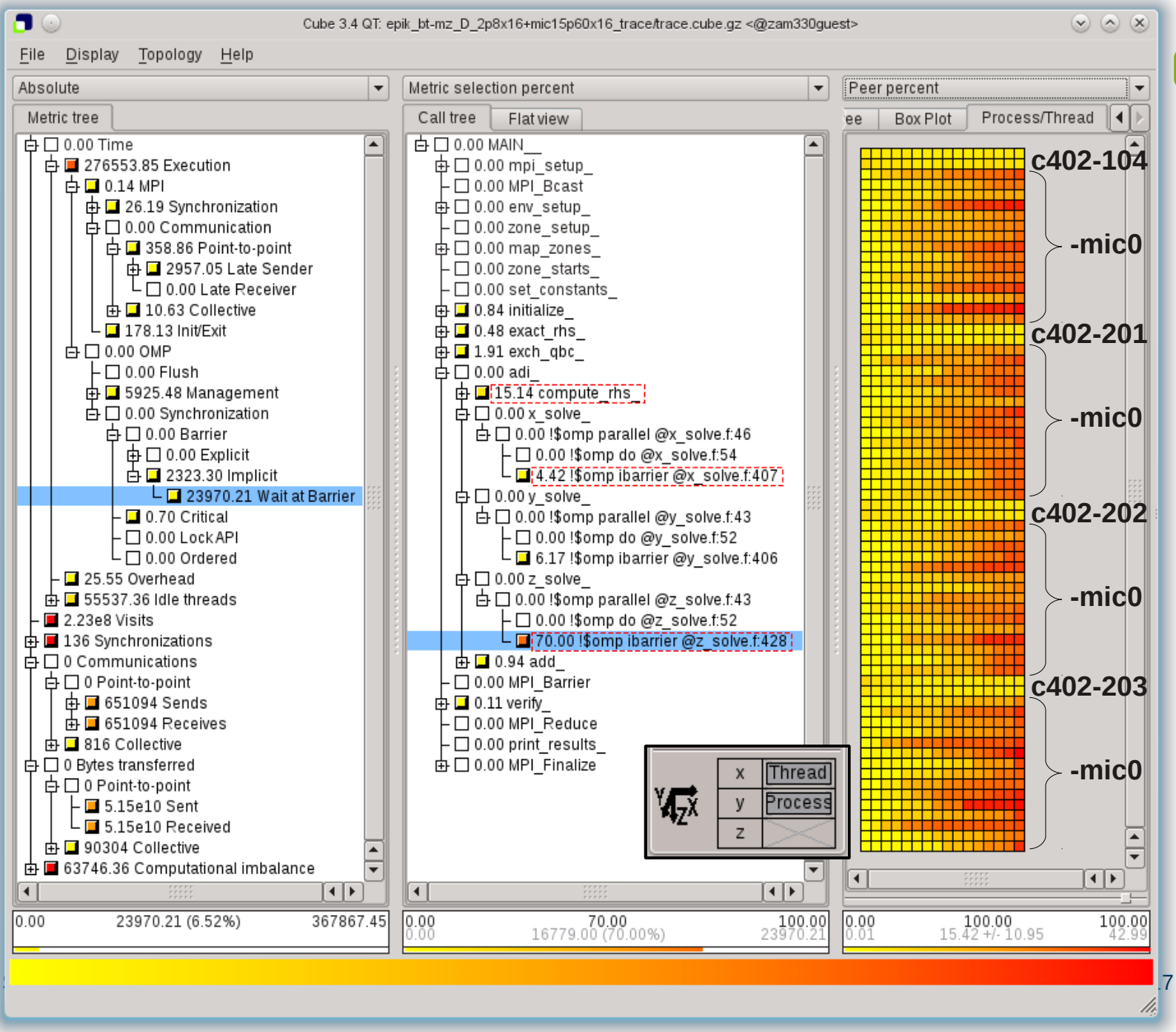
```
ELG_SION_FILES=-1  SCAN_ANALYZE_OPTS="-s -i"
scan  -t   ibrun.symm -c bt-mz_D.68.cpu -m bt-mz_D.68.mic
    S=C=A=N: Scalasca 1.4.3 trace collection and analysis
    S=C=A=N: ./epik_bt-mz_D_2p8x16+mic15p60x16_trace experiment archive
    ibrun.symm -c "bt-mz_D.68.cpu" -m "bt-mz_D.68.mic"
    [00000.0]EPIK: Created archive ./epik_bt-mz_D_2p8x16+mic15p60x16_trace
    [00000.0]EPIK: ELG_SION_FILES=63 determined automatically
    ….
    [00000.0]EPIK: Closed experiment ./epik_bt-mz_D_2p8x16+mic15p60x16_trace
    S=C=A=N: Collect done
    S=C=A=N: Analyze start
    ibrun.symm -c "$SCALASCA_DIR/bin/fe/scout.hyb -s -i" \
                    -m "$SCALASCA_DIR/bin/be/scout.hyb -s -i"
    Analyzing experiment archive ./epik_bt-mz_D_2p8x16+mic15p60x16_trace
    …
    S=C=A=N: ./epik_bt-mz_D_2p8x16+mic15p60x16_trace experiment complete
```

# Scalasca experiment characteristics

| Activity | Time |
|---|---|
| Reference (uninstrumented) execution | 325.41 s |
| Summary measurement execution<br>Summary report generation | 330.47 s [+1.6%]<br>0.49 s |
| Tracing measurement execution<br>Event trace generation<br>Event trace analysis (complete) | 330.98 s [+1.7%]<br>44.55 s<br>113.16 s |

Symmetric execution of BT-MZ class D configured with 68 MPI processes and 16 OpenMP threads per process on four *Stampede* Xeon Phi compute nodes. Custom instrumentation filter for Intel compiler (from preliminary score report). 5.08 GB event trace data (buffered). One SION event trace file per process.


Cube 3.4 QT: epik_bt-mz_D_2p8x16+mic15p60x16_trace/trace.cube.gz <@zam330guest>
File Display Topology Help
Absolute
Metric tree
0.00 Time
276553.85 Execution
0.14 MPI
26.19 Synchronization
0.00 Communication
358.86 Point-to-point
2957.05 Late Sender
0.00 Late Receiver
10.63 Collective
178.13 Init/Exit
0.00 OMP
0.00 Flush
5925.48 Management
0.00 Synchronization
0.00 Barrier
0.00 Explicit
2323.30 Implicit
23970.21 Wait at Barrier
0.70 Critical
0.00 Lock API
0.00 Ordered
25.55 Overhead
55537.36 Idle threads
2.23e8 Visits
136 Synchronizations
0 Communications
0 Point-to-point
651094 Sends
651094 Receives
816 Collective
0 Bytes transferred
0 Point-to-point
5.15e10 Sent
5.15e10 Received
90304 Collective
63746.36 Computational imbalance
0.00 23970.21 (6.52%) 367867.45
Metric selection percent
Call tree
Flat view
0.00 MAIN__
0.00 mpi_setup_
0.00 MPI_Bcast
0.00 env_setup_
0.00 zone_setup_
0.00 map_zones_
0.00 zone_starts_
0.00 set_constants_
0.84 initialize_
0.48 exact_rhs_
1.91 exch_qbc_
0.00 adi_
15.14 compute_rhs_
0.00 x_solve_
0.00 !$omp parallel @x_solve.f:46
0.00 !$omp do @x_solve.f:54
4.42 !$omp ibarrier @x_solve.f:407
0.00 y_solve_
0.00 !$omp parallel @y_solve.f:43
0.00 !$omp do @y_solve.f:52
6.17 !$omp ibarrier @y_solve.f:406
0.00 z_solve_
0.00 !$omp parallel @z_solve.f:43
0.00 !$omp do @z_solve.f:52
70.00 !$omp ibarrier @z_solve.f:428
0.94 add_
0.00 MPI_Barrier
0.11 verify_
0.00 MPI_Reduce
0.00 print_results_
0.00 MPI_Finalize
0.00 70.00 100.00
0.00 16779.00 (70.00%) 23970.21
Absolute
System tree
Box Plot
Proces
- Linux Intel Xeon Phi (stampede.
190.65 c402-104
4097.09 c402-104-mic0
166.14 c402-201
4046.19 c402-201-mic0
199.21 c402-202
3743.76 c402-202-mic0
- c402-203
108.48 Process 51
90.32 Process 52
- c402-203-mic0
176.73 Process 53
314.87 Process 54
242.97 Process 55
232.95 Process 56
409.37 Process 57
302.72 Process 58
179.15 Process 59
219.25 Process 60
313.63 Process 61
348.45 Process 62
354.37 Process 63
245.96 Process 64
- Process 65
0.01 Thread 0
5.54 Thread 1
15.01 Thread 2
15.01 Thread 3
26.13 Thread 4
27.16 Thread 5
29.22 Thread 6
31.54 Thread 7
31.75 Thread 8
31.34 Thread 9
All (1088 elements)
0.00 16779.00
Selected "Linux Intel Xeon Phi (stampede.tacc.utexas.edu)"


Cube 3.4 QT: epik_bt-mz_D_2p8x16+mic15p60x16_trace/trace.cube.gz <@zam330guest>
File Display Topology Help
Absolute
Metric selection percent
Peer percent
Metric tree
Call tree
Flat view
ee
Box Plot
Process/Thread
0.00 Time
276553.85 Execution
0.14 MPI
26.19 Synchronization
0.00 Communication
358.86 Point-to-point
2957.05 Late Sender
0.00 Late Receiver
10.63 Collective
178.13 Init/Exit
0.00 OMP
0.00 Flush
5925.48 Management
0.00 Synchronization
0.00 Barrier
0.00 Explicit
2323.30 Implicit
23970.21 Wait at Barrier
0.70 Critical
0.00 Lock API
0.00 Ordered
25.55 Overhead
55537.36 Idle threads
2.23e8 Visits
136 Synchronizations
0 Communications
0 Point-to-point
651094 Sends
651094 Receives
816 Collective
0 Bytes transferred
0 Point-to-point
5.15e10 Sent
5.15e10 Received
90304 Collective
63746.36 Computational imbalance
0.00 MAIN__
0.00 mpi_setup_
0.00 MPI_Bcast
0.00 env_setup_
0.00 zone_setup_
0.00 map_zones_
0.00 zone_starts_
0.00 set_constants_
0.84 initialize_
0.48 exact_rhs_
1.91 exch_qbc_
0.00 adi_
15.14 compute_rhs_
0.00 x_solve_
0.00 !$omp parallel @x_solve.f:46
0.00 !$omp do @x_solve.f:54
4.42 !$omp ibarrier @x_solve.f:407
0.00 y_solve_
0.00 !$omp parallel @y_solve.f:43
0.00 !$omp do @y_solve.f:52
6.17 !$omp ibarrier @y_solve.f:406
0.00 z_solve_
0.00 !$omp parallel @z_solve.f:43
0.00 !$omp do @z_solve.f:52
70.00 !$omp ibarrier @z_solve.f:428
0.94 add_
0.00 MPI_Barrier
0.11 verify_
0.00 MPI_Reduce
0.00 print_results_
0.00 MPI_Finalize
x
Thread
y
Process
z
c402-104
-mic0
c402-201
-mic0
c402-202
-mic0
c402-203
-mic0
0.00 23970.21 (6.52%) 367867.45
0.00 70.00 100.00
0.00 16779.00 (70.00%) 23970.21
0.00 100.00 100.00
0.01 15.42 +/- 10.95 42.99

# Scalasca BT-MZ case study insight

- 75% of total allocation time is actual computation
- 15% of time considered to be *Idle threads*
  - unused compute resources outside of parallel regions
- 1% of total time in MPI
  - similarly done outside of parallel regions
- 9% of total time in OpenMP parallelization overheads
  - 6.5% *OpenMP implicit barrier synchronization time* at end of parallel regions
    - 70% of this within *Z_solve* routine (lines 43-428)
    - unevenly distributed across threads *on devices,* though threads on hosts apparently well balanced
    - maximum 43 seconds, mean 15.4±11.0 seconds

# Current and future work

- Extend support for symmetric/heterogeneous MPMD measurement collection and analysis
  - mpiexec.hydra
  - consistency checks
  - additional metrics?
- Migration to new Score-P measurement infrastructure
  - support for additional threading models
  - support for OpenMP target offload

# Conclusions

- Port of Scalasca toolset to MIC was straightforward
  - configuration currently somewhat complicated due to evolving/maturing environment
  - provides familiar usage model
- Measurement and analysis of symmetric execution exploits combined host+device installation
  - smart mode determination
- Facilitates performance analysis for tuning/optimization of MPI and/or OpenMP

# Acknowledgments


- XSEDE (Jay Alameda)
- TACC (Stampede)
- Intel/JSC ExaCluster Laboratory
- DEEP Project (EU FP7)
- Scalasca development team

scalasca

## Further information

# **Sc**alable performance **a**nalysis of **la**rge-**sc**ale parallel **a**pplications

- toolset for scalable performance measurement & analysis of MPI, OpenMP and MPI+OpenMP parallel applications
- supporting most popular HPC computer systems
- available under New BSD open-source license
- sources, documentation & publications:
  - http://www.scalasca.org
  - mailto: scalasca@fz-juelich.de