

ParaTools

Drs. John C. Linford, Sameer Shende, Allen Malony
{jlinford, sameer, malony}@paratools.com

29 June 2017
Northrop Grumman Lunch-N-Learn
McLean, VA

Contact Info

Phone: 541-913-8797

Fax: 541-343-6086

info@paratools.com

ParaTools, Inc.
2836 Kincaid St.
Eugene, OR 97405, USA

Northrop Grumman Cync Program

- A partnership between Northrop Grumman and the Cyber Incubator@bwtech
- Innovative, technology-driven startup companies addressing critical market needs in:

NORTHROP GRUMMAN

bwtech
@UMBC
RESEARCH & TECHNOLOGY PARK

- Cyber
- Data Sciences
- Big Data
- Secure Mobility
- Cyber Physical Systems
- Critical Infrastructure Protection

ParaTools

ParaTools



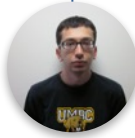
Dr. Allen Malony (CEO)



Dr. Sameer Shende (President)



Dr. John Linford



Dr. Sam Khuvis



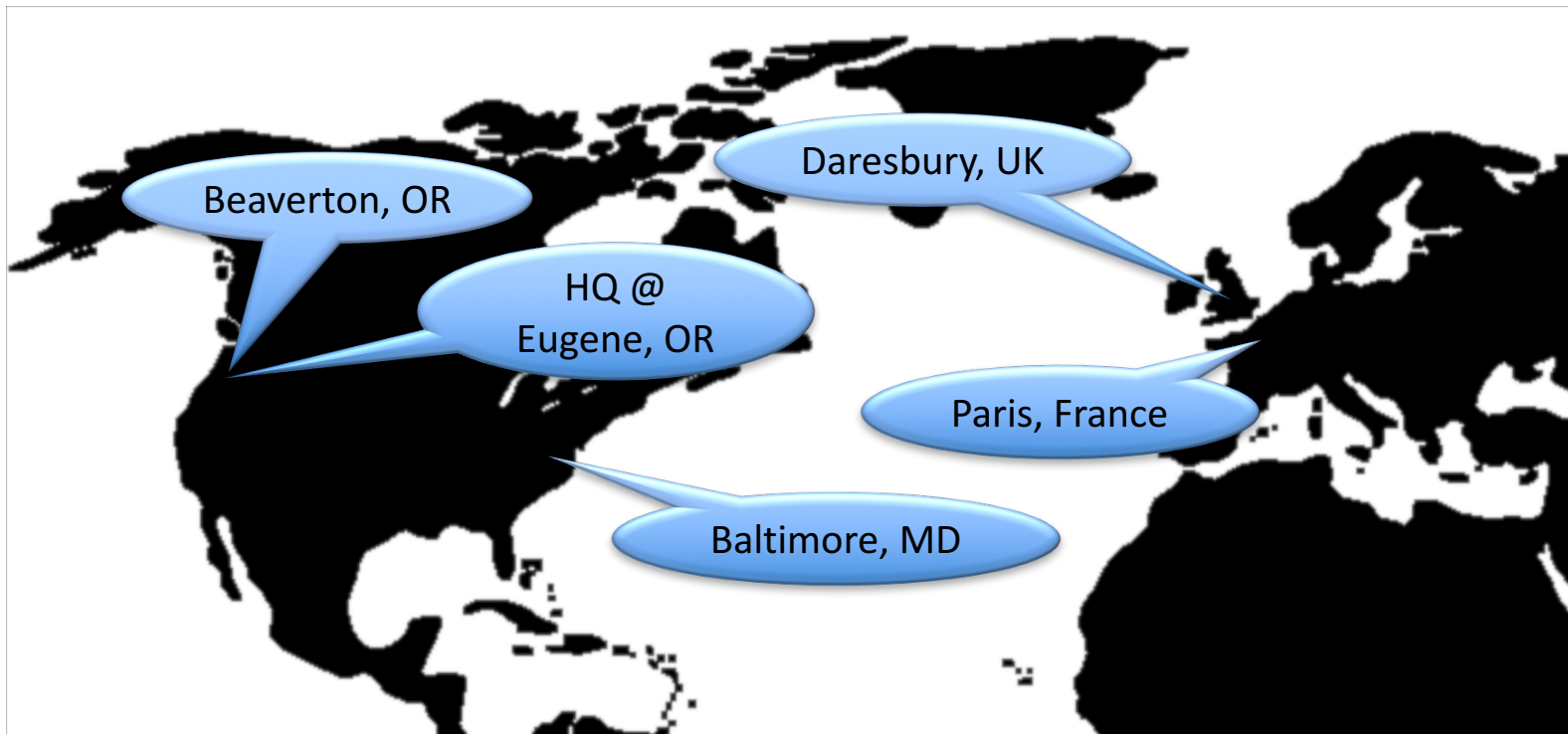
Dr. Nick Chaimov



Dr. David Mackay



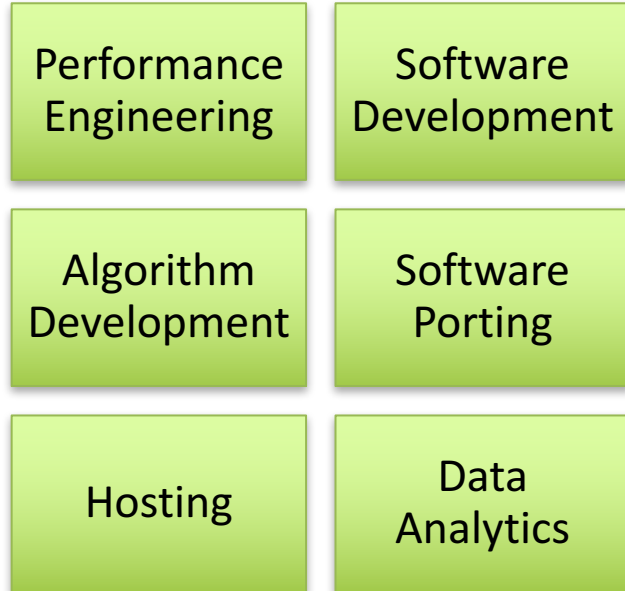
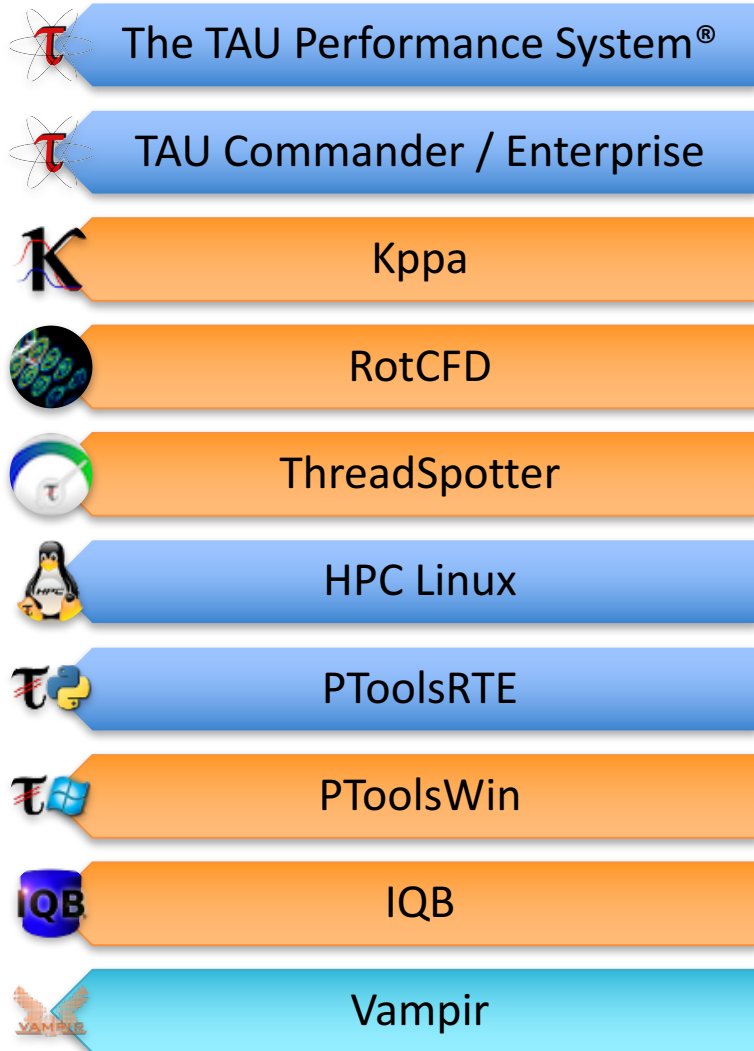
Dr. Jean-Baptiste Besnard



ParaTools Accelerates Software



Products and Services



ParaTools® and *IQB*® are registered trademarks of ParaTools, Inc.

ParaTools is the sole licensee of the *TAU Performance System*® trademark owned by the State of Oregon acting by and through the State Board of Higher Education on behalf of the University of Oregon.

In the Press



Rogue Wave Reveals New Releases and a Presentation Theater at SC14

BOULDER, CO -- (Marketwired) -- 11/14/14 -- [Rogue Wave Software](#), the largest independent provider of cross-platform software development tools and embedded components, will be highlighting their products at [Supercomputer 2014](#) from November 17-20 at Ernest N. Morial Convention Center in New Orleans. Rogue Wave will have new and updated product offerings, as well as several chances to educate, explore, and learn at this year's conference.

With recent releases of both [IMSL Fortran Numerical Library 7.1](#) and [Visualization for C++ 5.8](#), attendees can expect to learn about new performance improvements, code fixes, and updates to supported platforms. In addition to the C++ product release, Rogue Wave will also share the accompaniments made to the [Visualization product set](#). Adding [IViews](#) and [Elixir](#) expands the Visualization toolset for C++, Java, and Adobe Flex.

Also at this conference, Rogue Wave will join [ParaTools](#) in releasing the [source code for ThreadSpotter](#). A memory optimization tool that analyzes cache memory and thread communication in single and multicore systems, ThreadSpotter pinpoints performance issues and provides specific guidance on how to correct them, increasing program performance and improving developer productivity.



DEFENSE INDUSTRY DAILY

Defense program acquisition news, budget data, market briefings

Contact ▾ Subscribe ▾ Log in ▾ Search

DoD Supercomputers: Speeding Along the Digital Highway

Aug 10, 2010 11:44 UTC by Defense Industry Daily staff

Four teams get up to \$100 million supercomputers. (Aug 6/10)

The US Department of Defense Modernization Program (HPCMP) supercomputing capabilities... small high performance computing departments run... independent of the others. PCMP brings these capabilities together. The program network communications, and computational support new weapons systems, prepare US aircraft support long-term weather predictions to plan hu...



UMBC tech park lures Oregon software company

Feb 28, 2014, 1:33pm EST | **UPDATED:** Jun 9, 2014, 9:12am EDT



Sarah Gantz
Reporter - Baltimore Business Journal
Email | Twitter

A West Coast software company is expanding to Baltimore to tap into the area's growing cyber security industry.

ParaTools, based in Eugene, Ore., has opened a Baltimore office at the [University of Maryland](#), Baltimore County's [bwtech@UMBC](#) research and technology park. The company's software was designed to make complex

supercomputers run more efficiently. Now, ParaTools wants to use the same efficiency software to improve cyber operations.

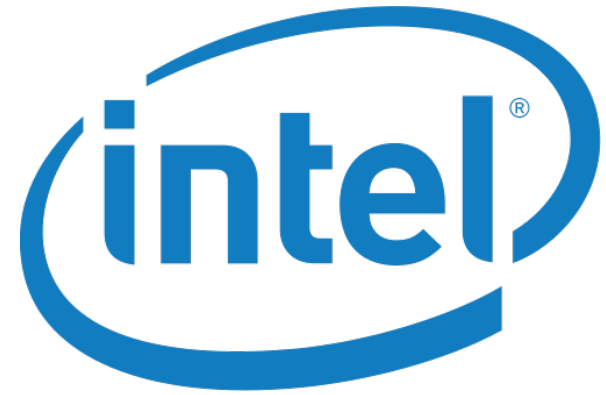
"With our skills in optimizing software and computer performance we think there's a lot of opportunity in the cyber," said [John Linford](#), who is heading up the company's Baltimore office.

ParaTools' software attaches itself to a system and creates a profile that shows how and where time is being spent. The information can be used to rewrite code in trouble spot and make the system run more smoothly.

A possible cyber use for the company's technology is improving efficiency of drone aircraft that scans for suspicious vehicles. Previously, the drones collected data and brought it home to be analyzed. With ParaTools' technology, could send data real-time, so they can more quickly address any problems identified.

- <http://tau.uoregon.edu/success/irmhd.pdf>
- <http://www.defenseindustrydaily.com/Up-to-1471M-to-HPTi-for-DoD-High-Performance-Computing-Work-05688/>
- <http://java.sys-con.com/node/3228954>
- <http://www.bizjournals.com/baltimore/blog/cyberbizblog/2014/02/umbc-tech-park-lures-oregon-software.html>

Partnerships



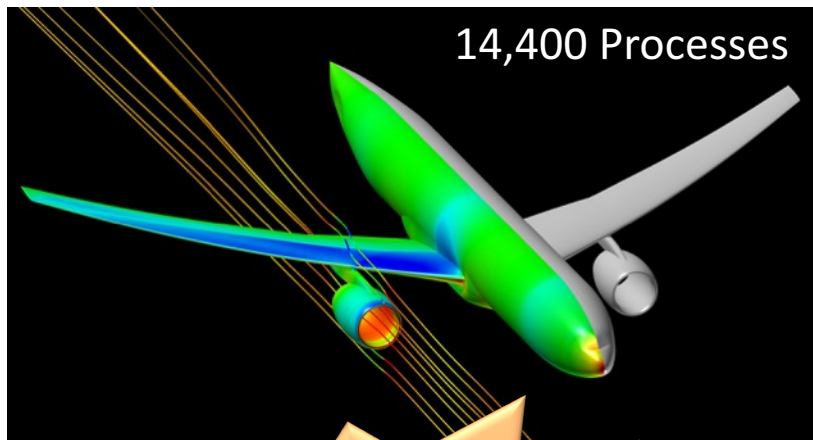
ParaTools LTD in Daresbury UK

- Located in the Sci-Tech Daresbury Innovation Centre
- In partnership with STFC Daresbury Laboratory
 - Luke Mason co-PI

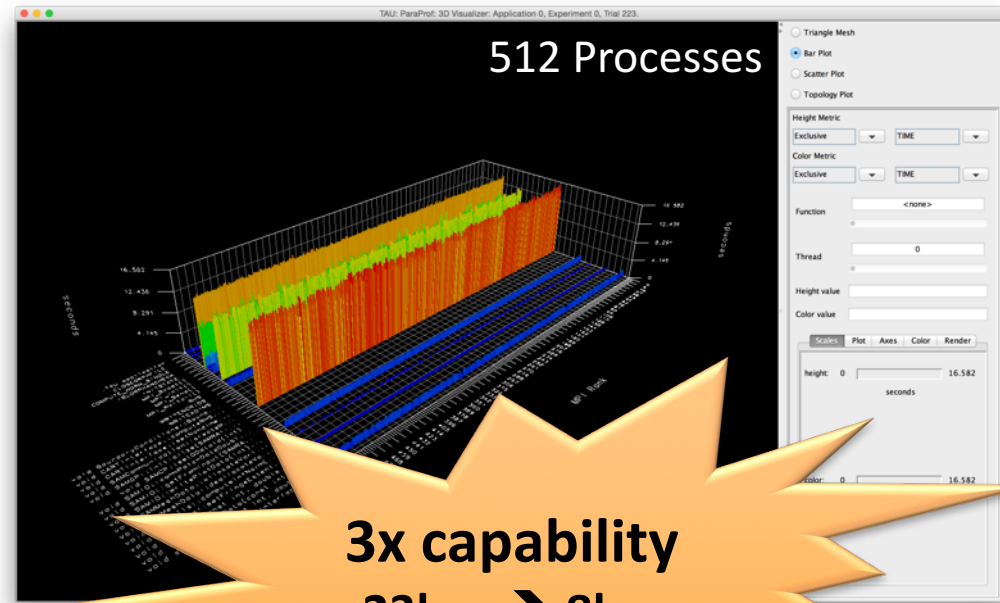


NASA, DOD, DOE, Industry

“These days I get excited about 1-2% speedups that I find....quite unusual to find something of this magnitude these days, especially with just a 2-line fix in the code! :)”



33% Faster



High Performance Cybersecurity

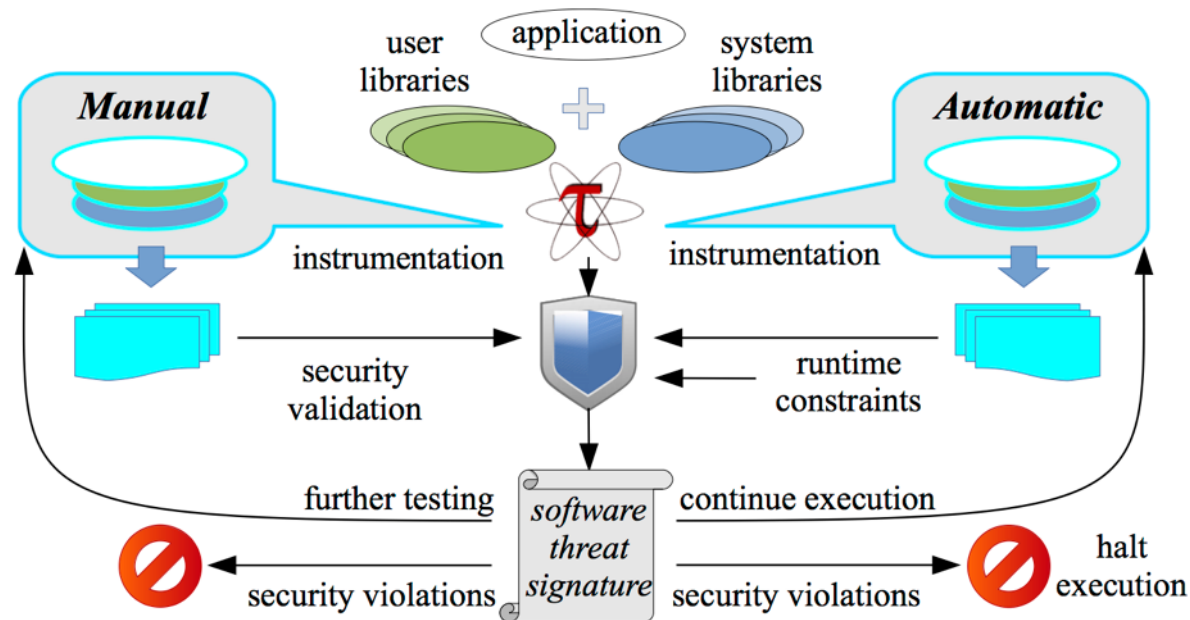
- Risk from inexperienced users, COTS, BYOD, and community/user-installable software packages.¹
- Performance impact of existing solutions is too high.



1. M. Garnaeva et al. *Kaspersky Security Bulletin 2015*. Kaspersky Labs, Dec. 2015.

Secure Intelligent Runtime Environment (SIREN)

- Profile-driven threat quantification and isolation:
 - Safely test and analyze untrusted software.
 - Dynamically react to suspicious shifts in behavior.
 - **Extremely low overhead, no virtualization.**



Low Overhead Application Isolation

- Monitor the application's ***threat signature*** to isolate software that attempts to violate system integrity.
- Complements existing security measures:
 - Firewalls.
 - VM-based isolation frameworks.
 - Container-based isolation.

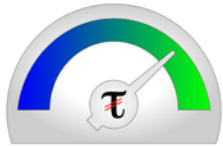


No VM required!

User-space Threat Quantification

Package Unicorn Threat Report

ParaTools



Threat Signature: 7.8

- Author1, Author2
- Project Webpage
- Software License
- Source Code File Count
- Storage Footprint
- Memory Footprint

Source Code Analysis

```
q = catalog.catalog()
d = q.get_writable_dir()
file = os.path.join(d, 'some_silly_file')
try:
    f = open(file, 'w')
    f.write('bob')
finally:
    f.close()
```

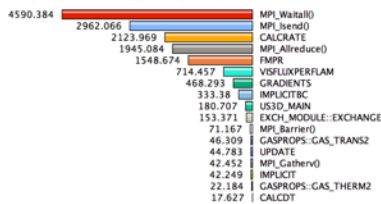
Network Access

Bytes Read	conn=128.223.202.191.port=56309	376	376	1	1	1	0
Bytes Read	<file="/home/users/sameer/python25_compiled/_db.linux225compiled_catalog">	1,251,077	890	4,400	8	1,507,322	1,615,572
Bytes Read	<file="/home/users/sameer/python25_compiled/dummy_catalog">	1,251,185	892	4,400	8	1,503,804	1,615,220
Bytes Read	<file="/home/users/sameer/python25_compiled/linux225compiled_catalog">	1,250,317	817	4,400	37	1,530,425	1,617,869
Bytes Read	<file="/home/users/sameer/python25_compiled/m2/_db.linux225compiled_catalog">	120	3	40	40	40	0
Bytes Read	<file="/home/users/sameer/python25_compiled/m3/_db.linux225compiled_catalog">	120	3	40	40	40	0
Bytes Read	<file="/home/users/sameer/python25_compiled/sc_9f9d51bc70ef21ca5c14f307980a29d80.cpp">	1,252,497	422	4,440	40	2,968,002	3,015,281
Bytes Read	<file="/home/users/sameer/python25_compiled/some_silly_file">	1,250,280	409	4,440	40	3,056,919	896,585
Bytes Read	<file="/home/users/sameer/python25_compiled/tmp/fYUj">	231	3	77	77	77	0

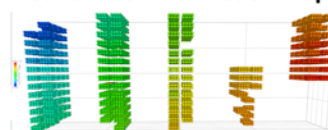
Files Access

```
Bytes Written <file="/home/users/sameer/python25_compiled/_db.linux225compiled_catalog">
Bytes Written <file="/home/users/sameer/python25_compiled/dummy_catalog">
Bytes Written <file="/home/users/sameer/python25_compiled/linux225compiled_catalog">
Bytes Written <file="/home/users/sameer/python25_compiled/m2/_db.linux225compiled_catalog">
Bytes Written <file="/home/users/sameer/python25_compiled/m3/_db.linux225compiled_catalog">
Bytes Written <file="/home/users/sameer/python25_compiled/sc_9f9d51bc70ef21ca5c14f307980a29d80.cpp">
Bytes Written <file="/home/users/sameer/python25_compiled/some_silly_file">
Bytes Written <file="/tmp/fYUj">
```

Performance Profile



Network Communication Map



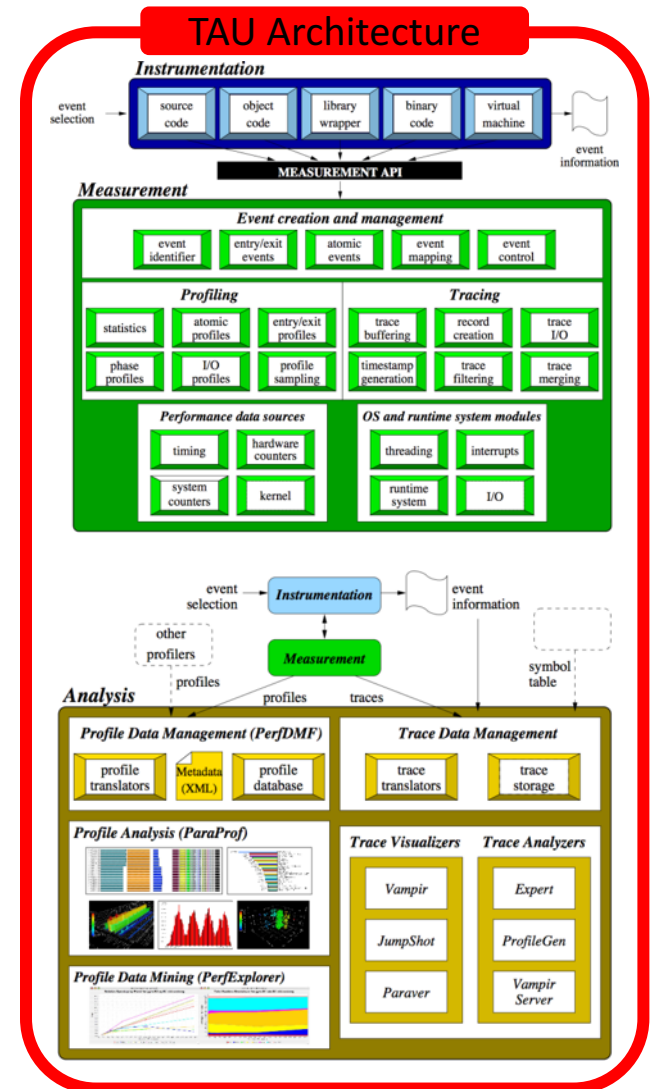
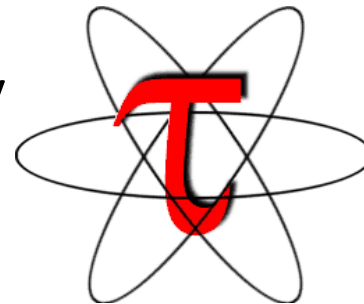
- Quantify the potential risk of using a package.
- Automatically filter by software behavior.
- Make system-level assertions about package characteristics:
 - “Only FIPS-compliant Python packages.”

NGC Lunch and Learn

THE TAU PERFORMANCE SYSTEM

The TAU Performance System®

- *Integrated toolkit* for performance problem solving
 - Instrumentation, measurement, analysis, visualization
 - Portable profiling and tracing
 - Performance data management and data mining
- Developed for over **20 years**
- Direct and indirect measurement
- *Free, open source, BSD license*
- Available on all HPC platforms (and some non-HPC)
- <http://tau.uoregon.edu/>



Use Cases

- Cybersecurity
 - Profile-driven threat quantification and isolation
- Code Modernization
 - FraPPE: the Framework for Parallel Program Engineering (US ARMY)
- Software Optimization
 - Load balancing
 - “Hot spot” identification
 - Improvement quantification
- Performance Regression Testing

TAU Supports All Platforms

C/C++

Fortran

pthread

Intel GNU

MinGW

Insert
yours
here

CUDA

UPC

OpenACC

Intel MIC

LLVM

Linux

BlueGene

Android

GPI

Java

OpenMP

Cray

Windows

Fujitsu

MPC

Python

MPI

Sun

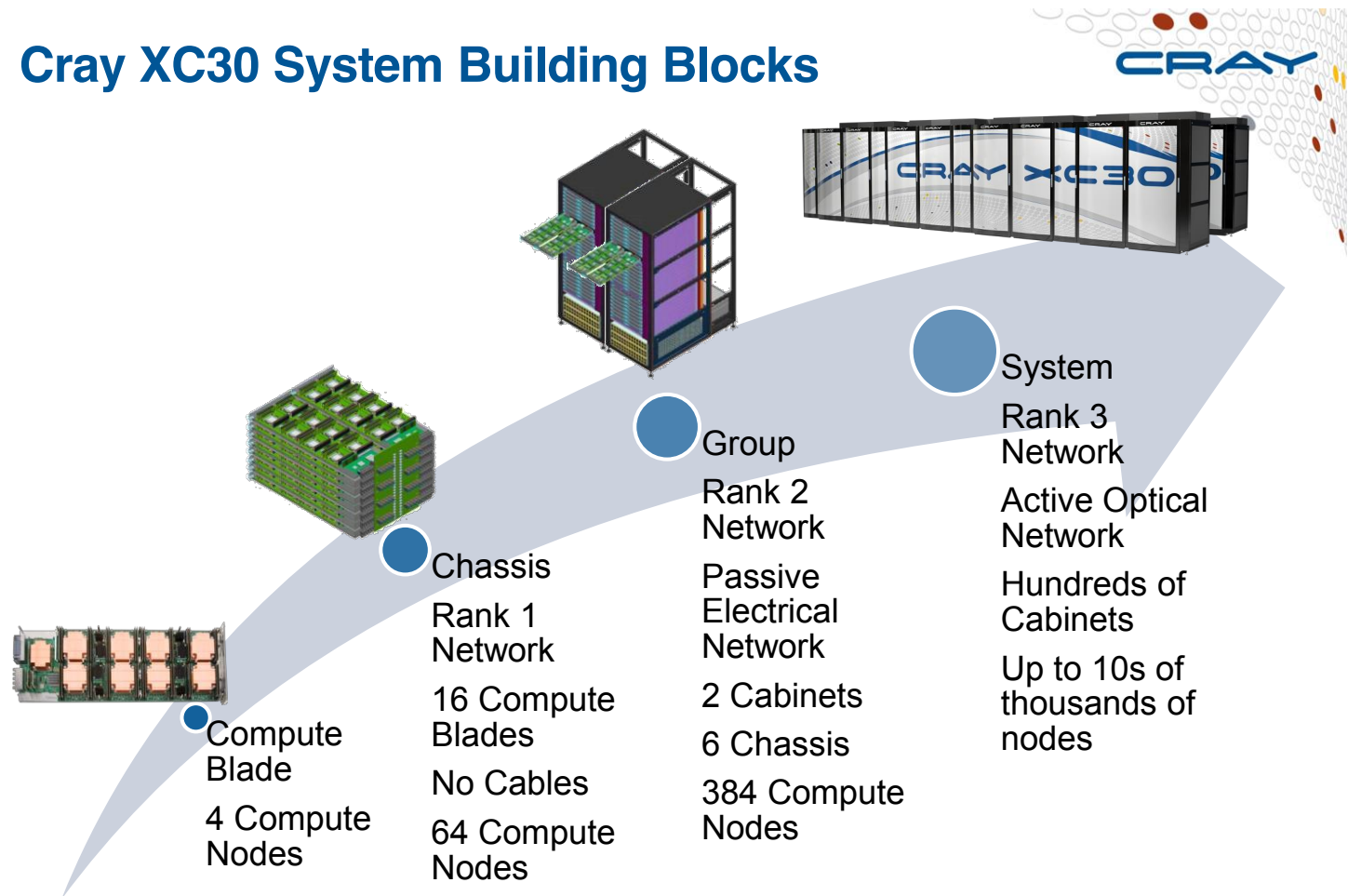
AIX

ARM

OS X

Instrumentation at Every System Layer

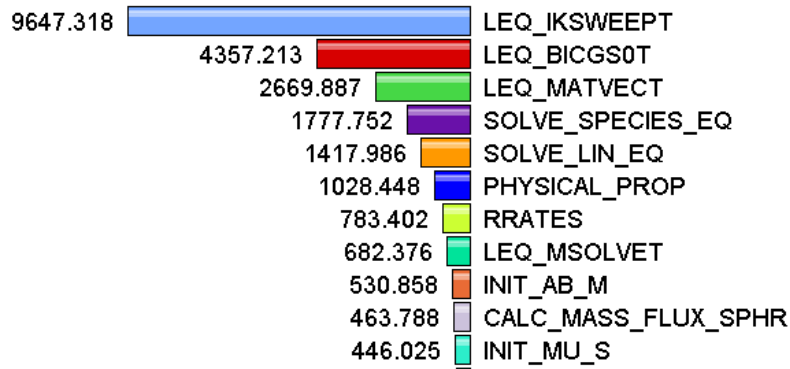
Cray XC30 System Building Blocks



Copyright © Cray Inc.

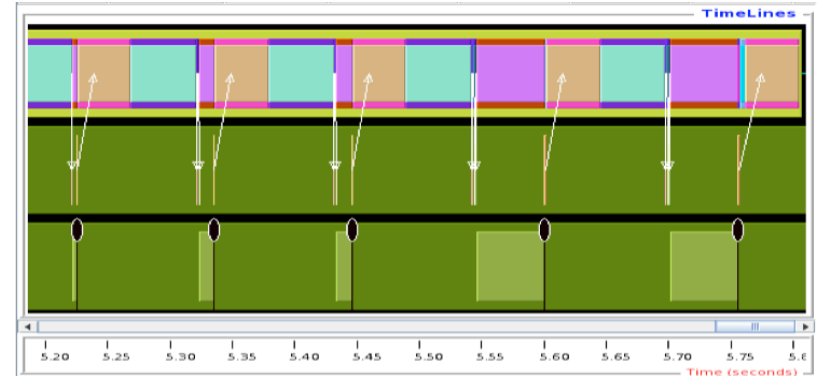
Types of Event Data

Profiling



Shows
how much time
was spent in each
routine

Tracing



Shows
when events
take place on a
timeline

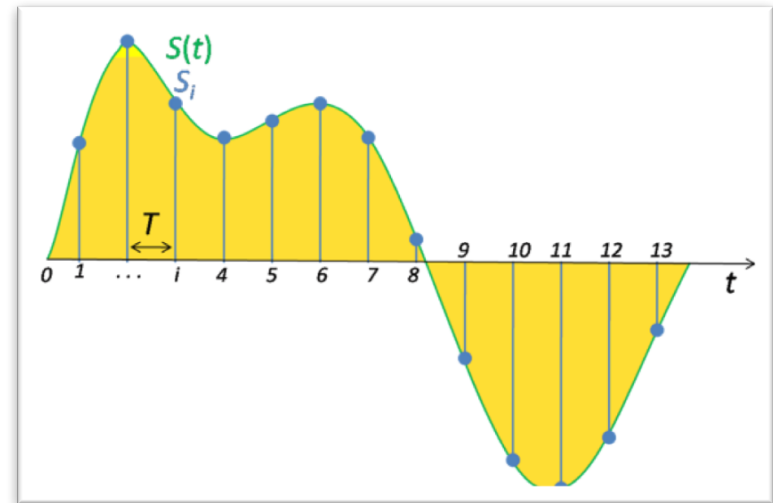
Event Measurement

Direct via Probes

```
call TAU_START('potential')  
// code  
call TAU_STOP('potential')
```

- Exact measurement
- Fine-grain control
- Calls inserted into code

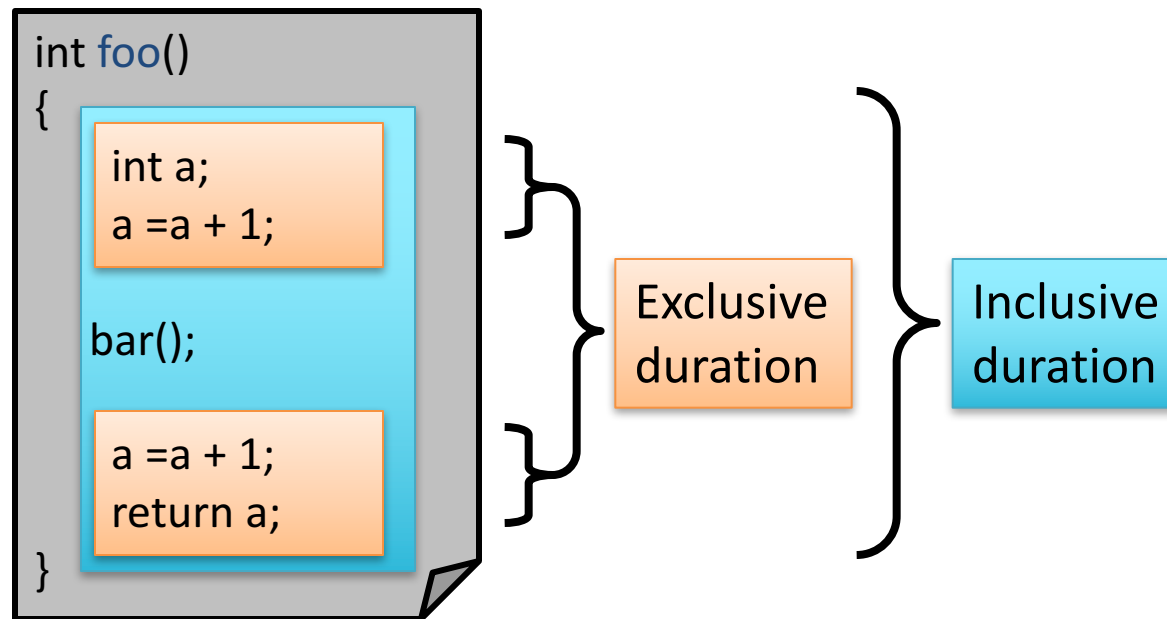
Indirect via Sampling



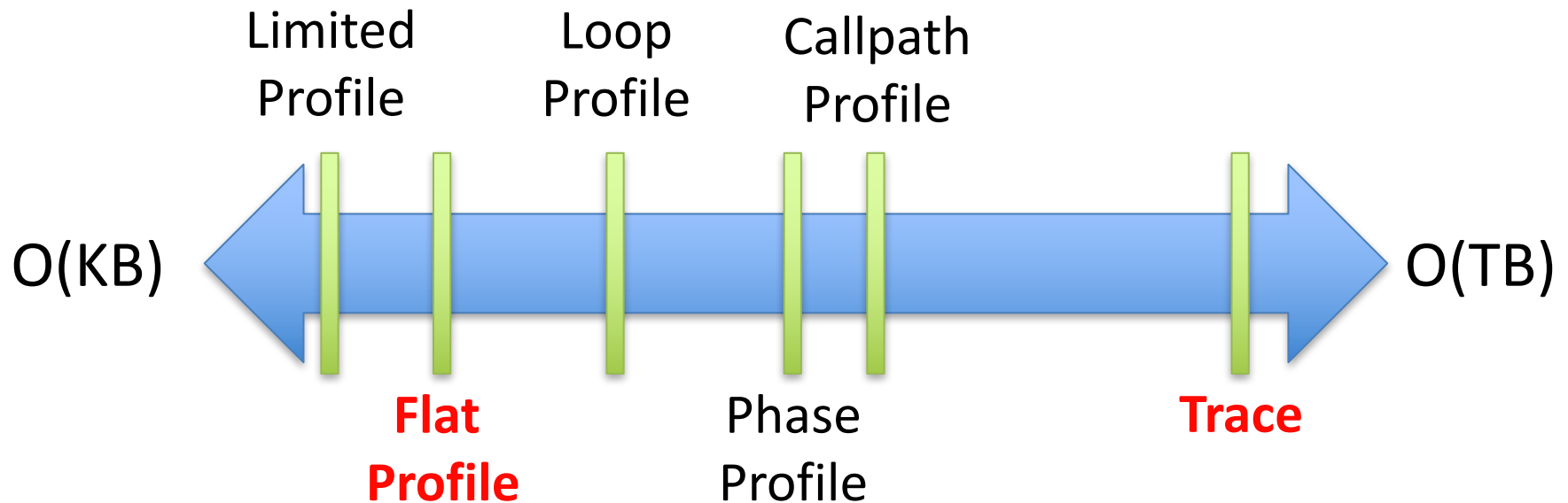
- No code modification
- Minimal effort
- Relies on debug symbols (**-g** option)

Inclusive vs. Exclusive Measurements

- **Exclusive** measurements for **region only**
- **Inclusive** measurements **includes child** regions



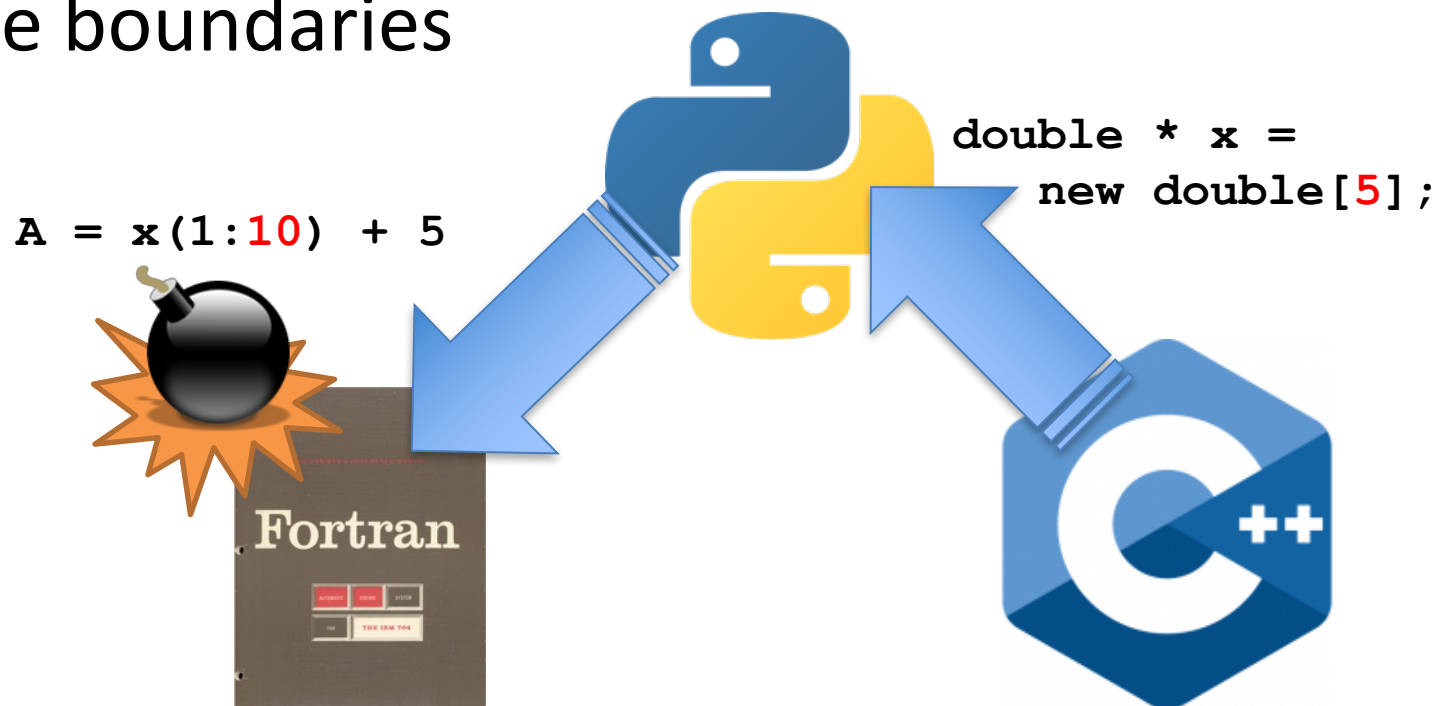
How Much Data do you Want?



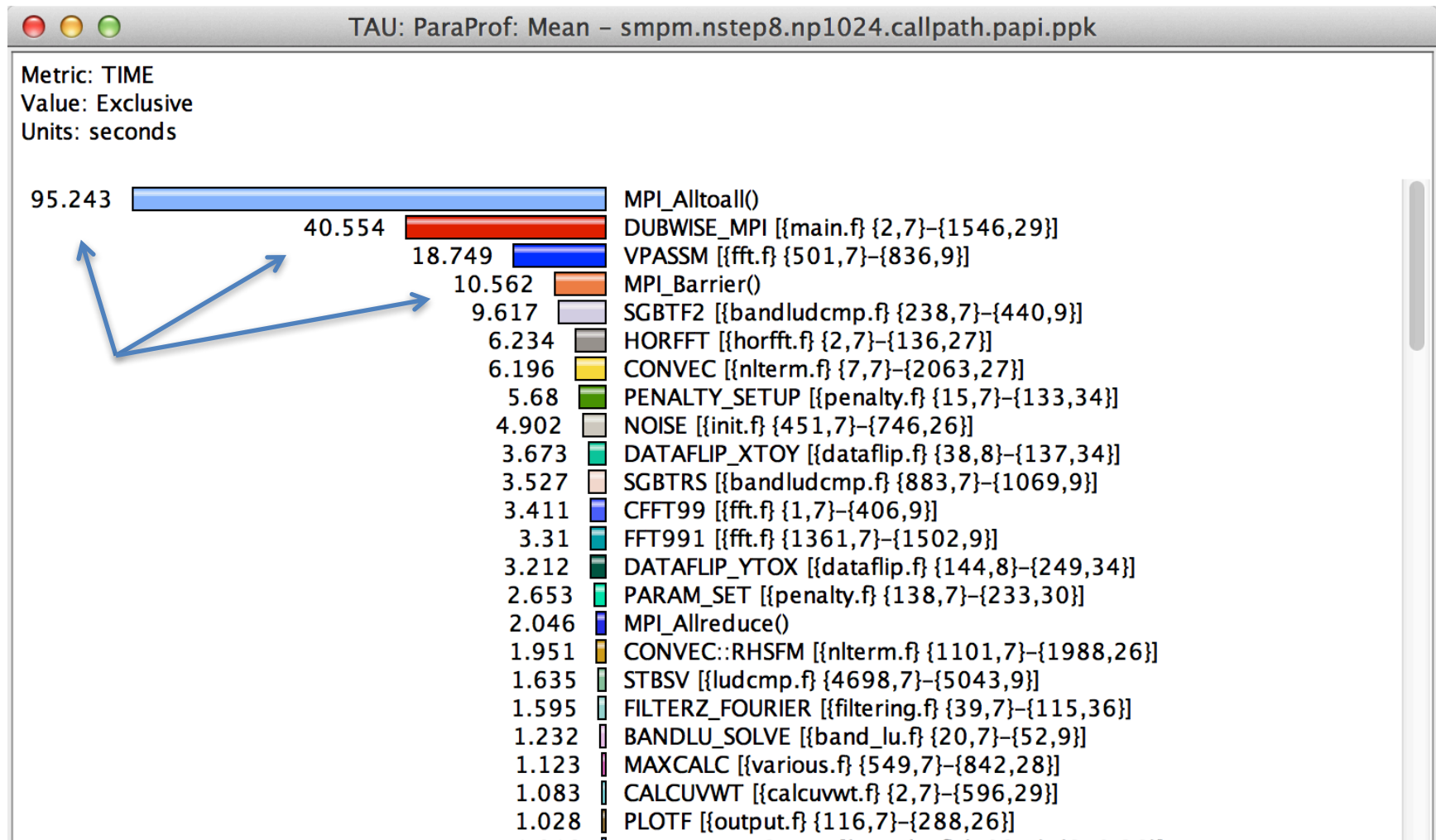
All levels support multiple metrics/counters

Multi-Language Debugging

- Identify the source location of a crash by unwinding the system callstack
- Identify memory errors (off-by-one, etc.) across language boundaries

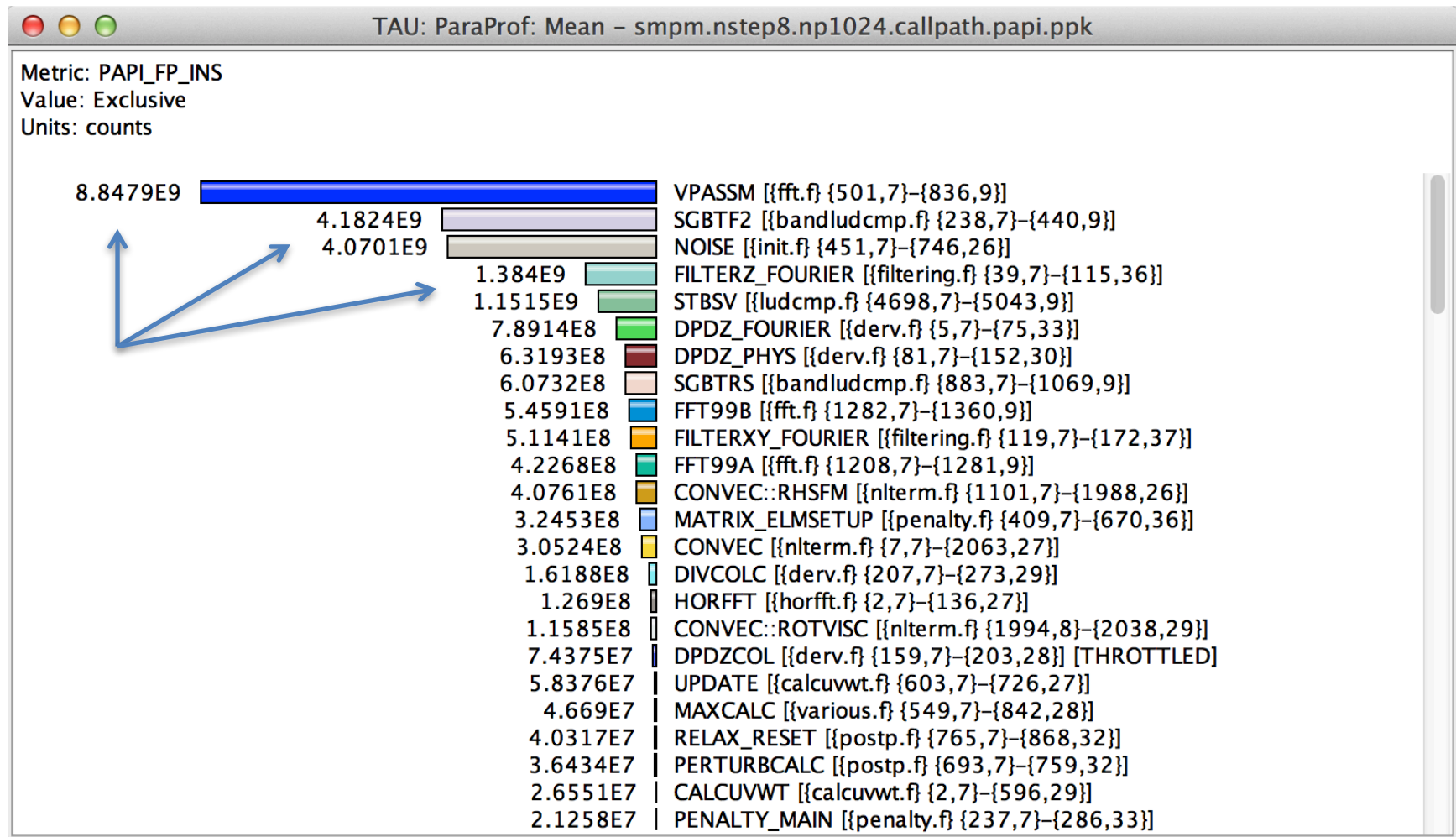


How Much Time per Code Region?



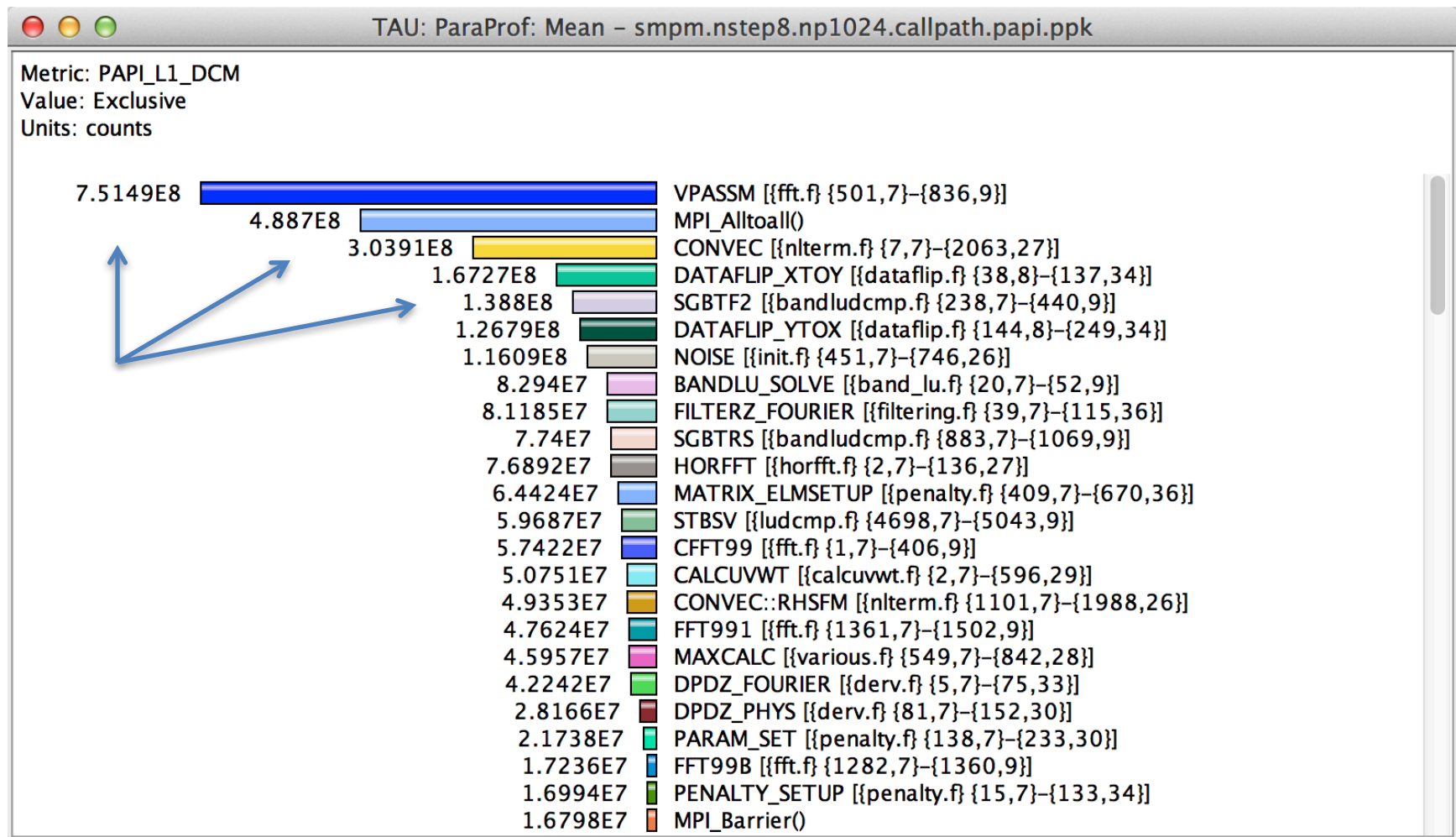
% **paraprof** (Click on label, e.g. "Mean" or "node 0")

How Many Instructions per Code Region?



% paraprof (Options → Select Metric... → Exclusive... → PAPI_FP_INS)

How Many L1/L2/L3 Cache Misses?



% paraprof (Options → Select Metric... → Exclusive... → PAPI_L1_DCM)

How Much Memory Does the Code Use?

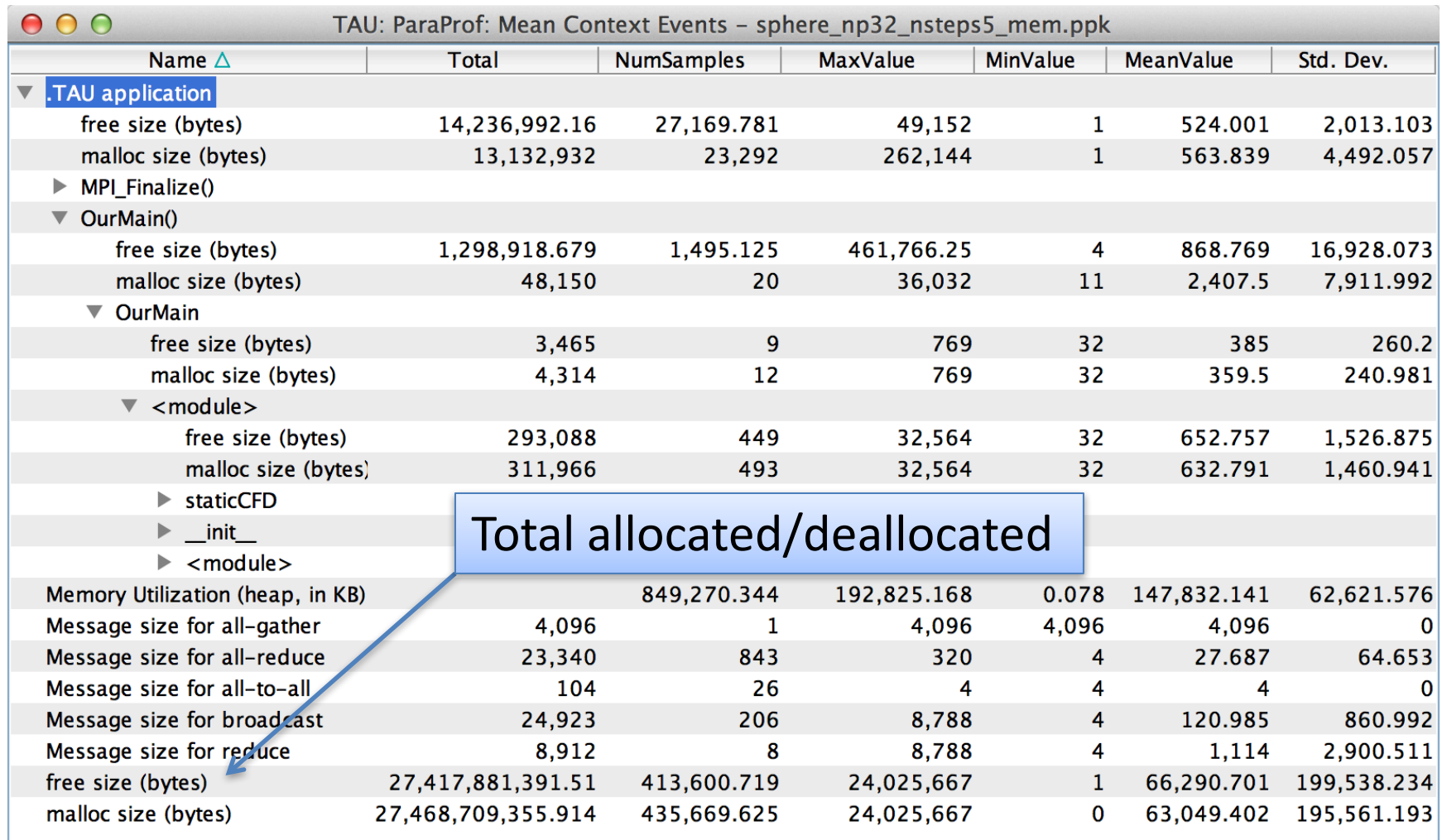
Name Δ	Total	NumSamples	MaxValue	MinValue	MeanValue	Std. Dev.
▼ TAU application						
free size (bytes)	14,236,992.16	27,169.781	49,152	1	524.001	2,013.103
malloc size (bytes)	13,132,932	23,292	262,144	1	563.839	4,492.057
▶ MPI_Finalize()						
▼ OurMain()						
free size (bytes)	1,298,918.679	1,495.125	461,766.25	4	868.769	16,928.073
malloc size (bytes)	48,150	20	36,032	11	2,407.5	7,911.992
▼ OurMain						
free size (bytes)	3,465	9	769	32	385	260.2
malloc size (bytes)	4,314	12	769	32	359.5	240.981
▼ <module>						
free size (bytes)	293,088	449	32,564	32	652.757	1,526.875
malloc size (bytes)	311,966	493	32,564	32	632.791	1,460.941
▶ staticCFD						
▶ __init__						
▶ <module>						
Memory Utilization (heap, in KB)		849,270.344	192,825.168	0.078	147,832.141	62,621.576
Message size for all-gather	4,096	1	4,096	4,096	4,096	0
Message size for all-reduce	23,340	843	320	4	27.687	64.653
Message size for all-to-all	104	26	4	4	4	0
Message size for broadcast	24,923	206	8,788	4	120.985	860.992
Message size for reduce	8,912	8	8,788	4	1,114	2,900.511
free size (bytes)	27,417,881,391.51	413,600.719	24,025,667	1	66,290.701	199,538.234
malloc size (bytes)	27,468,709,355.914	435,669.625	24,025,667	0	63,049.402	195,561.193

High-water mark



⌘ paraprof (Right-click label [e.g “node 0”] → Show Context Event Window)

How Much Memory Does the Code Use?

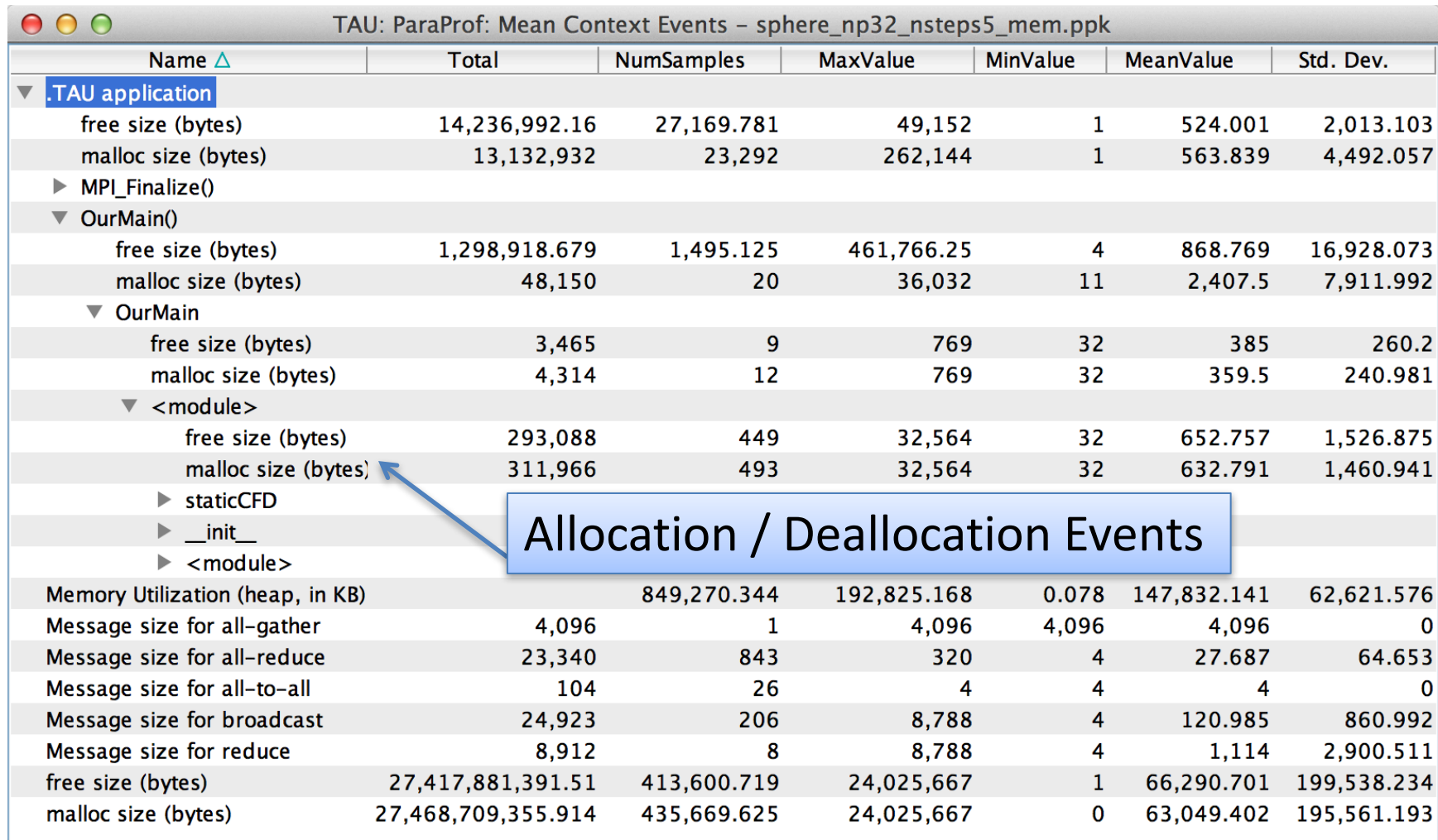


The screenshot shows a TAU context event window titled "TAU: ParaProf: Mean Context Events - sphere_np32_nsteps5_mem.ppk". The window displays a table with columns: Name, Total, NumSamples, MaxValue, MinValue, MeanValue, and Std. Dev. The data is organized into a tree structure under ".TAU application". A blue callout box with the text "Total allocated/deallocated" points to the "free size (bytes)" and "malloc size (bytes)" rows at the bottom of the table.

Name	Total	NumSamples	MaxValue	MinValue	MeanValue	Std. Dev.
▼ .TAU application						
free size (bytes)	14,236,992.16	27,169.781	49,152	1	524.001	2,013.103
malloc size (bytes)	13,132,932	23,292	262,144	1	563.839	4,492.057
▶ MPI_Finalize()						
▼ OurMain()						
free size (bytes)	1,298,918.679	1,495.125	461,766.25	4	868.769	16,928.073
malloc size (bytes)	48,150	20	36,032	11	2,407.5	7,911.992
▼ OurMain						
free size (bytes)	3,465	9	769	32	385	260.2
malloc size (bytes)	4,314	12	769	32	359.5	240.981
▼ <module>						
free size (bytes)	293,088	449	32,564	32	652.757	1,526.875
malloc size (bytes)	311,966	493	32,564	32	632.791	1,460.941
▶ staticCFD						
▶ __init__						
▶ <module>						
Memory Utilization (heap, in KB)		849,270.344	192,825.168	0.078	147,832.141	62,621.576
Message size for all-gather	4,096	1	4,096	4,096	4,096	0
Message size for all-reduce	23,340	843	320	4	27.687	64.653
Message size for all-to-all	104	26	4	4	4	0
Message size for broadcast	24,923	206	8,788	4	120.985	860.992
Message size for reduce	8,912	8	8,788	4	1,114	2,900.511
free size (bytes)	27,417,881,391.51	413,600.719	24,025,667	1	66,290.701	199,538.234
malloc size (bytes)	27,468,709,355.914	435,669.625	24,025,667	0	63,049.402	195,561.193

⌘ paraprof (Right-click label [e.g "node 0"] → Show Context Event Window)

Where is Memory Allocated / Deallocated?



Name Δ	Total	NumSamples	MaxValue	MinValue	MeanValue	Std. Dev.
▼ TAU application						
free size (bytes)	14,236,992.16	27,169.781	49,152	1	524.001	2,013.103
malloc size (bytes)	13,132,932	23,292	262,144	1	563.839	4,492.057
▶ MPI_Finalize()						
▼ OurMain()						
free size (bytes)	1,298,918.679	1,495.125	461,766.25	4	868.769	16,928.073
malloc size (bytes)	48,150	20	36,032	11	2,407.5	7,911.992
▼ OurMain						
free size (bytes)	3,465	9	769	32	385	260.2
malloc size (bytes)	4,314	12	769	32	359.5	240.981
▼ <module>						
free size (bytes)	293,088	449	32,564	32	652.757	1,526.875
malloc size (bytes)	311,966	493	32,564	32	632.791	1,460.941
▶ staticCFD						
▶ __init__						
▶ <module>						
Memory Utilization (heap, in KB)		849,270.344	192,825.168	0.078	147,832.141	62,621.576
Message size for all-gather	4,096	1	4,096	4,096	4,096	0
Message size for all-reduce	23,340	843	320	4	27.687	64.653
Message size for all-to-all	104	26	4	4	4	0
Message size for broadcast	24,923	206	8,788	4	120.985	860.992
Message size for reduce	8,912	8	8,788	4	1,114	2,900.511
free size (bytes)	27,417,881,391.51	413,600.719	24,025,667	1	66,290.701	199,538.234
malloc size (bytes)	27,468,709,355.914	435,669.625	24,025,667	0	63,049.402	195,561.193

⌘ paraprof (Right-click label [e.g “node 0”] → Show Context Event Window)

What are the I/O Characteristics?

TAU: ParaProf: Context Events for thread: n,c,t, 1,0,0 - samarc_obe_4p_iomem_cp.ppk


Name	Total	MeanValue	NumSamples	MinValue	MaxValue	Std. Dev.
▼ .TAU application						
▶ read()						
▶ fopen64()						
▶ fclose()						
▼ OurMain()						
malloc size	25,235	1,097.174	23	11	12,032	2,851.143
free size	22,707	1,746.692	13	11	12,032	3,660.642
▼ OurMain [{{wrapper.py}}{3}]						
▶ read()						
malloc size	3,877	323.083	12	32	981	252.72
free size	1,536	219.429	7	32	464	148.122
▶ fopen64()						
▶ fclose()						
▼ <module> [{{obe.py}}{8}]						
▼ writeRestartData [{{samarcInterface.py}}{145}]						
▼ samarcWriteRestartData						
▼ write()						
WRITE Bandwidth (MB/s) <file="samarc/restore.00002/nodes.00004/proc.00001">		74.565	117	0	2,156.889	246.386
WRITE Bandwidth (MB/s) <file="samarc/restore.00001/nodes.00004/proc.00001">		77.594	117	0	1,941.2	228.366
WRITE Bandwidth (MB/s)		76.08	234	0	2,156.889	237.551
Bytes Written <file="samarc/restore.00002/nodes.00004/proc.00001">	2,097,552	17,927.795	117	1	1,048,576	133,362.946
Bytes Written <file="samarc/restore.00001/nodes.00004/proc.00001">	2,097,552	17,927.795	117	1	1,048,576	133,362.946
Bytes Written	4,195,104	17,927.795	234	1	1,048,576	133,362.946
▶ open64()						


Write bandwidth per file

Bytes written to each file

⌘ **paraprof** (Right-click label [e.g "node 0"] → Show Context Event Window)

What are the I/O Characteristics?

Name 	Total	NumSamples	MaxValue	MinValue	MeanValue	Std. Dev.
▶ Incl						
▶ Initialize						
▶ LoadBodyEuler						
▶ LoadMesh						
MPI-IO Bytes Written	4,328,712	144	893,152	0	30,060.5	128,042.696
MPI-IO Write Bandwidth (MB/s)		144	196.86	0	3.421	16.87
▶ MPI_Allgatherv()						
▶ MPI_Bcast()						
▶ MPI_Comm_create()						
▶ MPI_File_close()						
▶ MPI_File_open()						
▶ MPI_File_write_all()						
▶ MPI_File_write_at()						
▶ MPI_Finalize()						
▶ MPI_Gather()						
▶ MPI_Gatherv()						



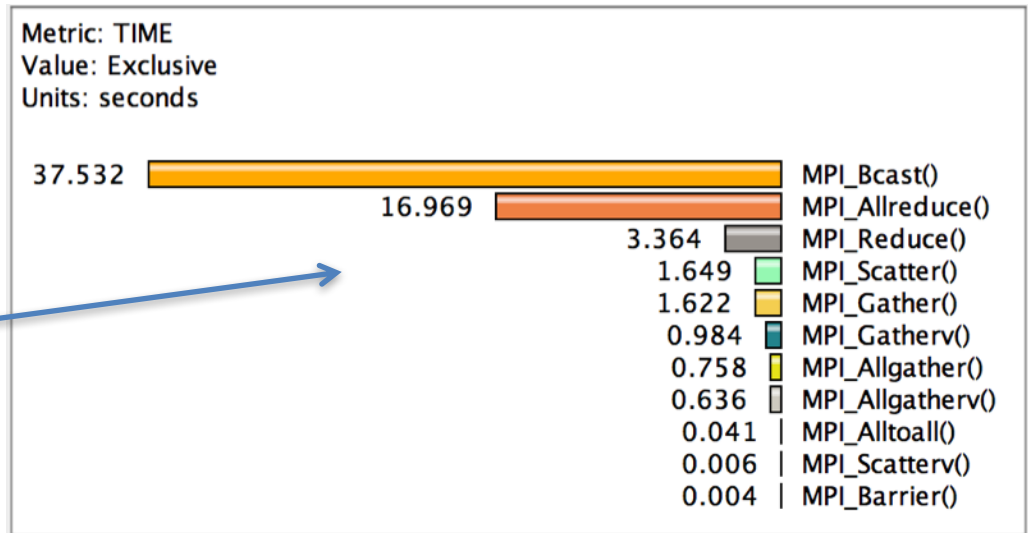
⌘ **paraprof** (Right-click label [e.g “node 0”] → Show Context Event Window)

How Much Time is Spent in Collectives?

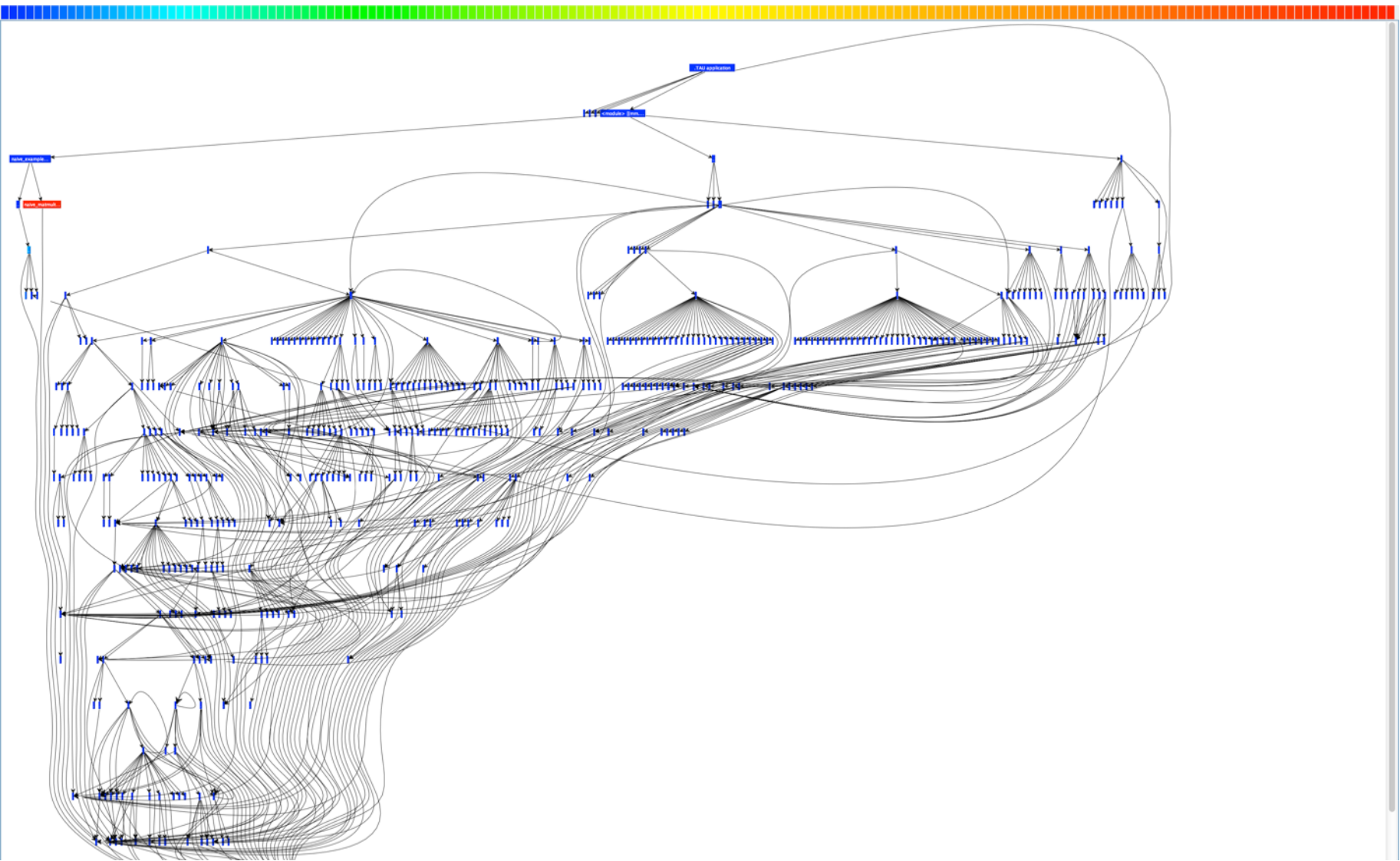
Name Δ	Total	Num...	MaxValue	MinValue	MeanValue	Std. Dev.
▶ MPI_Wait()						
▶ MPI_Waitall()						
Message size for all-gather	305,753,268	72	172,215,296	4	4,246,573.167	22,551,605.859
Message size for all-reduce	163,308	632	21,908	4	258.399	897.725
Message size for all-to-all	112	14	8	8	8	0
Message size for broadcast	692,208,045.5	3,346	18,117,620	0	206,876.284	1,284,673.036
Message size for gather	6,901,452.378	15.312	1,387,306.625	4	450,707.094	483,216.499
Message size for reduce	66,812	1,520	56	4	43.955	21.598
Message size for scatter	63,147.906	146	62,567.906	4	432.52	5,160.063

Message sizes

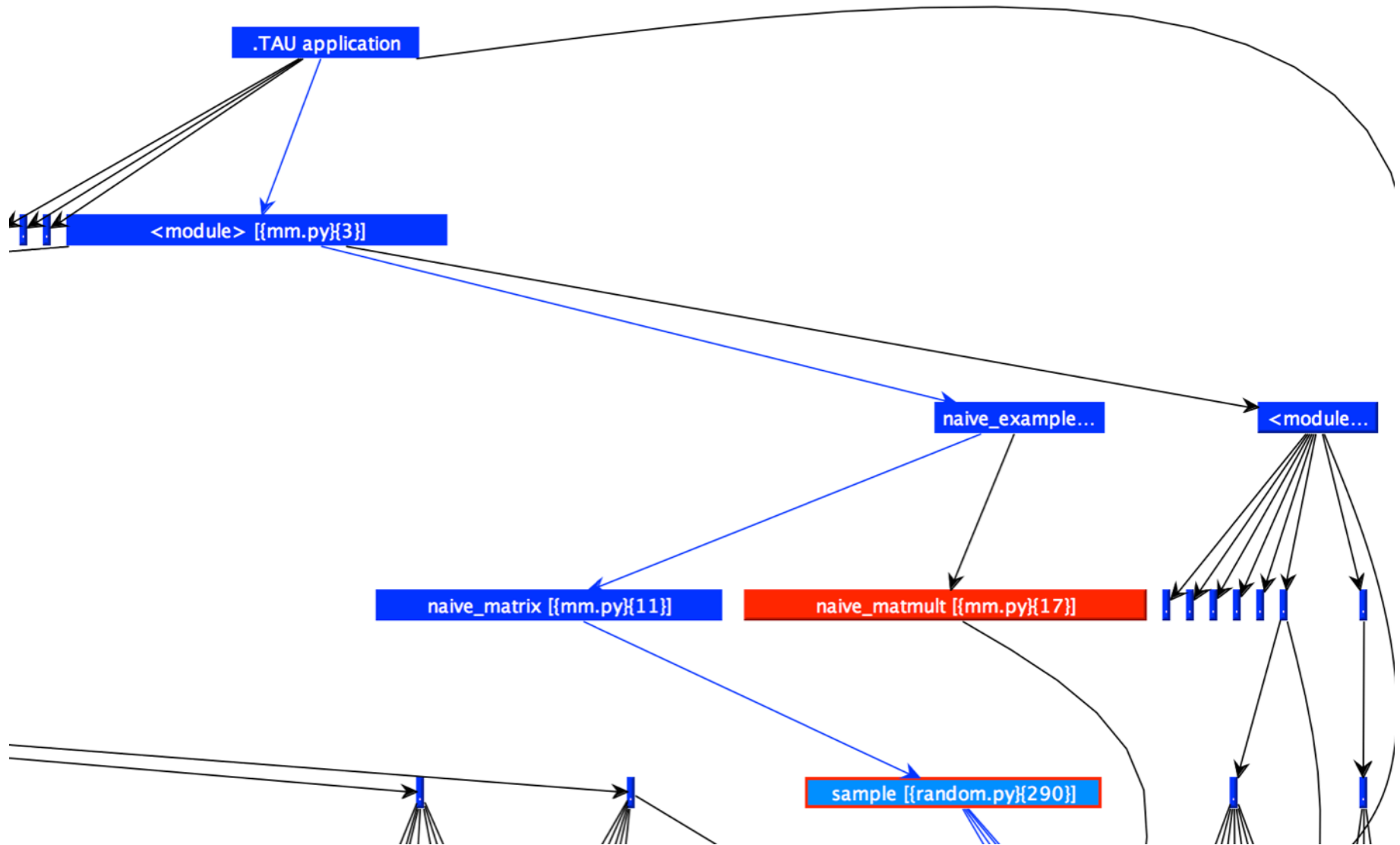
Time spent in collectives



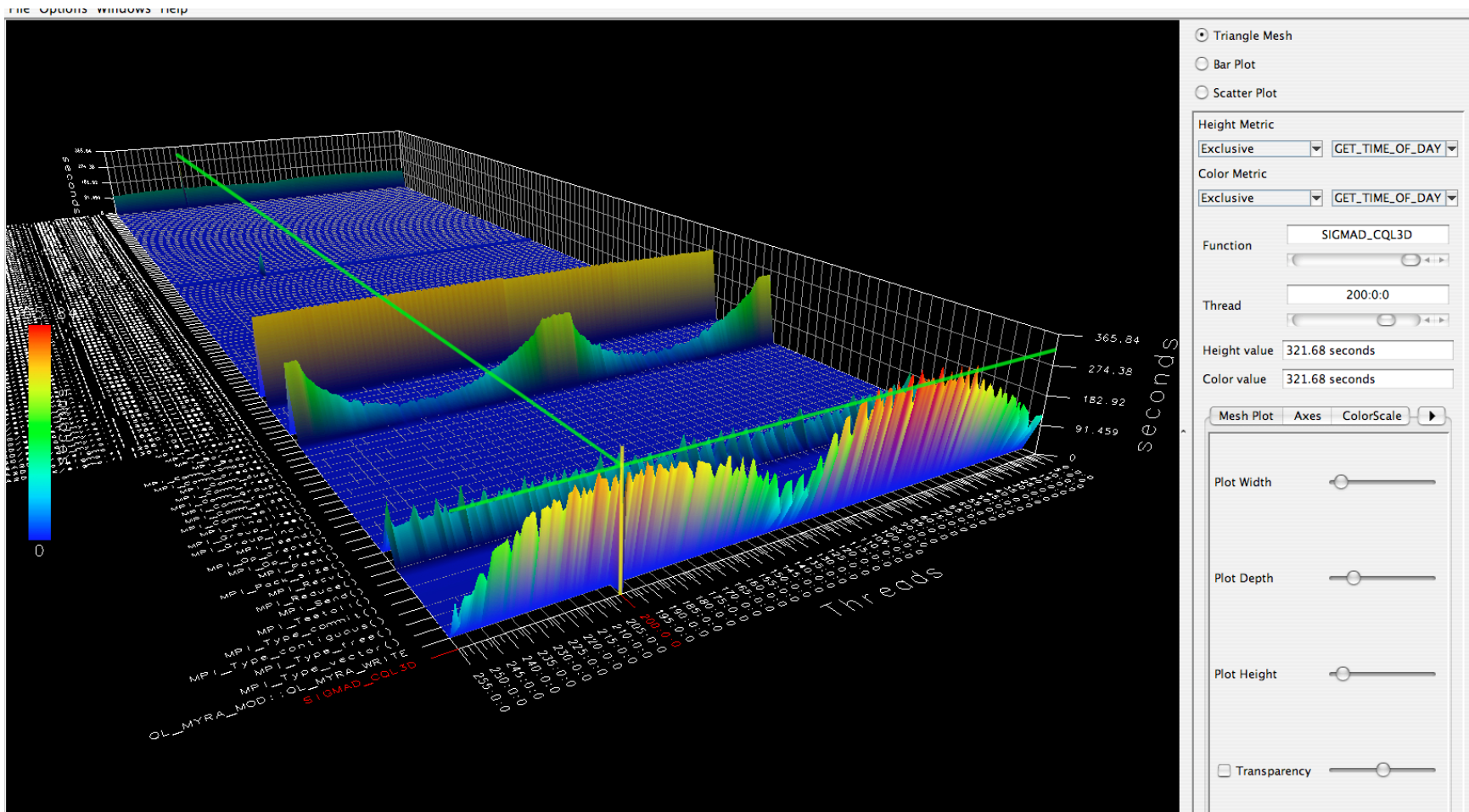
What is the Application Structure?



Application Callgraph Navigation

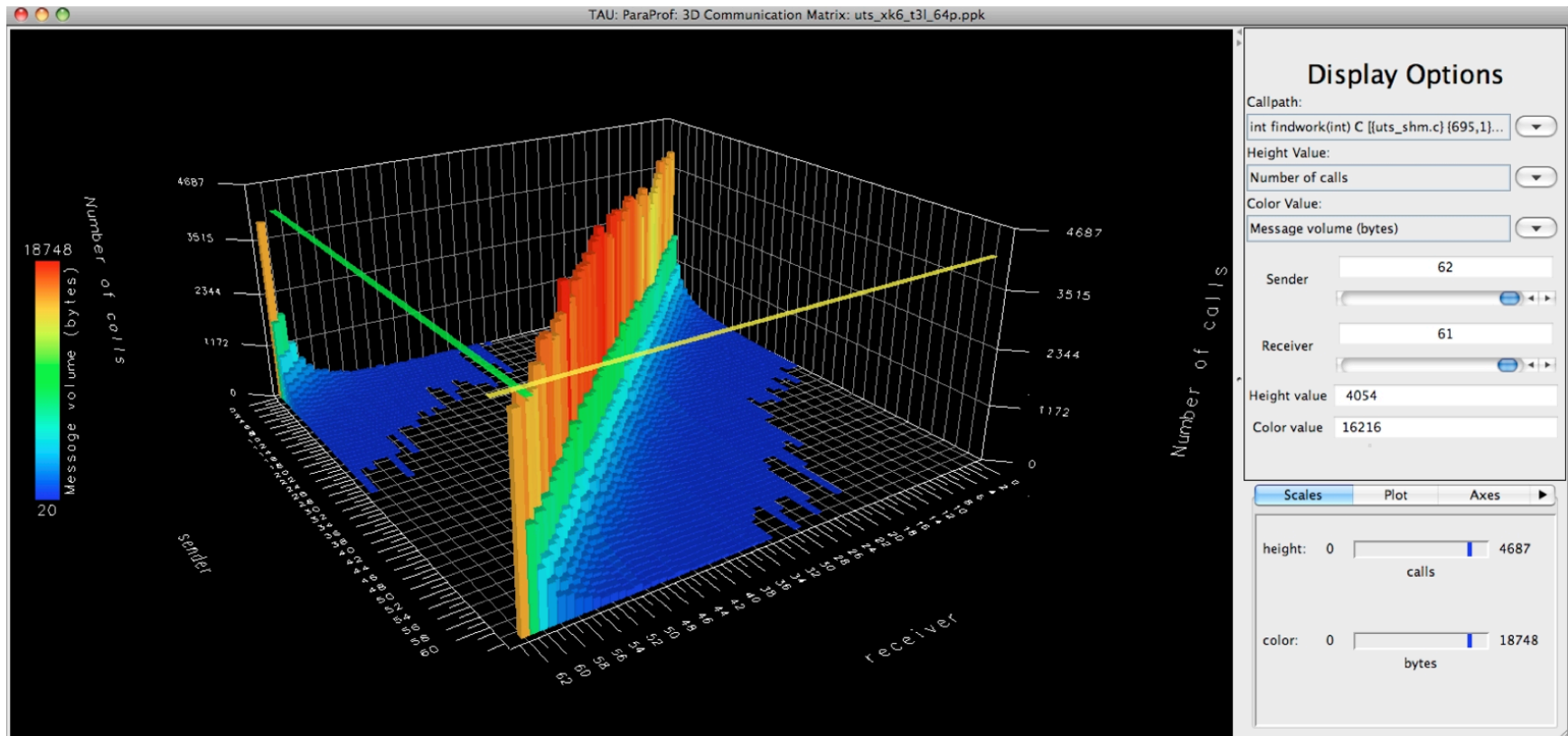


3D Profile Visualization



% **paraprof** (Windows → 3D Visualization)

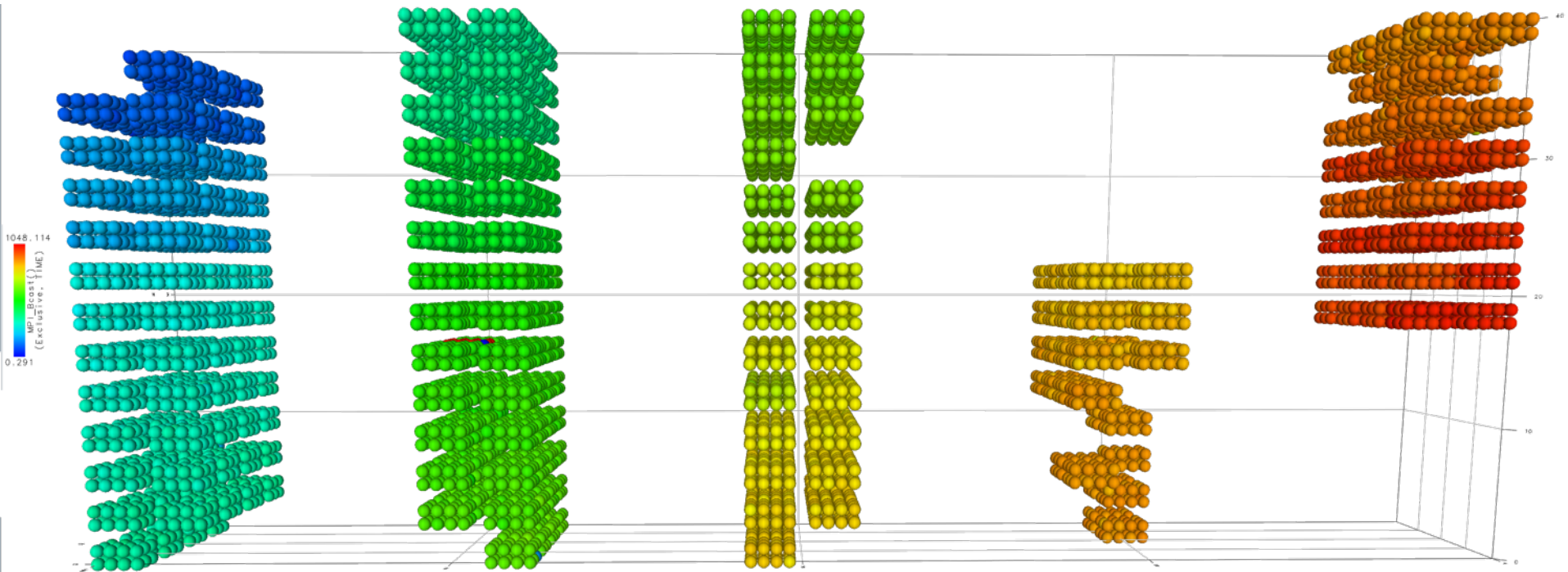
3D Communication Visualization



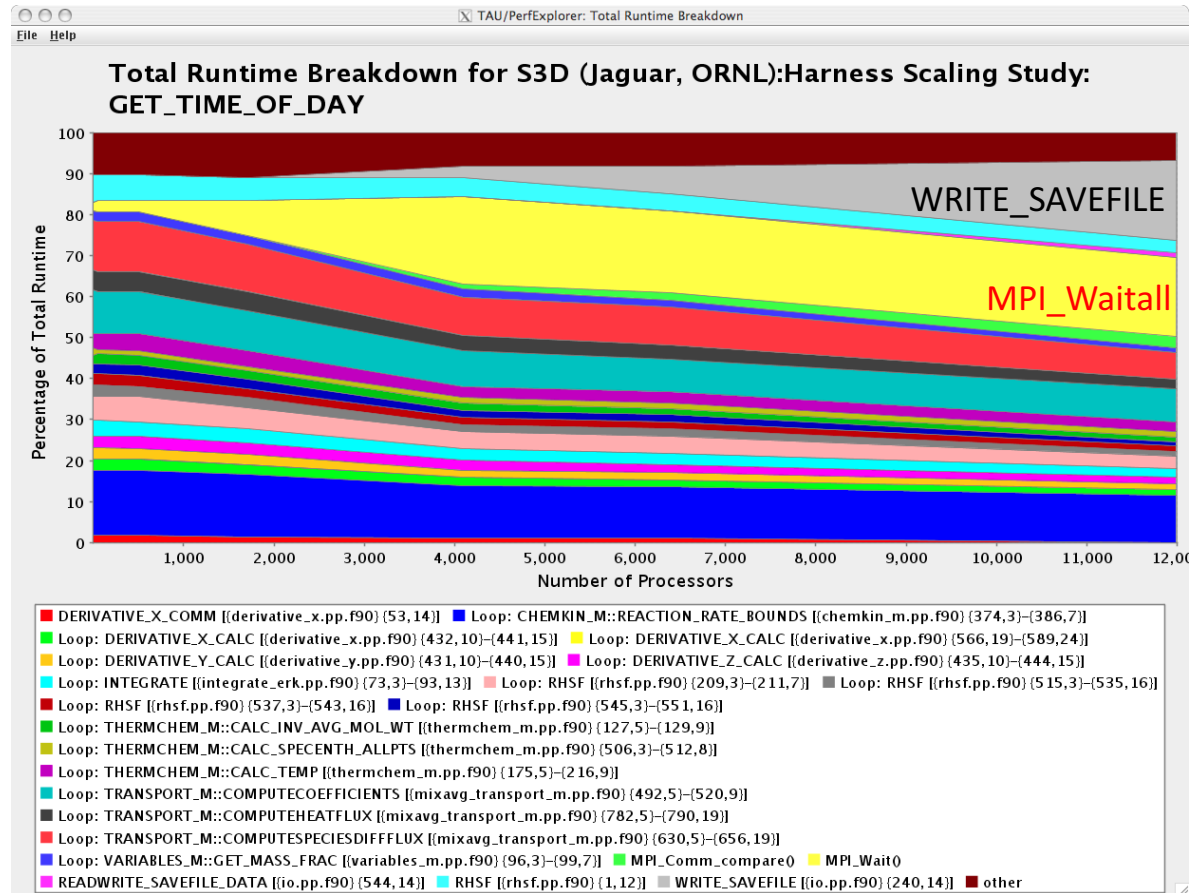
```
% qsub -env TAU_COMM_MATRIX=1 ...
```

```
% paraprof (Windows → 3D Communication Matrix)
```

3D Topology Visualization

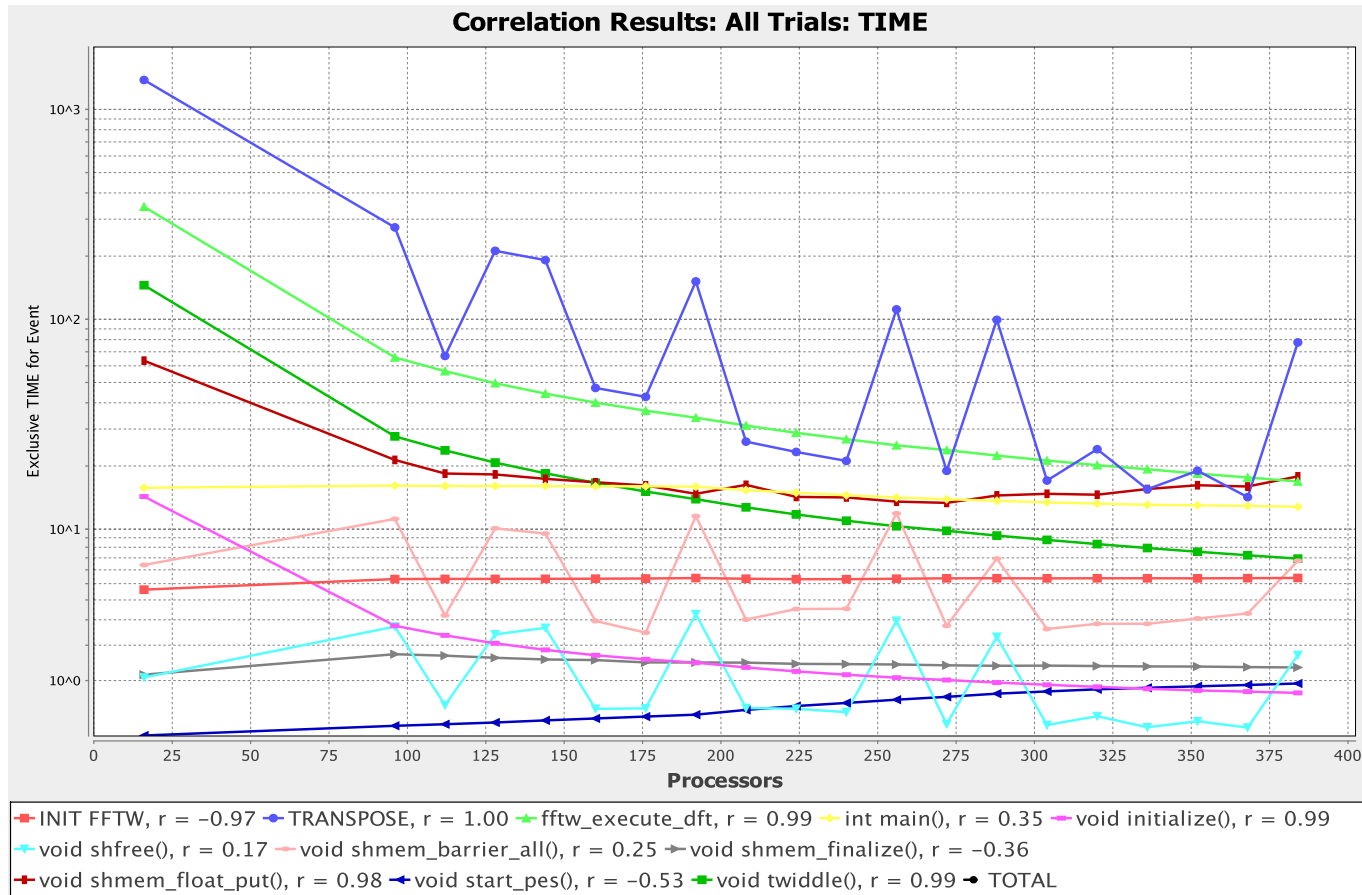


How Well Does Each Routine Scale?



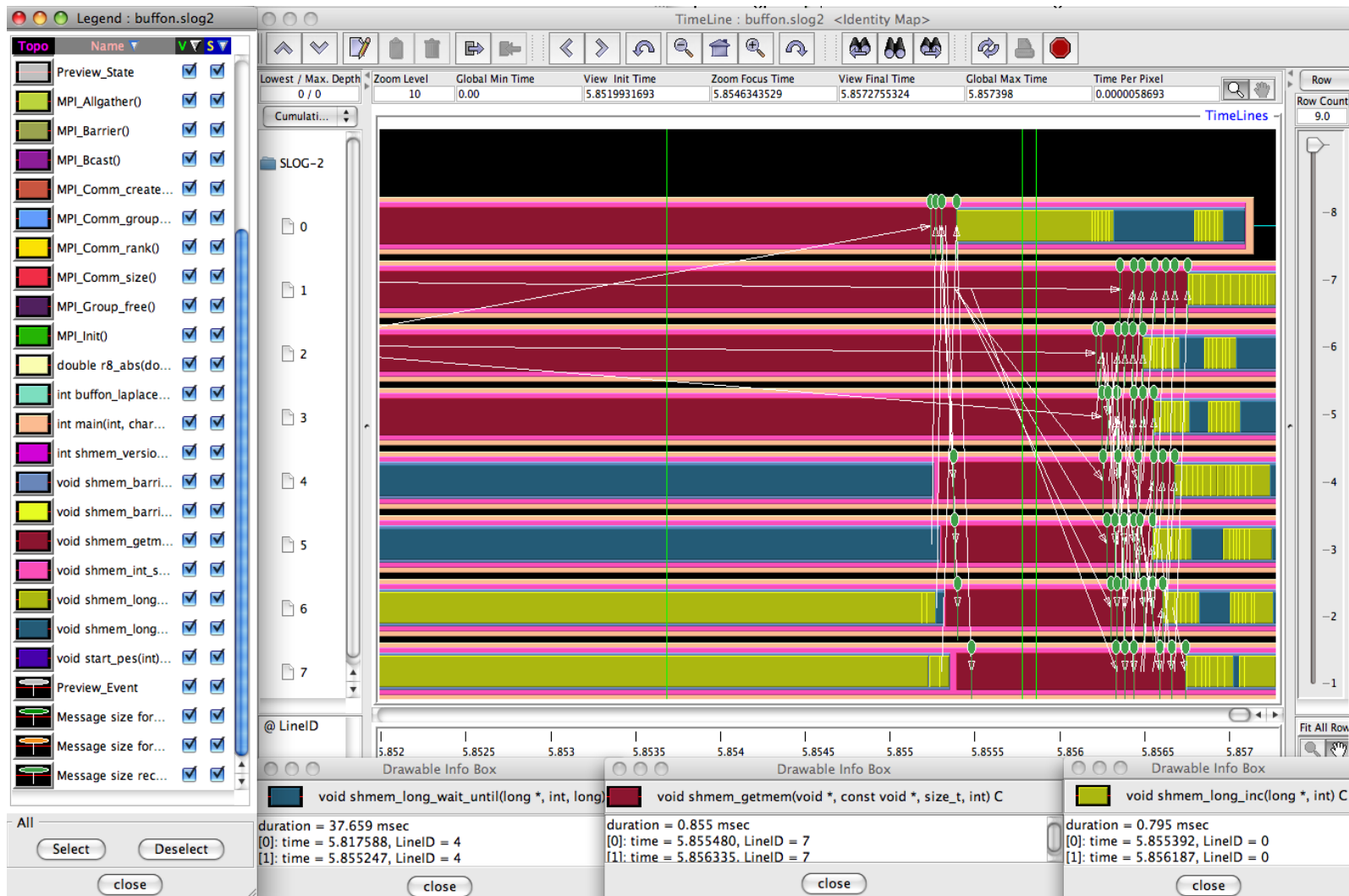
% perfexplorer (Charts → Runtime Breakdown)

Which Events Correlate with Runtime?

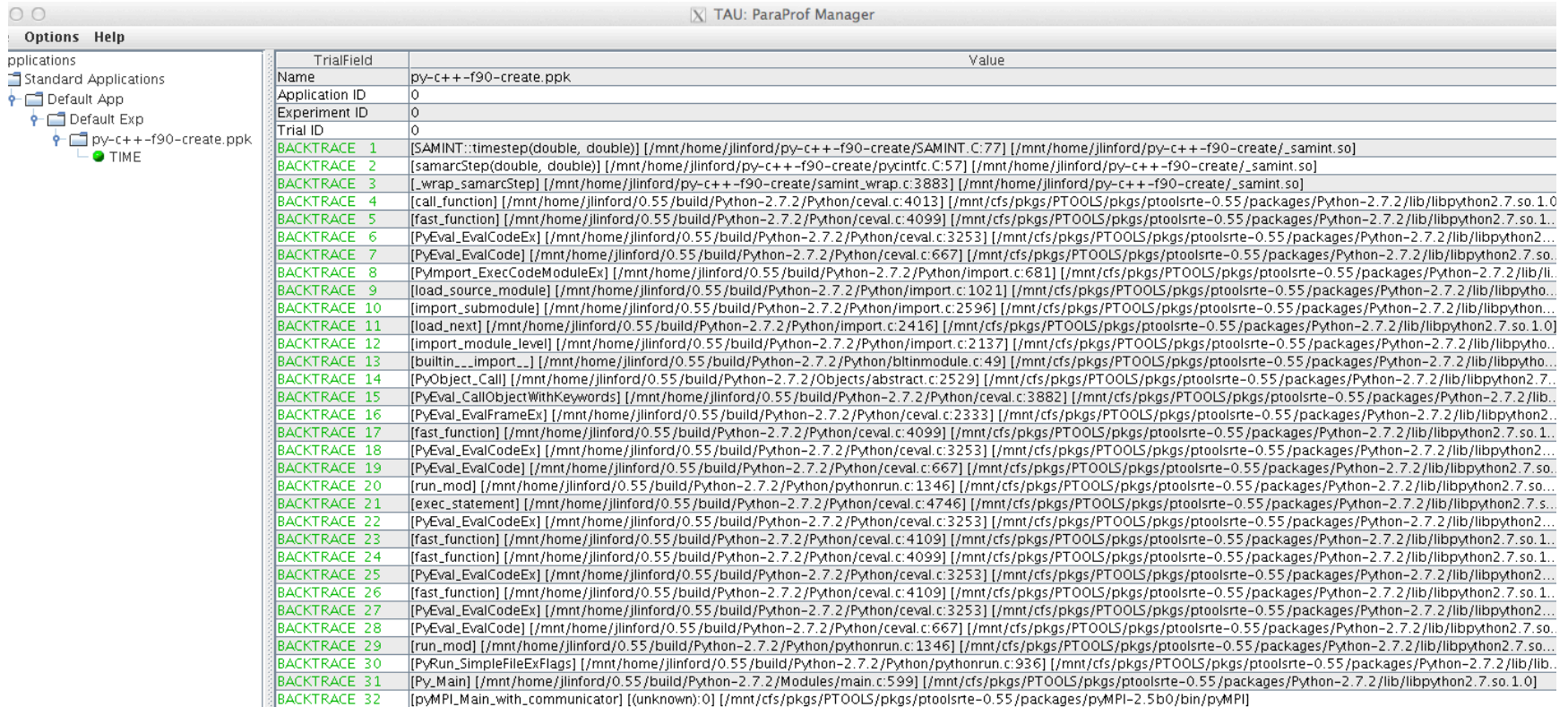


% **perfexplorer** (Charts → Correlate Events with Total Runtime)

When do Events Occur?



What Caused My Application to Crash?



The screenshot shows the TAU: ParaProf Manager interface. On the left, there is a sidebar with 'Options' and 'Help' buttons, and a tree view of applications. The main window displays a table of backtraces for a trial named 'py-c++-f90-create.ppk'. The table has columns for 'TrialField' and 'Value'. The backtraces are numbered from 1 to 32, with the first one being a SAMINT::timestep call and the last one being a pyMPI.Main call.

TrialField	Value
Name	py-c++-f90-create.ppk
Application ID	0
Experiment ID	0
Trial ID	0
BACKTRACE 1	[SAMINT::timestep(double, double)] [/mnt/home/jlinford/py-c++-f90-create/SAMINT.C:77] [/mnt/home/jlinford/py-c++-f90-create/_samint.so]
BACKTRACE 2	[samarcStep(double, double)] [/mnt/home/jlinford/py-c++-f90-create/pycintfc.C:57] [/mnt/home/jlinford/py-c++-f90-create/_samint.so]
BACKTRACE 3	[_wrap_samarcStep] [/mnt/home/jlinford/py-c++-f90-create/samint_wrap.c:3883] [/mnt/home/jlinford/py-c++-f90-create/_samint.so]
BACKTRACE 4	[call_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4013] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 5	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4099] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 6	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 7	[PyEval_EvalCode] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:667] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 8	[PyImport_ExecCodeModuleEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:681] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 9	[load_source_module] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:1021] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 10	[import_submodule] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:2596] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 11	[load_next] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:2416] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 12	[import_module_level] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:2137] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 13	[builtin___import_] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/bltinmodule.c:49] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 14	[PyObject_Call] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Objects/abstract.c:2529] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 15	[PyEval_CallObjectWithKeywords] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3882] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 16	[PyEval_EvalFrameEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:2333] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 17	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4099] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 18	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 19	[PyEval_EvalCode] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:667] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 20	[run_mod] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/pythonrun.c:1346] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 21	[exec_statement] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4746] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 22	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 23	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4109] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 24	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4099] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 25	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 26	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4109] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 27	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 28	[PyEval_EvalCode] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:667] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 29	[run_mod] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/pythonrun.c:1346] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 30	[PyRun_SimpleFileExFlags] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/pythonrun.c:936] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 31	[Py_Main] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Modules/main.c:599] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 32	[pyMPI_Main_with_communicator] [(unknown):0] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/pyMPI-2.5b0/bin/pyMPI]

```
% qsub -env TAU_TRACK_SIGNALS=1 ...
```

```
% paraprof
```


What Caused My Application to Crash?

Right-click to see source code



Name	Value
BACKTRACE 1	[SAMINT::timestep(double, double)] [/mnt/home/jlinford/py-c++-f90-create/SAMINT::timestep.c:10] [jlinford/py-c++-f90-create/_samint.so]
BACKTRACE 2	[samarcStep(double, double)] [/mnt/home/jlinford/py-c++-f90-create/pycintfc.C:10] [py-c++-f90-create/_samint.so]
BACKTRACE 3	[_wrap_samarcStep] [/mnt/home/jlinford/py-c++-f90-create/samint_wrap.c:3883] [/mnt/home/jlinford/py-c++-f90-create/_samint.so]
BACKTRACE 4	[call_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4013] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 5	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4099] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 6	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 7	[PyEval_EvalCode] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:667] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 8	[PyImport_ExecCodeModuleEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:681] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 9	[load_source_module] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:1021] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 10	[import_submodule] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:2596] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 11	[load_next] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:2416] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 12	[import_module_level] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/import.c:2137] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 13	[builtin__import_] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/builtinmodule.c:49] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 14	[PyObject_Call] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Objects/abstract.c:2529] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 15	[PyEval_CallObjectWithKeywords] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3882] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 16	[PyEval_EvalFrameEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:2333] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 17	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4099] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 18	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 19	[PyEval_EvalCode] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:667] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 20	[run_mod] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/pythonrun.c:1346] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 21	[exec_statement] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4746] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 22	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 23	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4109] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 24	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4099] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 25	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 26	[fast_function] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:4109] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 27	[PyEval_EvalCodeEx] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:3253] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 28	[PyEval_EvalCode] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/ceval.c:667] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 29	[run_mod] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/pythonrun.c:1346] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 30	[PyRun_SimpleFileExFlags] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Python/pythonrun.c:936] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 31	[Py_Main] [/mnt/home/jlinford/0.55/build/Python-2.7.2/Modules/main.c:599] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/Python-2.7.2/lib/libpython2.7.so.1.0]
BACKTRACE 32	[pyMPI_Main_with_communicator] [(unknown):0] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/pyMPI-2.5b0/bin/pyMPI]
BACKTRACE 33	[main] [(unknown):0] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/pyMPI-2.5b0/bin/pyMPI]
BACKTRACE 34	[_libc_start_main] [(unknown):0] [/lib64/libc-2.5.so]
BACKTRACE 35	[_start] [(unknown):0] [/mnt/cfs/pkgs/PTOOLS/pkgs/ptoolsrte-0.55/packages/pyMPI-2.5b0/bin/pyMPI]

What Caused My Application to Crash?

```
TAU: ParaProf: Source Browser: /mnt/home/jlinford/py-c++-f90-create/SAMINT.C
File Help
65  /*
66  *****
67  *
68  * Take a timestep - advance solution from "time" to "time + dt"
69  *
70  *****
71  */
72  void SAMINT::timestep(const double time,
73                      const double dt)
74  {
75      cout << "SAMINT::timestep()" << endl;
76      timestep_(time,dt);
77      int x = 4 / (4-4);
78      cout << " x = "<<x<<endl;
79  }
80
81  /*
82  *****
83  *
84  * Write data to output
85  * (visit, fieldview, or overgrid - set in samarc input file)
86  *
87  *****
88  */
89  void SAMINT::writePlotData(const double time,
90                           const int step)
91  {
92      cout << "SAMINT::writePlotData()" << endl;
93  }
```

← Error shown in ParaProf Source Browser

Performance Provenance

TAU: ParaProf Manager

Applications

- Standard Applications
 - Default (jdbc:h2:/Users/jlinfof/.ParaProf/perfdmf/perfdmf;AUTO_SERVER=TRUE)
 - ArmyPhasell (jdbc:postgresql://east01.paratools.com:5432/ArmyPhasell)
 - geos (jdbc:postgresql://east01.paratools.com:5432/geos)
 - GraviT (jdbc:postgresql://east01.paratools.com:5432/GraviT)
 - kppa (jdbc:postgresql://east01.paratools.com:5432/kppa)
 - KY05 (jdbc:postgresql://east01.paratools.com:5432/KY05)
 - KY06 (jdbc:postgresql://east01.paratools.com:5432/KY06)
- All Trials
 - Application-STELLAR-subset
 - Application-SAMCart.lightning
 - Experiment-scaling.O3_io_nonblocking_omp
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn1.np1.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn16.np16.1**
 - TIME
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn2.np2.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn24.np1024.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn24.np128.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn24.np2048.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn24.np256.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn24.np32.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn24.np512.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn24.np64.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn4.np4.1
 - SAMCart.scaling.O3_io_nonblocking_omp.craycnl.ppn8.np8.1
 - Experiment-scaling.O3
 - Experiment-scaling.O3_nonblocking
 - Experiment-O3_buffered_nonblocking_omp
 - All Trials
 - Application-SAMCart.x86_64
 - Application-SAMCart.mic
 - Experiment-scaling.O3
 - Experiment-scaling.O3_io_nonblocking.sampling
 - Experiment-scaling.O3_omp
 - Experiment-scaling.O3_novc.sampling
 - SAMCart.scaling.O3_novc.sampling.mic.np1
 - Experiment-scaling_omp_balanced
 - Experiment-scaling
 - Experiment-scaling_omp_compact
 - Experiment-scaling.O3_io_nonblocking_omp
 - Experiment-scaling.baseline
 - Experiment-scaling.O3_io_nonblocking
 - SAMCart.scaling.O3_io_nonblocking.mic.np1
 - SAMCart.scaling.O3_io_nonblocking.mic.np2
 - SAMCart.scaling.O3_io_nonblocking.mic.np4
 - SAMCart.scaling.O3_io_nonblocking.mic.np8
 - SAMCart.scaling.O3_io_nonblocking.mic.np16
 - SAMCart.scaling.O3_io_nonblocking.mic.np32
 - SAMCart.scaling.O3_io_nonblocking.mic.np64
 - SAMCart.scaling.O3_io_nonblocking.mic.np128
 - All Trials
 - Application-SAMCart.craycnl
- perfexplorer_working (jdbc:h2:/Users/jlinfof/.ParaProf/perfexplorer_working;AUTO_SERVER=TRUE)

TrialField	Value
Name	SAMCart.scaling.O3_io_nonblocking_omp.cr...
Application ID	0
Experiment ID	0
Trial ID	321
data_source	0
node_count	16
contexts_per_node	1
threads_per_context	1
total_threads	16
Application	SAMCart.lightning
CPU Cores	12
CPU MHz	2701.000
CPU Type	Intel(R) Xeon(R) CPU E5-2697 v2 @ 2.70GHz
CPU Vendor	GenuineIntel
CWD	/p/work1/jlinfof/CART/craycnl_tau/16.1...
Cache Size	30720 KB
Command Line	./samcart-exec samarc/input.samarc.16proc
Executable	/var/opt/cray/alps/spool/2021621/samca...
Experiment	scaling.O3_io_nonblocking_omp
File Type Index	1
File Type Name	TAU profiles
Hostname	nid00687
Local Time	2015-08-07T21:12:52+00:00
MPI Processor Name	nid00687
Memory Size	66072168 kB
Node Name	nid00687
OS Machine	x86_64
OS Name	Linux
OS Release	3.0.101-0.311.1_0502.8394-cray_ari_c
OS Version	#1 SMP Wed Sep 10 04:09:26 UTC 2014
Starting Timestamp	1438981972561763
TAU Architecture	default
TAU Config	-arch=craycnl -papi=/opt/cray/papi/5.3...
TAU Makefile	/p/home/jlinfof/workspace/CART/tau2/c...
TAU MetaData Merge Time	4.9E-05 seconds
TAU Version	2.24.1-git
TAU_BFD_LOOKUP	on
TAU_CALLPATH	off
TAU_CALLPATH_DEPTH	2
TAU_CALLSITE_LIMIT	1
TAU_COMM_MATRIX	off
TAU_COMPENSATE	off
TAU_CUPTI_API	runtime
TAU_EBS_KEEP_UNRESOLVED_ADDR	off
TAU_IBM_BG_HWP_COUNTERS	off
TAU_MAX_THREADS	1
TAU_MEASURE_TAU	off
TAU_MEMDBG_PROTECT_ABOVE	off
TAU_MEMDBG_PROTECT_BELOW	off
TAU_MEMDBG_PROTECT_FREE	off
TAU_OPENMP_RUNTIME	on
TAU_OPENMP_RUNTIME_EVENTS	on
TAU_OPENMP_RUNTIME_STATES	off
TAU_PROFILE	on
TAU_PROFILE_FORMAT	nprofile

NGC Lunch and Learn

CASE STUDY: STRAND SOLVER

DoD CFD Applications



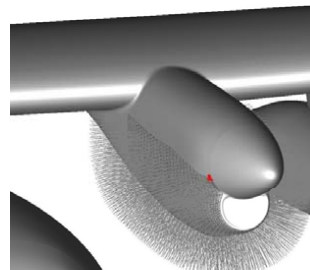
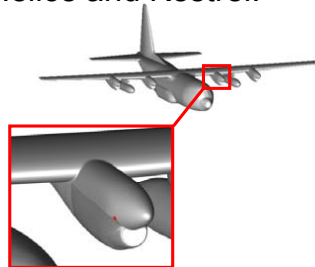
Strand Technology

Technology Drivers

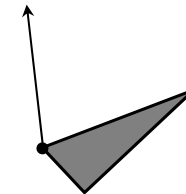
- Timeliness (automation of mesh generation)
- Timeliness (automation and scalability of domain connectivity)
- Timeliness/Physical accuracy (computational efficiency and scalability of aerodynamic solvers)
- Processor architecture (small memory footprint maps well to hierarchical memory architectures, e.g., multi-core, GPU)

CREATE-AV Example

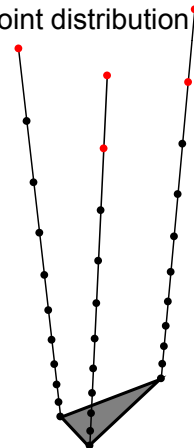
This is a new meshing paradigm introduced in 2007 by current members of the CREATE-AV technical staff. The technology is being matured in the Helios product and will be deployed through both Helios and Kestrel.



Strand
pointing vector



Strand
point distribution

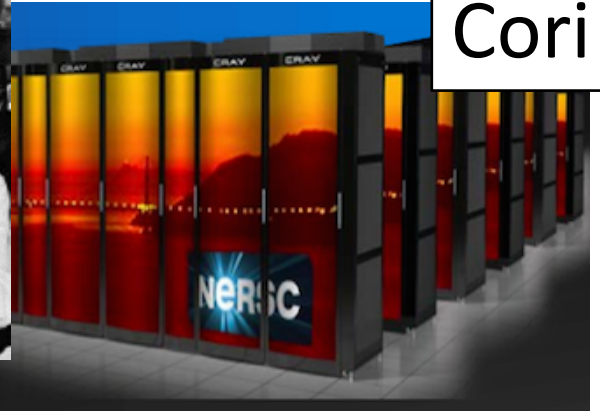


Target Platform: x86_64 + Xeon Phi

Armstrong [XC30]



Cori



Haise [iDataPlex]



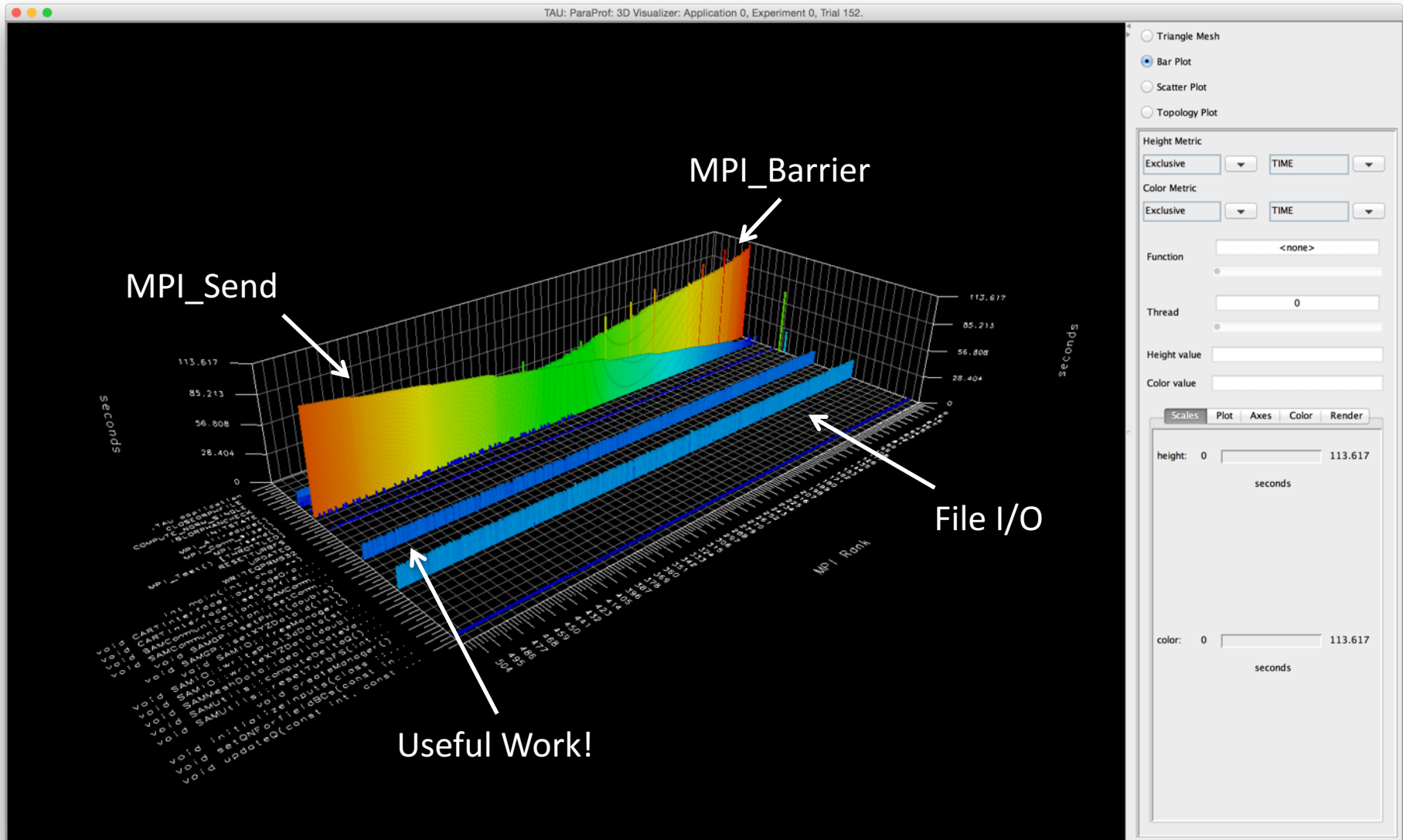
Lightning [XC30]



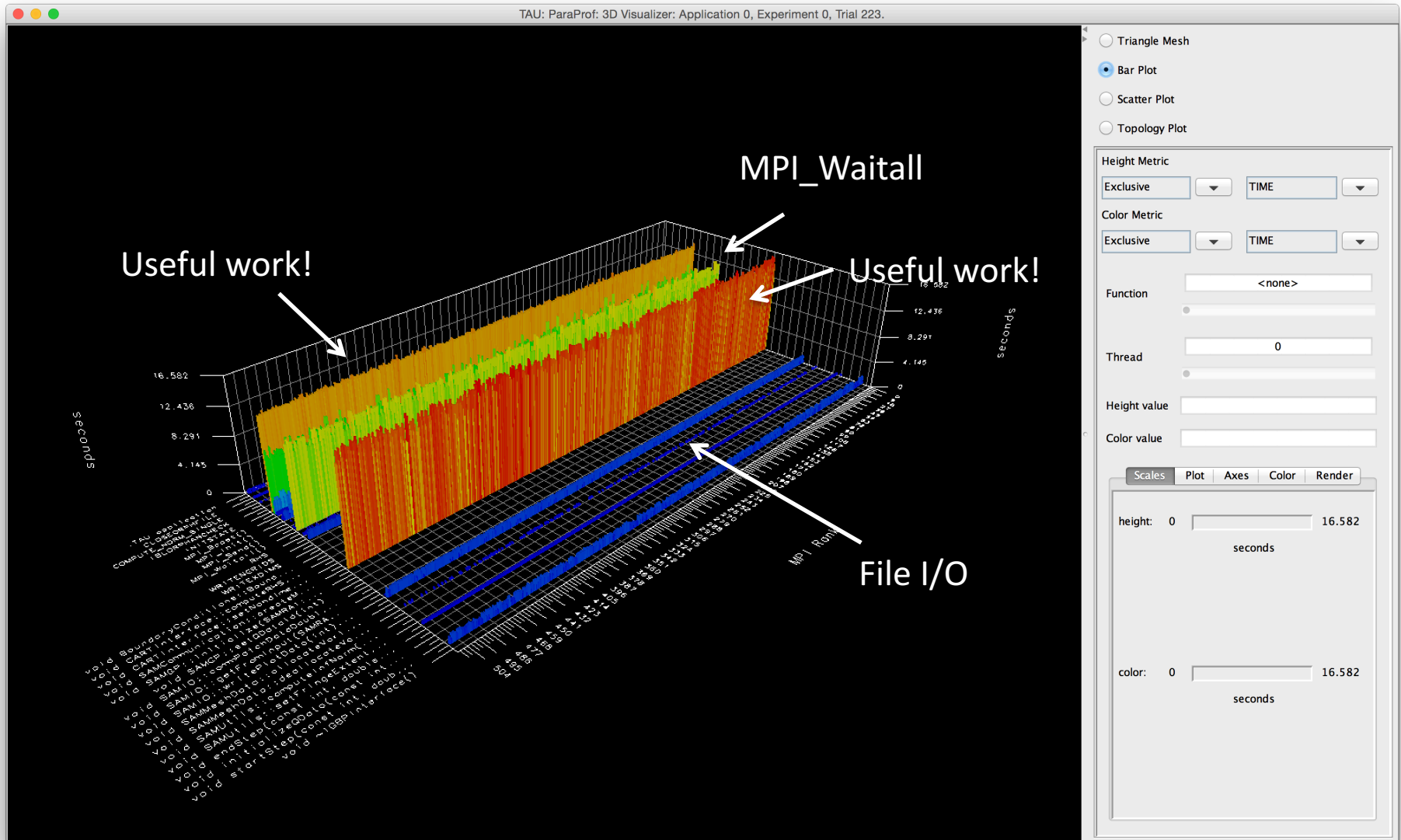
Kilrain [iDataPlex]



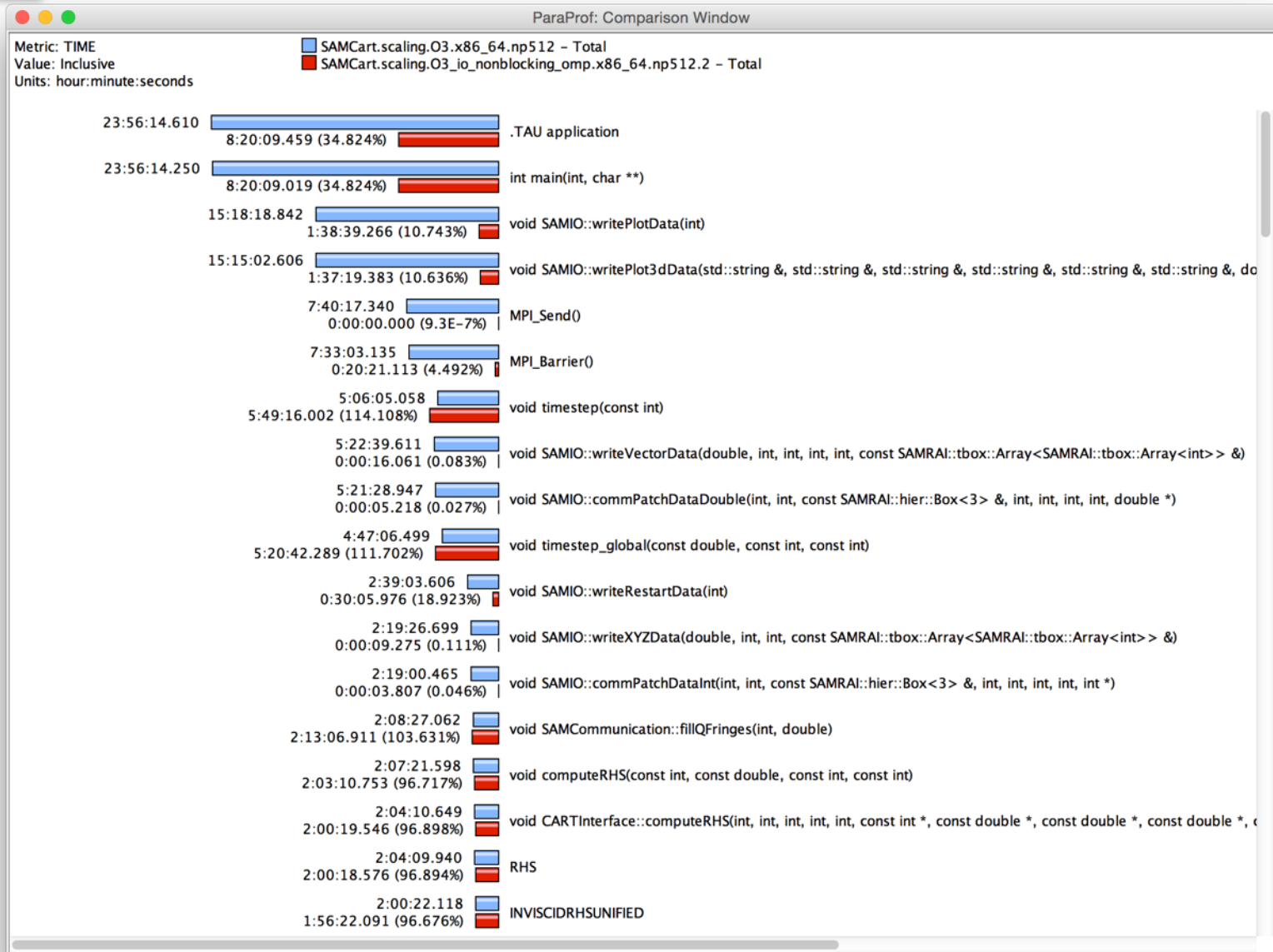
Initial Profile



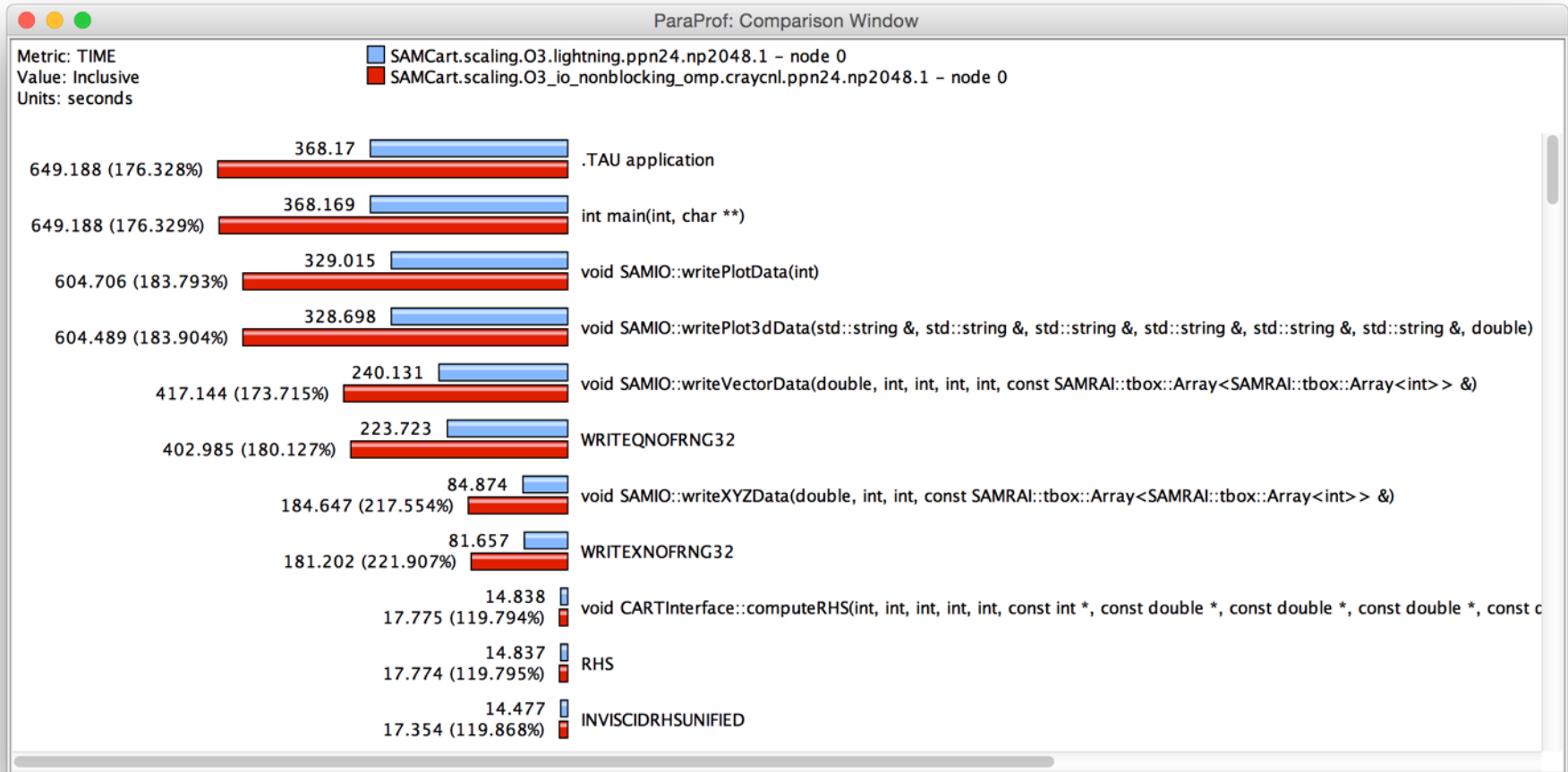
Nonblocking MPI and Optimized I/O



65% Runtime Reduction (~2x faster)

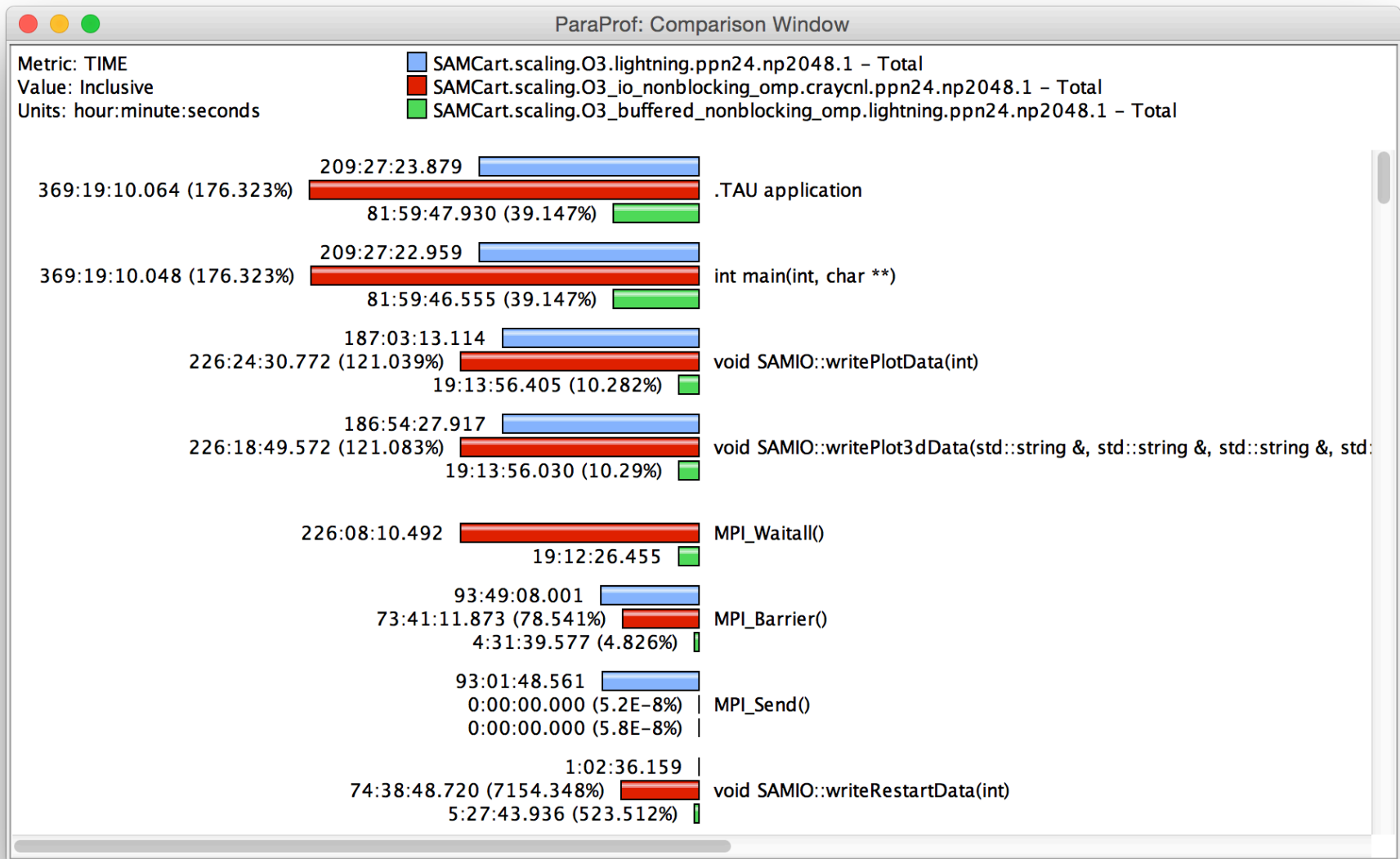


Cray XC30



Slower! What happened???

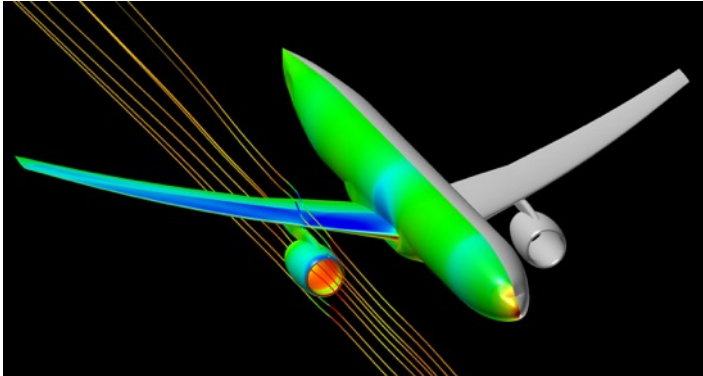
Use Appropriate Stripe & Buffered I/O



NGC Lunch and Learn

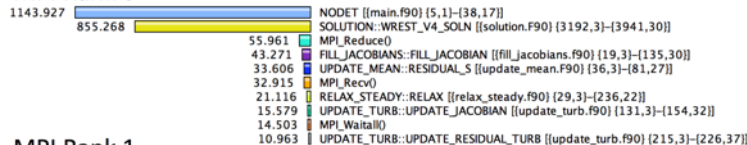
CASE STUDY: NASA FUN3D

FUN3D Wing-Body-Nacelle

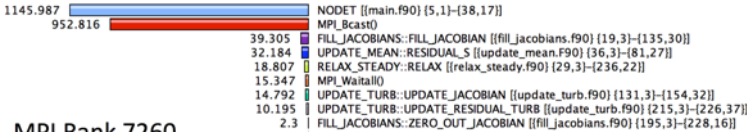


- 1,651,089,924 grid points
- 5,902,801,476 tetrahedral elements
- 1,310,290,264 prismatic elements
- 14,400 Ivy Bridge cores

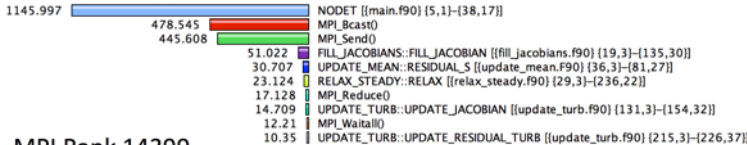
MPI Rank 0



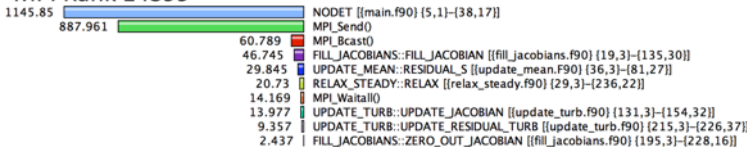
MPI Rank 1



MPI Rank 7260



MPI Rank 14399



MPI Rank 0

Name	Total	NumSamples	MaxValue	MinValue	MeanValue	Std. Dev.
Message size for all-gather	2,046,555,416	1,132	19,240,000	4	1,807,911.145	3,307,678.302
Message size for broadcast	39,795,349,570	196,653	75,171,056	1	202,363.298	352,969.91
Message size for gather	16,586,550,740	57,602	82,050,416	4	287,950.952	492,059.37
Message size for reduce	606,584	36,136	344	4	16.786	64.327
Message size received from all nodes	106,507,952,672	223,816	5,038,840	12	475,872.827	1,136,221.111
Message size received in wait	845,737,888	166,100	311,840	12	5,091.739	17,266.492
Message size sent to all nodes	769,221,120	126,660	327,680	20	6,073.118	19,893.424

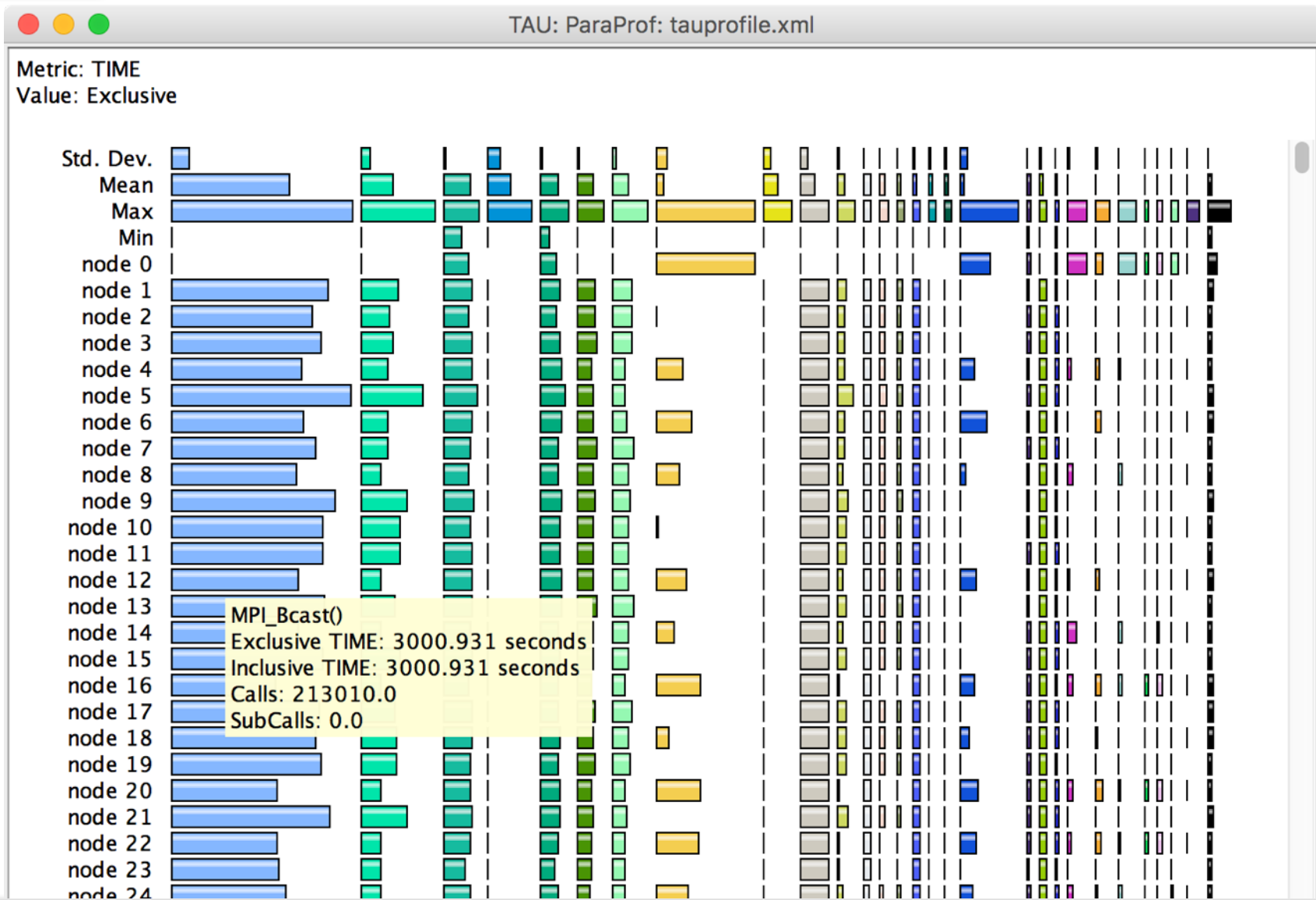
MPI Rank 1

Name	Total	NumSamples	MaxValue	MinValue	MeanValue	Std. Dev.
Message size for all-gather	2,046,555,416	1,132	19,240,000	4	1,807,911.145	3,307,678.302
Message size for broadcast	39,795,349,570	196,653	75,171,056	1	202,363.298	352,969.91
Message size for reduce	606,584	36,136	344	4	16.786	64.327
Message size received from all nodes	546,987,316	137,400	245,040	20	3,980.985	12,778.438
Message size received in wait	546,987,316	137,400	245,040	20	3,980.985	12,778.438
Message size sent to all nodes	503,971,872	105,844	4,637,520	20	4,761.459	20,996.819
Message size sent to node 0	7,420,032	4	4,637,520	927,504	1,855,008	1,606,484.052

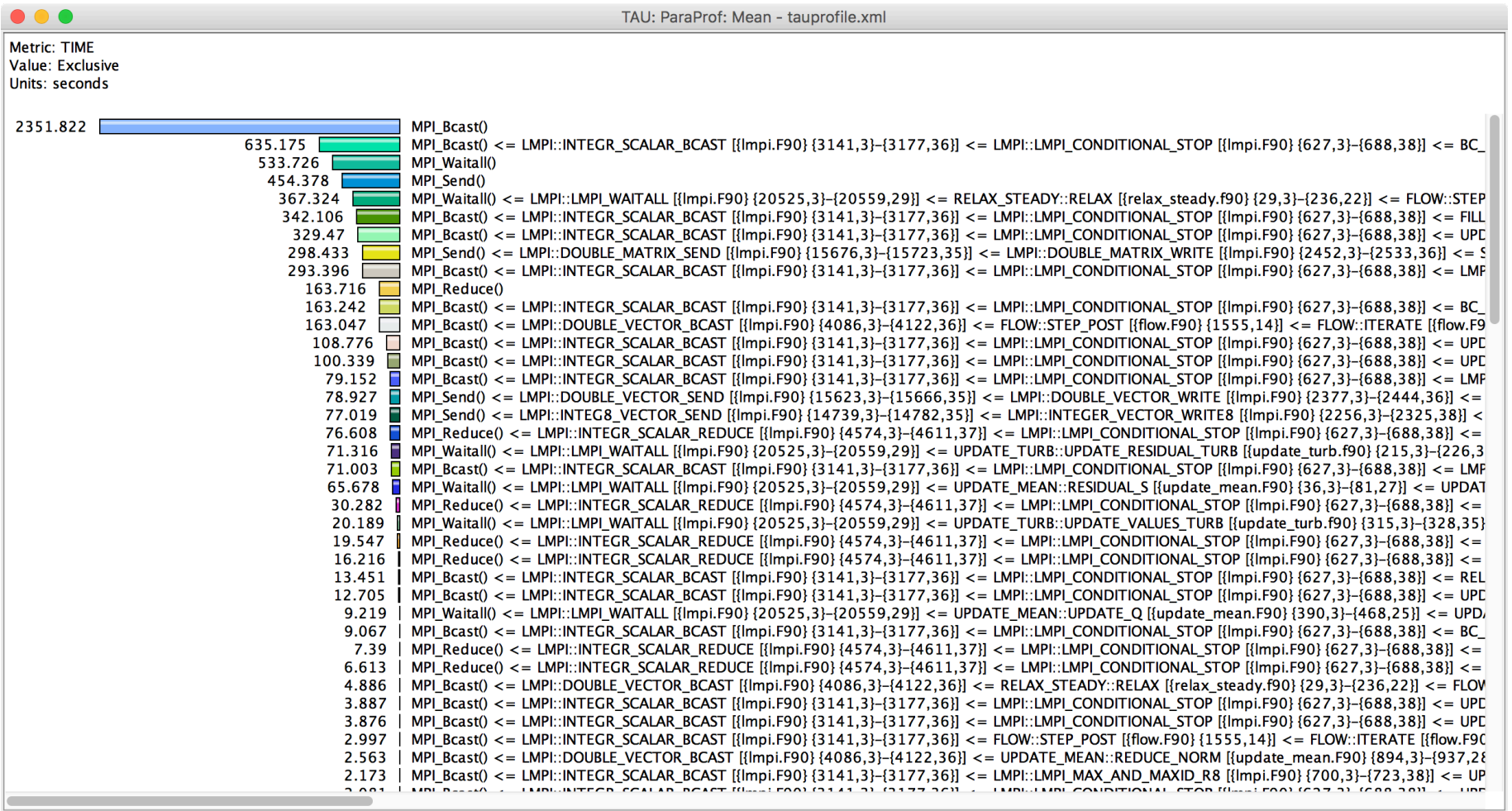
MPI Rank 14399

Name	Total	NumSamples	MaxValue	MinValue	MeanValue	Std. Dev.
Message size for all-gather	2,046,555,416	1,132	19,240,000	4	1,807,911.145	3,307,678.302
Message size for broadcast	39,795,349,570	196,653	75,171,056	1	202,363.298	352,969.91
Message size for reduce	606,584	36,136	344	4	16.786	64.327
Message size received from all nodes	495,423,720	112,100	199,280	20	4,419.48	13,658.624
Message size received in wait	495,423,720	112,100	199,280	20	4,419.48	13,658.624
Message size sent to all nodes	455,176,512	122,464	4,632,120	20	3,716.819	18,857.997
Message size sent to node 0	7,411,392	4	4,632,120	926,424	1,852,848	1,604,613.437

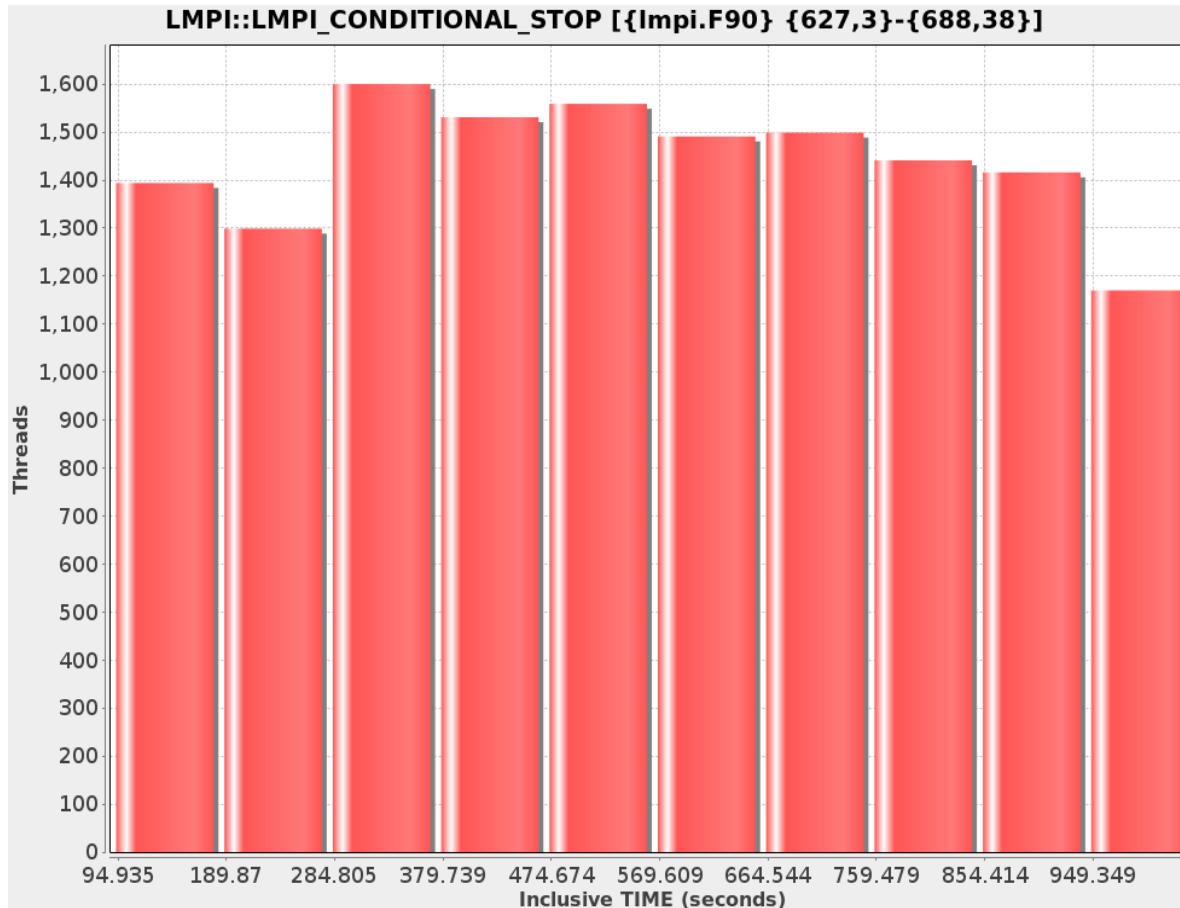
MPI_Bcast in Unstacked Bar Chart



Callpath Profile



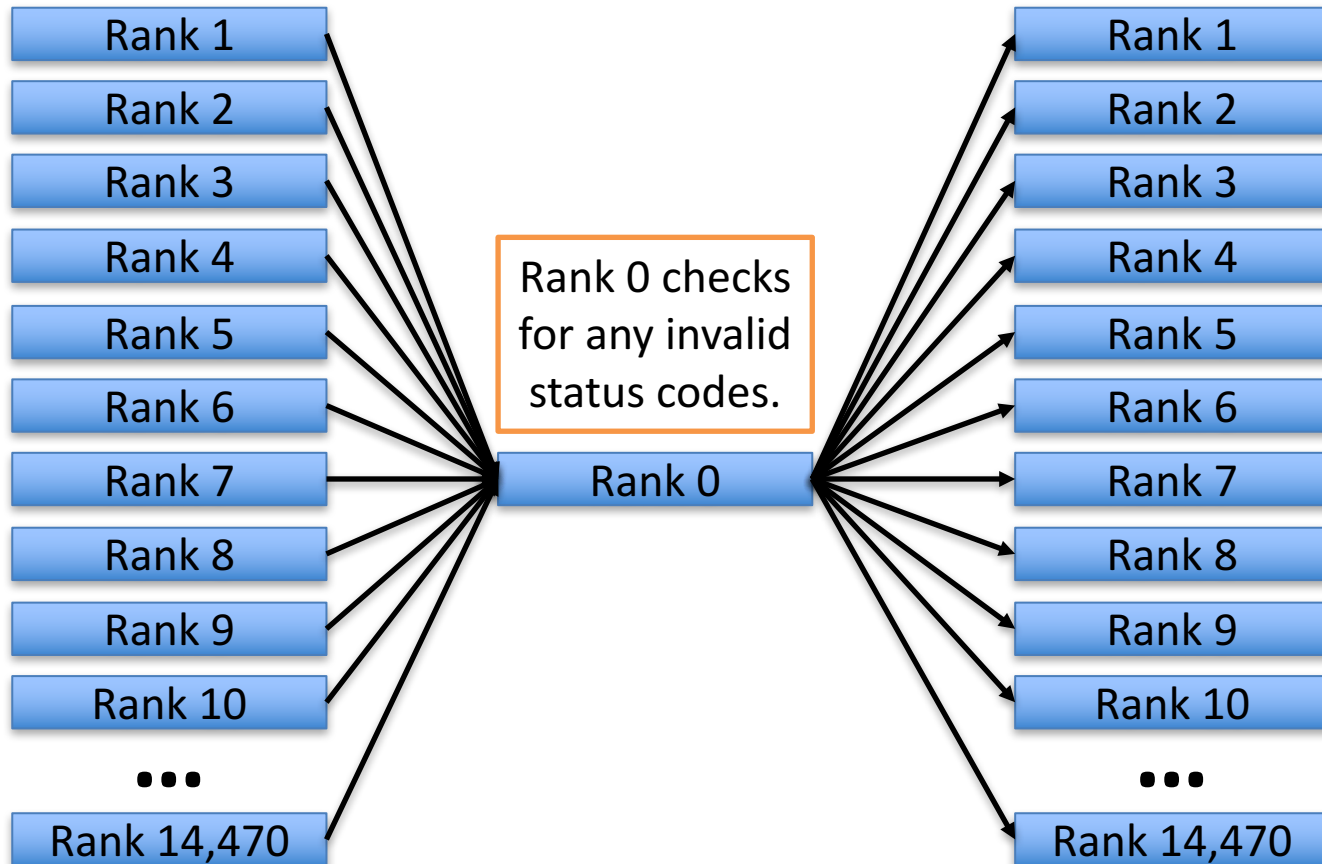
Broad histogram indicates bottleneck.



What's going on here?

All non-zero ranks send a status code to Rank 0.

Rank 0 broadcasts global status code to all ranks.



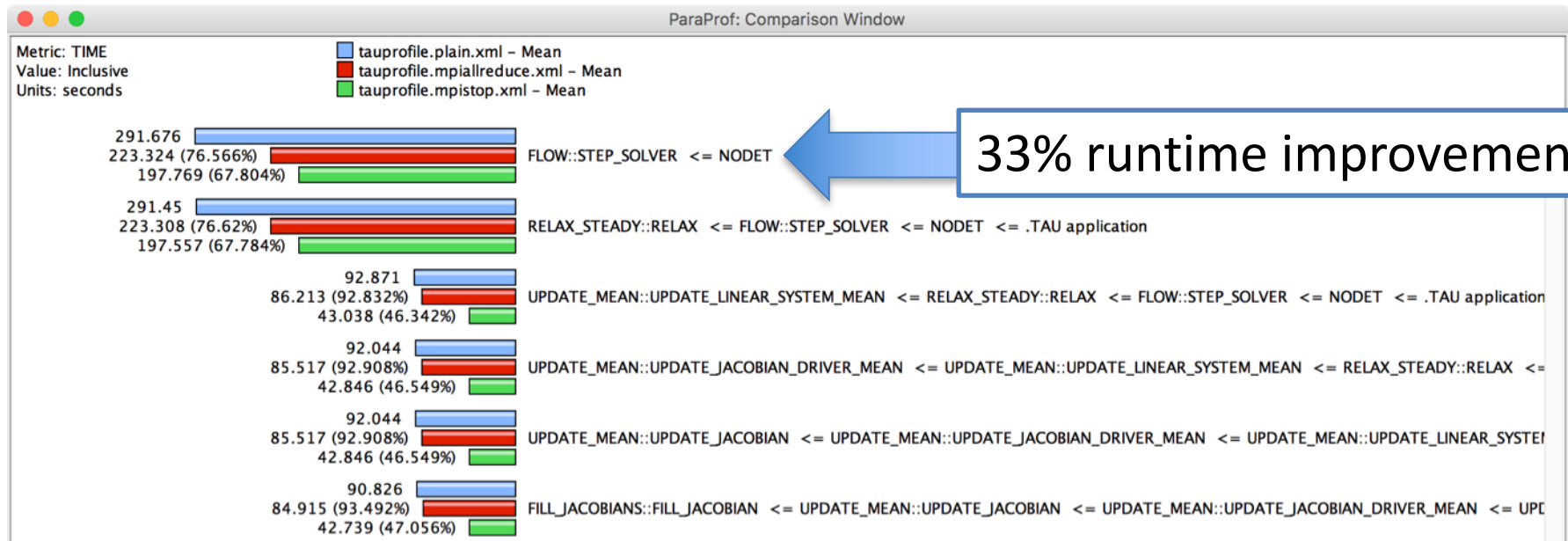
Two Solutions

MPI_Allreduce()

- $O(\log_2 N)$ vs. $O(N)$

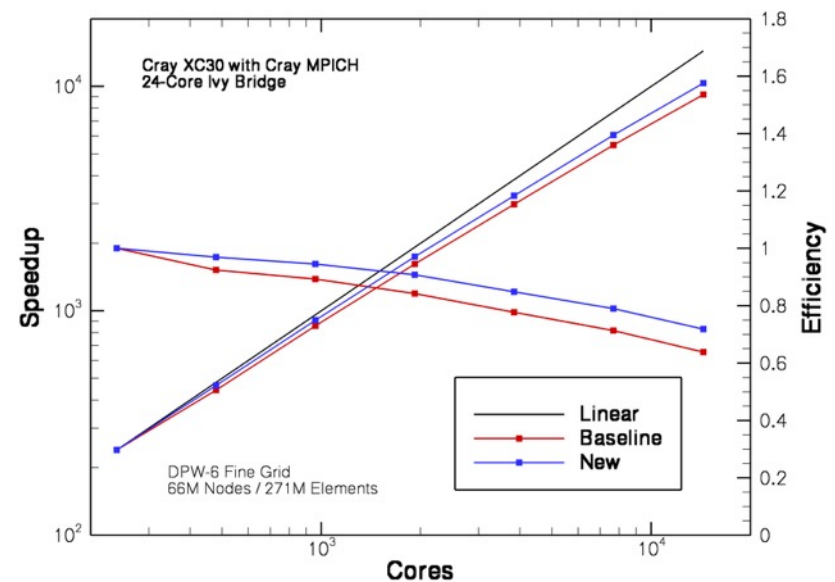
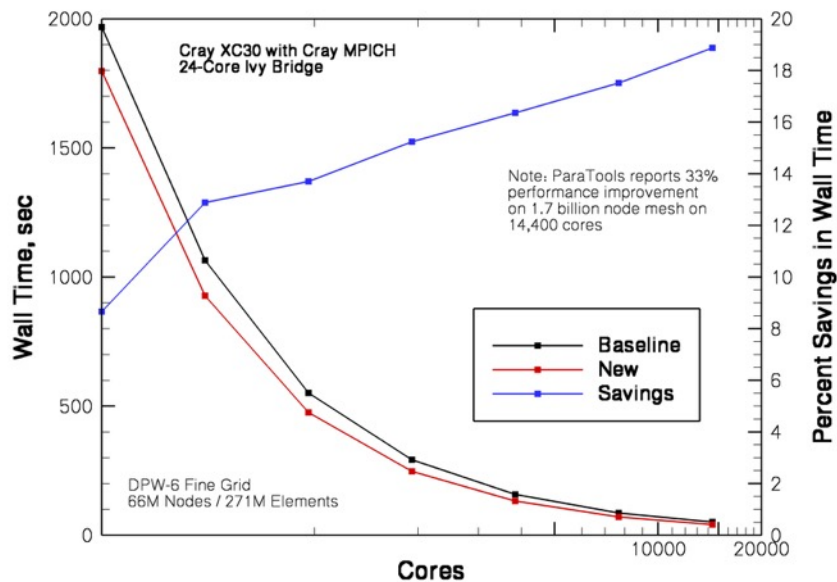
MPI_Abort()

- $O(1)$ vs. $O(N)$



FUN3D Wing-Body-Nacelle

“These days I get excited about 1-2% speedups that I find....quite unusual to find something of this magnitude these days, especially with just a 2-line fix in the code! :)”

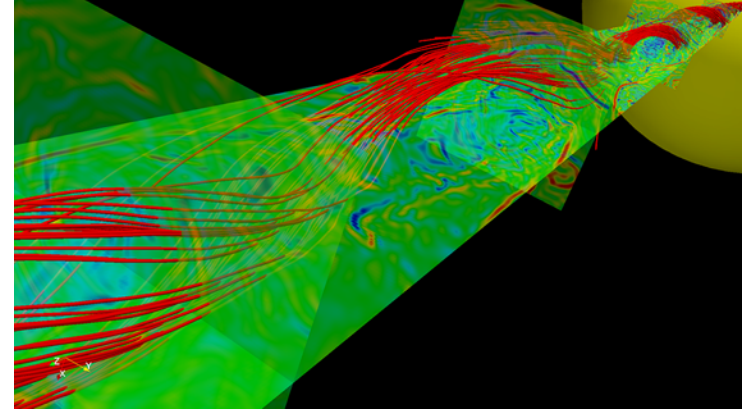


NGC Lunch and Learn

CASE STUDY: IRMHD

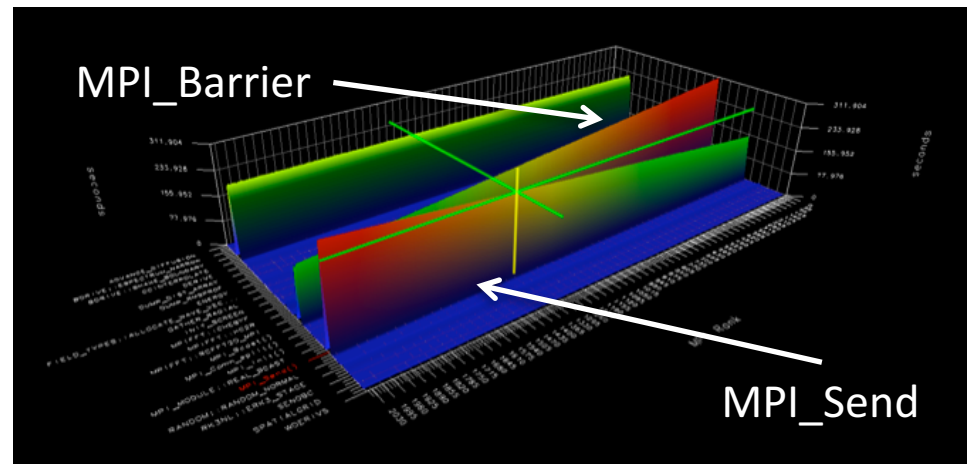
IRMHD on Intrepid and Mira

- INCITE magnetohydrodynamic simulation to understand solar winds and coronal heating
 - First direct numerical simulations of Alfvén wave (AW) turbulence in extended solar atmosphere accounting for inhomogeneities
 - Team
 - University of New Hampshire (Jean Perez and Benjamin Chandran)
 - ALCF (Tim Williams)
 - University of Oregon (Sameer Shende)
- IRMHD (Inhomogeneous Reduced Magnetohydrodynamics)
 - Fortran 90 and MPI
 - Excellent weak and strong scaling properties
 - Tested and benchmarked on Intrepid and Mira
- HPC Source article and ALCF news
<https://www.alcf.anl.gov/articles/furthering-understanding-coronal-heating-and-solar-wind-origin>



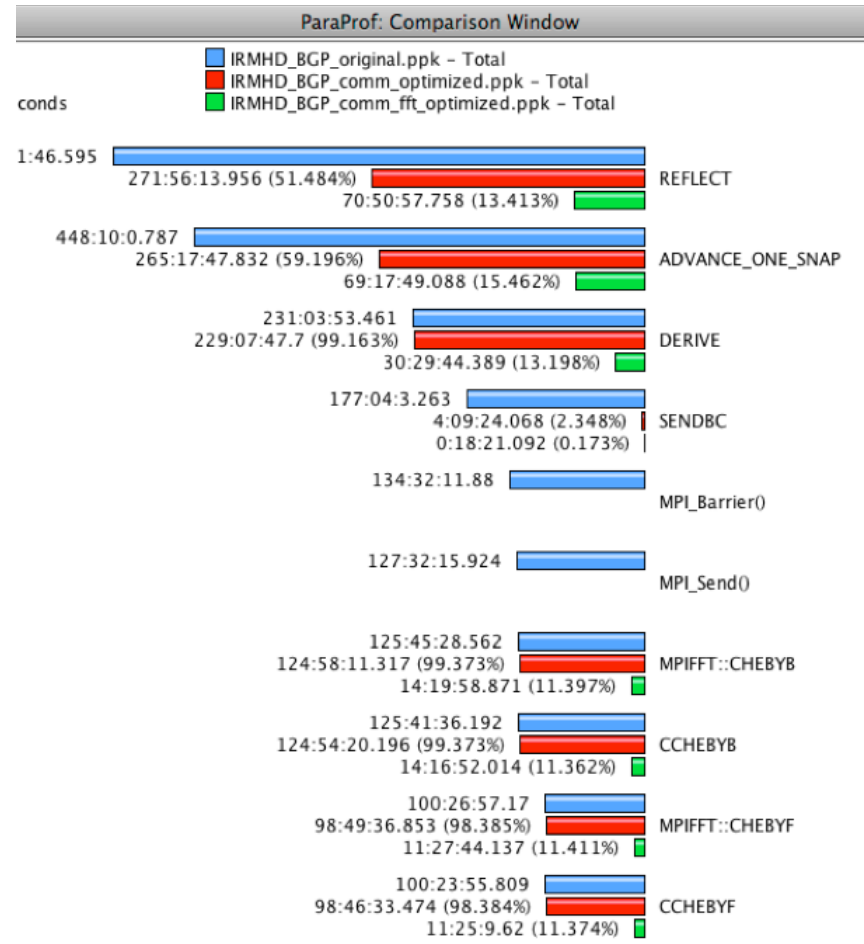
IRMHD Communication Analysis

- Source-based (direct) instrumentation
- MPI instrumentation and volume measurement
- IRMHD exhibited significant synchronous communication bottlenecks
- On 2,408 cores of BG/P:
 - **MPI_Send** and **MPI_Bcast** take significant time
 - Opportunities for communication/computation overlap
 - Identified possible targets for computation improvements



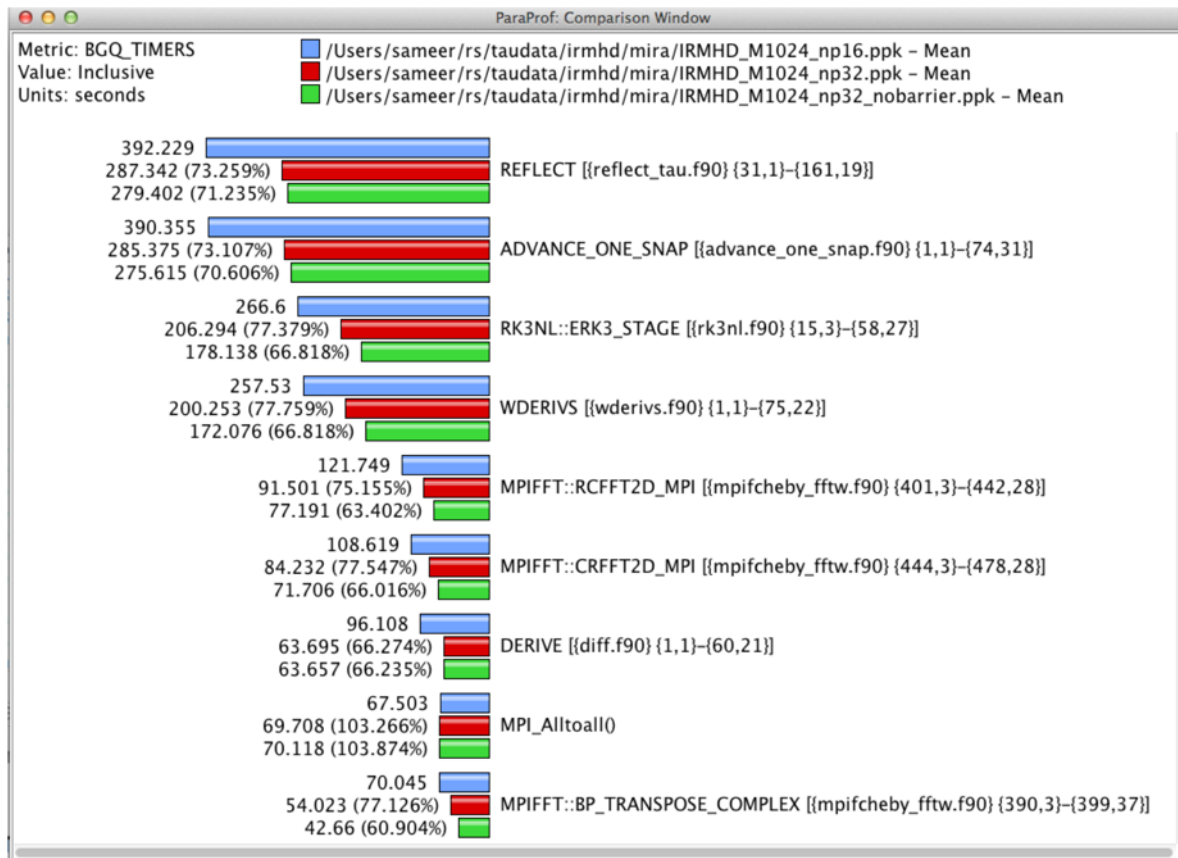
IRMHD Optimization on Intrepid (BG/P)

- On 2,408 cores, overall execution time reduced from 528.18 core hours to 70.8 core hours (**>7x improvement**)
- Non-blocking communication substrate
- More efficient implementation of underlying FFT



IRMHD Optimization on MIRA (BG/Q)

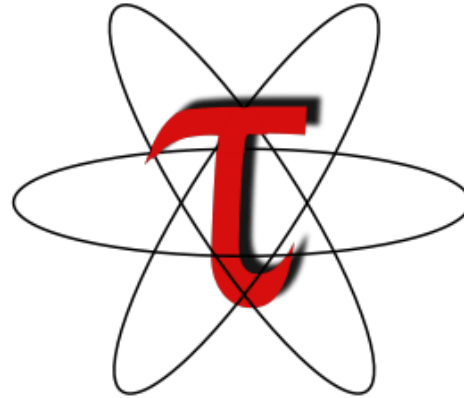
- Oversubscribe nodes: 32k ranks vs. 16k per node
- Overall time improvement: 71.23% of original



NGC Lunch and Learn

CONCLUSION

Try it out!



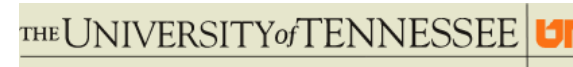
<http://www.taucommander.com>

<http://www.hpclinux.com> [LiveDVD]

Free download, open source, BSD license

Acknowledgements

- Department of Energy
 - Office of Science
 - Argonne National Laboratory
 - Oak Ridge National Laboratory
 - NNSA/ASC Trilabs (SNL, LLNL, LANL)
- HPCMP DoD PETTT Program
- National Science Foundation
 - Glassbox, SI-2
- University of Tennessee
- University of New Hampshire
 - Jean Perez, Benjamin Chandran
- University of Oregon
 - Allen D. Malony, Sameer Shende
 - Kevin Huck, Wyatt Spear
- TU Dresden
 - Holger Brunst, Andreas Knupfer
 - Wolfgang Nagel
- Research Centre Jülich
 - Bernd Mohr
 - Felix Wolf



UNIVERSITY OF OREGON



TAU Performance System

REFERENCE

Online References

- PAPI:
 - PAPI documentation is available from the PAPI website:
<http://icl.cs.utk.edu/papi/>
- TAU:
 - TAU Users Guide and papers available from the TAU website:
<http://tau.uoregon.edu/>
- VAMPIR:
 - VAMPIR website:
<http://www.vampir.eu/>
- Scalasca:
 - Scalasca documentation page:
<http://www.scalasca.org/>
- Eclipse PTP:
 - Documentation available from the Eclipse PTP website:
<http://www.eclipse.org/ptp/>

Compiling Fortran Codes with TAU

- **If your Fortran code uses free format in .f files (fixed is default for .f):**
% export TAU_OPTIONS='-optPdtF95Opts="-R free" -optVerbose'
- **To use the compiler based instrumentation instead of PDT (source-based):**
% export TAU_OPTIONS='-optComplnst -optVerbose'
- **If your Fortran code uses C preprocessor directives (#include, #ifdef, #endif):**
% export TAU_OPTIONS='-optPreProcess -optVerbose'
- **To use an instrumentation specification file:**
% export TAU_OPTIONS=
 '-optTauSelectFile=select.tau -optVerbose -optPreProcess'

Example select.tau file

```
BEGIN_INSTRUMENT_SECTION  
loops file="*" routine="#"  
memory file="foo.f90" routine="#"  
io file="abc.f90" routine="FOO"  
END_INSTRUMENT_SECTION
```


Generate a PAPI profile with 2 or more counters

```
% export TAU_MAKEFILE=$TAU/Makefile.tau-papi-mpi-pdt
% export TAU_OPTIONS='-optTauSelectFile=select.tau -optVerbose'
% cat select.tau
BEGIN_INSTRUMENT_SECTION
loops routine="#"
END_INSTRUMENT_SECTION

% export PATH=$TAU_ROOT/bin:$PATH
% make F90=tau_f90.sh
(Or edit Makefile and change F90=tau_f90.sh)
%
% export TAU_METRICS=TIME:PAPI_FP_INS:PAPI_L1_DCM
% mpirun -np 4 ./a.out
% paraprof --pack app.ppk
Move the app.ppk file to your desktop.
% paraprof app.ppk
Choose Options -> Show Derived Metrics Panel -> "PAPI_FP_INS",
click "/", "TIME", click "Apply" and choose the derived metric.
```

Tracking I/O

```
% export TAU_MAKEFILE=$TAU/Makefile.tau-papi-mpi-pdt
% export PATH=$TAU_ROOT/bin:$PATH
% export TAU_OPTIONS='-optTrackIO -optVerbose'
% make CC=tau_cc.sh CXX=tau_cxx.sh F90=tau_f90.sh
% mpirun -n 4 ./a.out
% paraprof -pack ioprofile.ppk
% export TAU_TRACK_IO_PARAMS=1
% mpirun -n 4 ./a.out
```

Installing and Configuring TAU

•Installing PDT:

- `wget http://tau.uoregon.edu/pdt.tgz`
- `./configure --prefix=<dir>; make ; make install`

•Installing TAU:

- `wget http://tau.uoregon.edu/tau.tgz`
- `./configure -bfd=download -pdt=<dir> -papi=<dir> ...`
- `make install`

•Using TAU:

- `export TAU_MAKEFILE=<taudir>/<arch>/lib/Makefile.tau-<TAGS>`
- `make CC=tau_cc.sh CXX=tau_cxx.sh F90=tau_f90.sh`

Compile-Time Options (TAU_OPTIONS)

% tau_compiler.sh

-optVerbose	Turn on verbose debugging messages
-optComplnst	Use compiler based instrumentation
-optNoComplnst	Do not revert to compiler instrumentation if source instrumentation fails.
-optTrackIO	Wrap POSIX I/O call and calculates vol/bw of I/O operations
-optMemDbg	Runtime bounds checking (see TAU_MEMDBG_* env vars)
-optKeepFiles	Does not remove intermediate .pdb and .inst.* files
-optPreProcess	Preprocess sources (OpenMP, Fortran) before instrumentation
-optTauSelectFile=" <i><file></i> "	Specify selective instrumentation file for <i>tau_instrumentor</i>
-optTauWrapFile=" <i><file></i> "	Specify path to <i>link_options.tau</i> generated by <i>tau_gen_wrapper</i>
-optHeaderInst	Enable Instrumentation of headers
-optTrackUPCR	Track UPC runtime layer routines (used with tau_upc.sh)
-optPdtF95Opts=""	Add options for Fortran parser in PDT (f95parse/gfparse) ...

Runtime Environment Variables

Environment Variable	Default	Description
TAU_TRACE	0	Setting to 1 turns on tracing
TAU_CALLPATH	0	Setting to 1 turns on callpath profiling
TAU_TRACK_MEMORY_LEAKS	0	Setting to 1 turns on leak detection (for use with <code>-optMemDbg</code> or <code>tau_exec</code>)
TAU_MEMDBG_PROTECT_ABOVE	0	Setting to 1 turns on bounds checking for dynamically allocated arrays. (Use with <code>-optMemDbg</code> or <code>tau_exec -memory_debug</code>).
TAU_CALLPATH_DEPTH	2	Specifies depth of callpath. Setting to 0 generates no callpath or routine information, setting to 1 generates flat profile and context events have just parent information (e.g., Heap Entry: foo)
TAU_TRACK_IO_PARAMS	0	Setting to 1 with <code>-optTrackIO</code> or <code>tau_exec -io</code> captures arguments of I/O calls
TAU_TRACK_SIGNALS	0	Setting to 1 generate debugging callstack info when a program crashes
TAU_COMM_MATRIX	0	Setting to 1 generates communication matrix display using context events
TAU_THROTTLE	1	Setting to 0 turns off throttling. Enabled by default to remove instrumentation in lightweight routines that are called frequently
TAU_THROTTLE_NUMCALLS	100000	Specifies the number of calls before testing for throttling
TAU_THROTTLE_PERCALL	10	Specifies value in microseconds. Throttle a routine if it is called over 100000 times and takes less than 10 usec of inclusive time per call
TAU_COMPENSATE	0	Setting to 1 enables runtime compensation of instrumentation overhead
TAU_PROFILE_FORMAT	Profile	Setting to "merged" generates a single file. "snapshot" generates xml format
TAU_METRICS	TIME	Setting to a comma separated list generates other metrics. (e.g., TIME:P_VIRTUAL_TIME:PAPI_FP_INS:PAPI_NATIVE_<event>\\:<subevent>)