An Approach to Creating Performance Visualizations in a Parallel Profile Analysis Tool

Wyatt Spear, Allen D. Malony, Chee Wai Lee,
Scott Biersdorff, and Sameer Shende

Department of Computer and Information Science
University of Oregon

Motivation

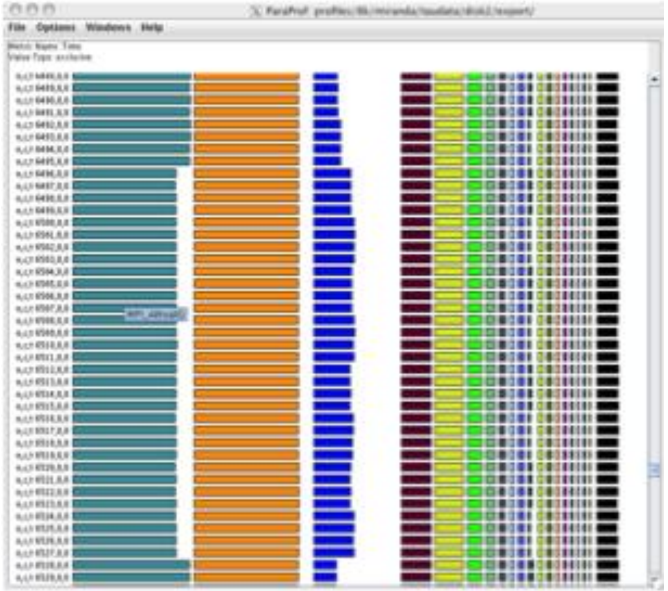
- Large-scale parallel application performance analysis
  - Requires scalable performance measurement
  - Ability to understand multi-dimensional performance data
  - Automated analysis and diagnosis is still a challenge
  - Performance interpretation continues to involve the user
- How to build better tool capabilities to support this goal?
  - Need to be able to handle data dimensionality
  - Presentation of performance information is important
- Performance visualization is an opportunity
  - Convey visual characteristics and traits in the data
  - Aids in data exploration and pattern analysis
  - 3D visualization helps in identifying relations and scale

Visualization Challenges

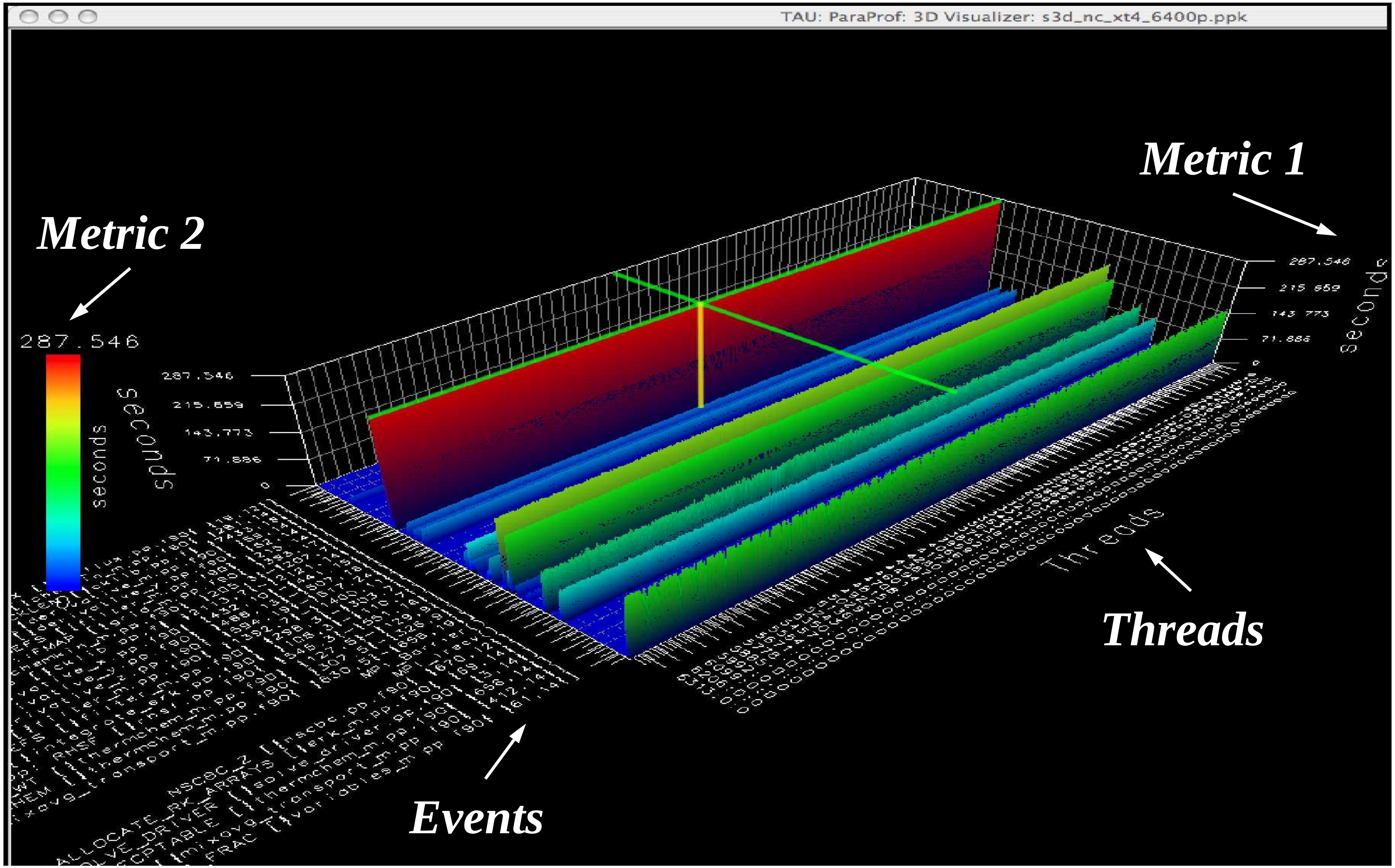
- Good visualization is a design process
  - Integrates properties of the performance data (as understood by the user) with the graphical aspects for good visual form Performance data lacks a natural semantic visual basis
  - Utilize a variety of graphical forms and visualization types (e.g., statistical, informational, physical, abstract)
- Must deal with scale dimensions: # cores, events, metrics
  - 3D graphics helps, but design challenge remains
- Practical concerns (with creating new visualizations)
  - A few "canned" views limit visualization types
  - Visualization environments would need adaptation
  - Can not expect users to program with visualization libraries

A Parallel Profile Analysis Tool

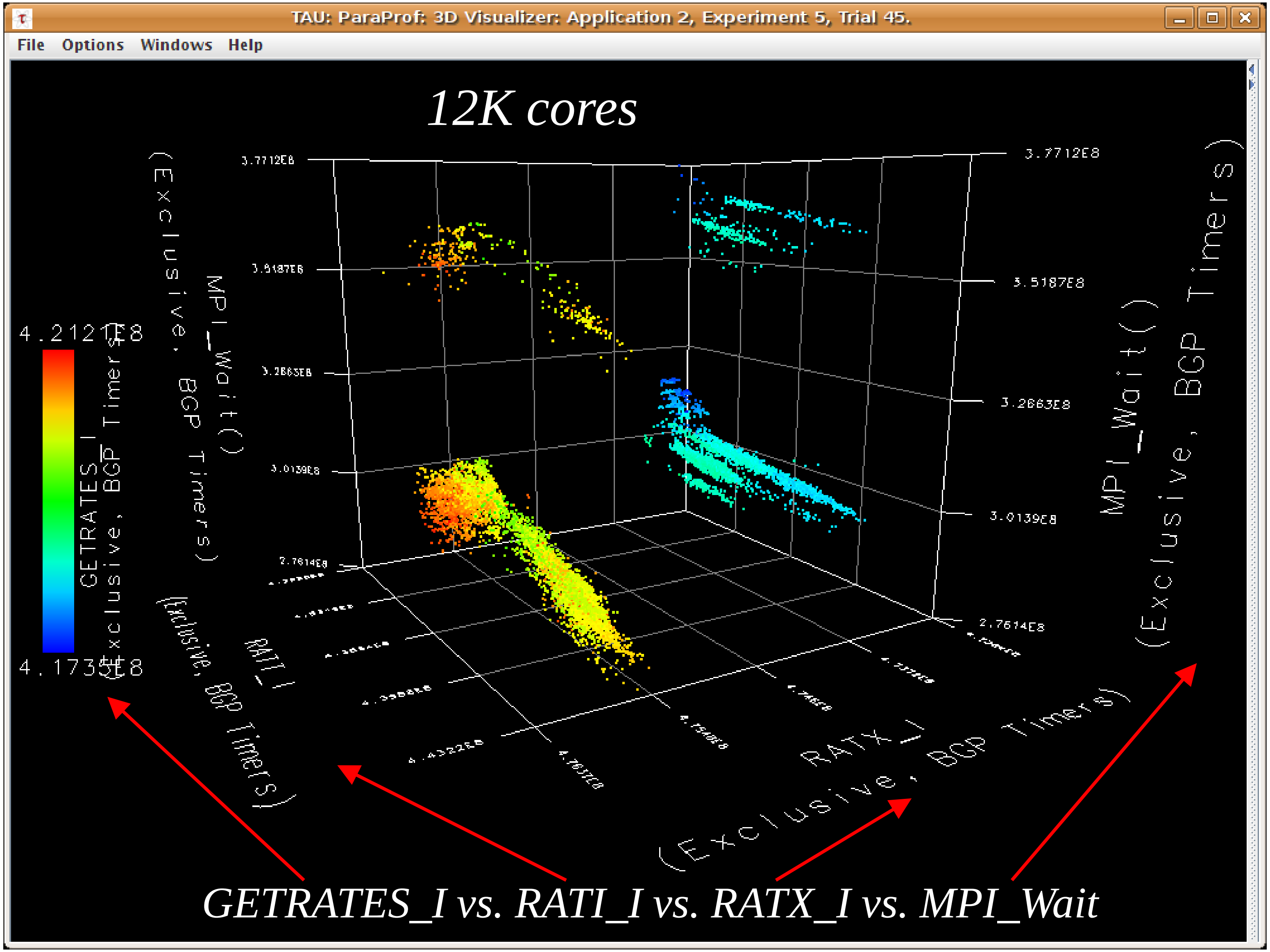
- TAU represents a scalable profile measurement tool
  - Visualize profile of execution for 100,000+ threads
  - 100+ events and multiple performance metrics per thread
- TAU ParaProf tool processes parallel profiles
- ParaProf provides "canned" views
  - 2D: bargraph, histogram
  - 3D: full profile, correlation
- Example: S3D flow solver for simulation of turbulent combustion
  - 12K cores execution
  - IBM BG/P (Intrepid) and Cray XT4 (Jaguar)



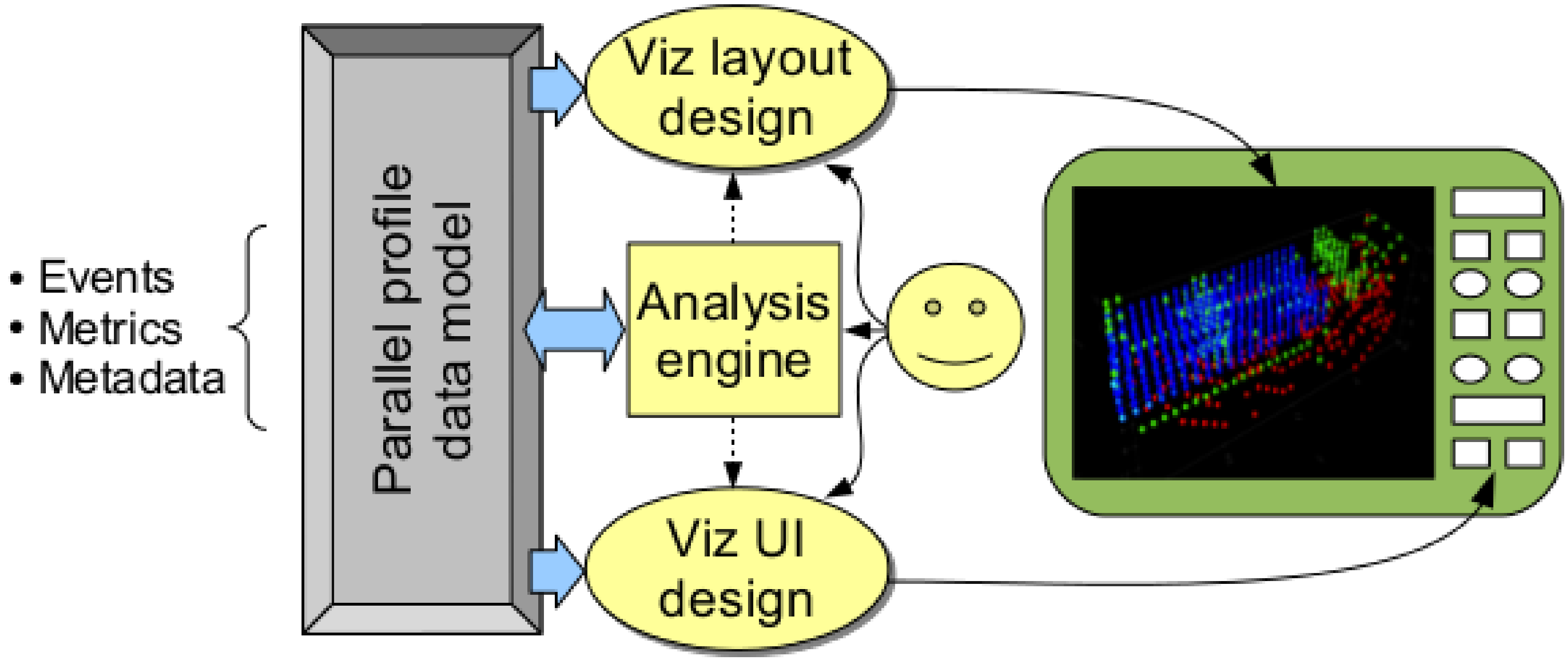
S3D Full Profile View (Jaguar, 12K cores)



S3D 3D Correlation Cube (Intrepid, 12K cores)



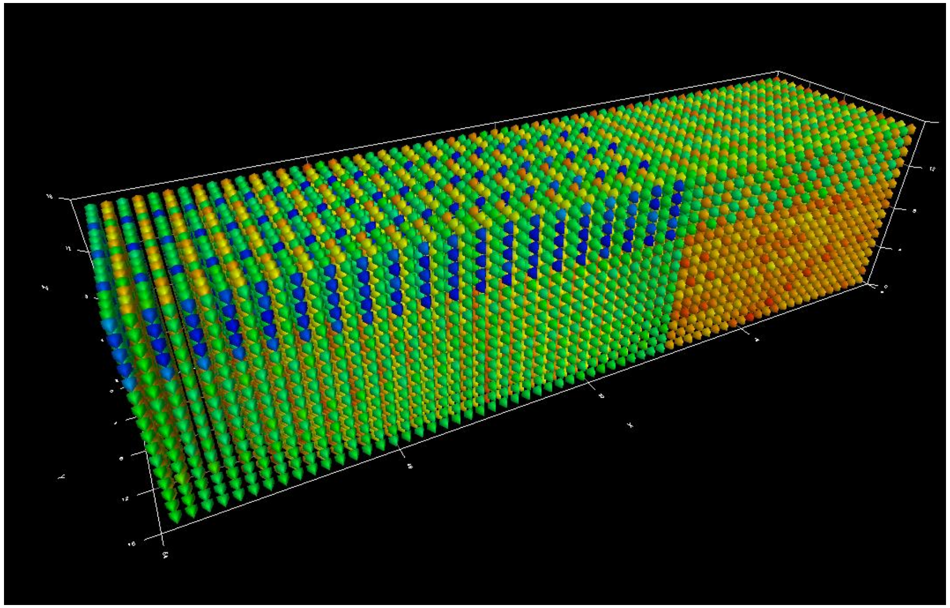
Extending Visualization Support in Profile Tool



- ❐ User defines visualization
  - ❍ Based on performance data model
- ❐ Specifies layout based on events, metrics, and metadata
- ❐ UI provides control of data binding and visualization

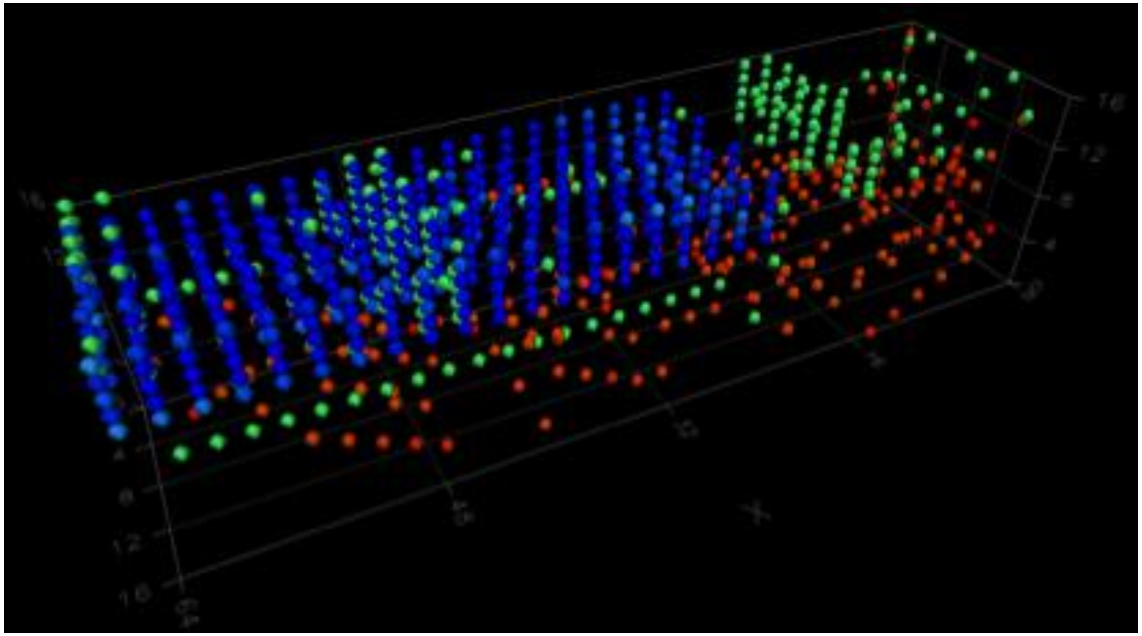
Using Process Topology Metadata

- Inspired by the Scalasca CUBE topology display
- Each point represents a thread of execution (MPI process)
  - Positioned according to the Cartesian (x,y,z,t) coordinates
- Color is determined by selected event/metric value
- Topology information can be recorded in TAU metadata
- ParaProf reads metadata to determine topology and create layout
- Example: Sweep3D 3D neutron transport application
  - 16K run on BG/P
  - Color is exclusive time in the “sweep” function



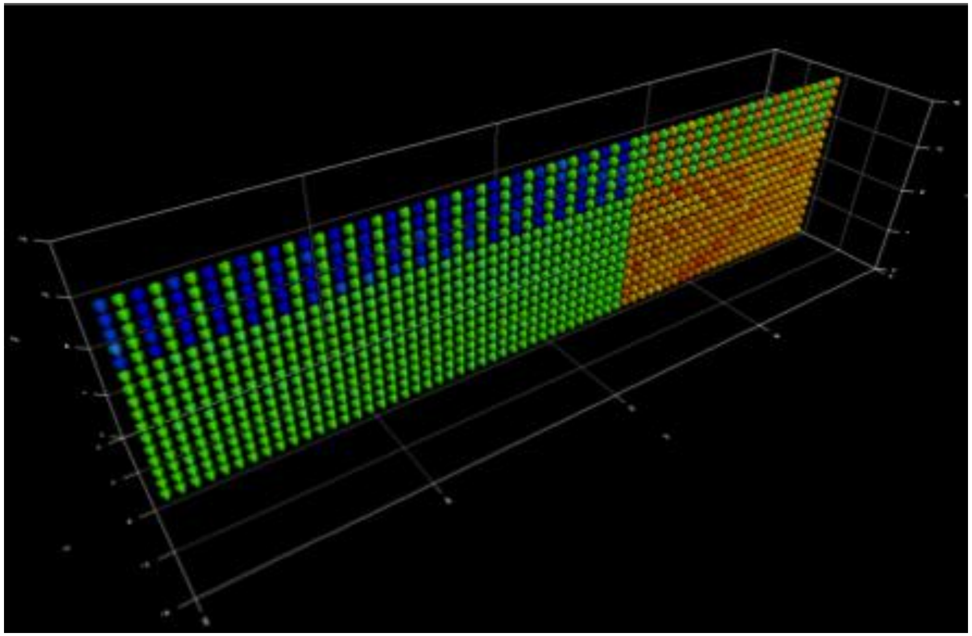
Viewing Internal Structure

- Dense topologies can hide internal structure in the visualization
- Restrict visibility by color value to expose performance patterns



- ParaProf visualization UI now allows for range filtering
  - Mid-level values can be excluded
  - Include high outliers (hotspots)
  - Include low outliers (under-utilized ranks)

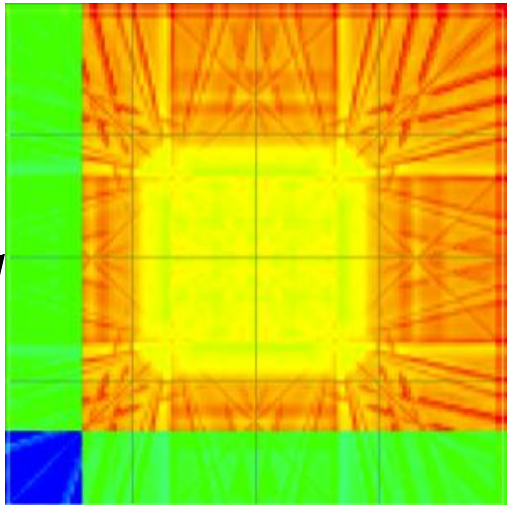
Slicing to Reduce Dimensionality



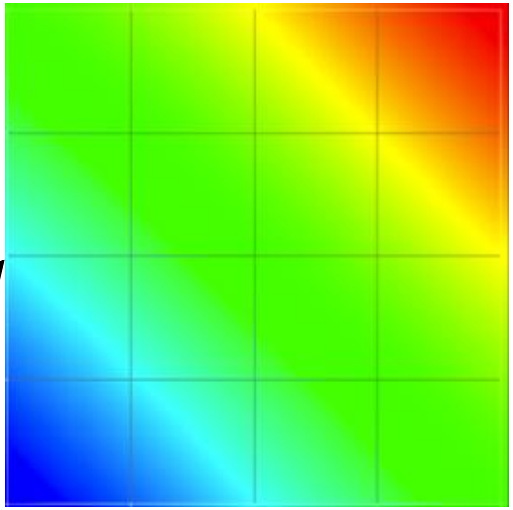
- Restrict visibility to sub-dimensions
  - Slices along spatial axes
- ParaProf visualization UI provides dimensionality reduction control
  - Multiple axis controls allow selection of planes, lines, or an individual point
- Several alternatives for value reduction
  - Show only value of points
  - Averaging the color value for all points in the selected area

Custom Topologies

- Certain views may hide deeper inter-process behavior
- Spatially dependent performance issues may be revealed by manipulating topology
- Sweep3D profile with alternative Cartesian mapping exposes distribution of computational effort

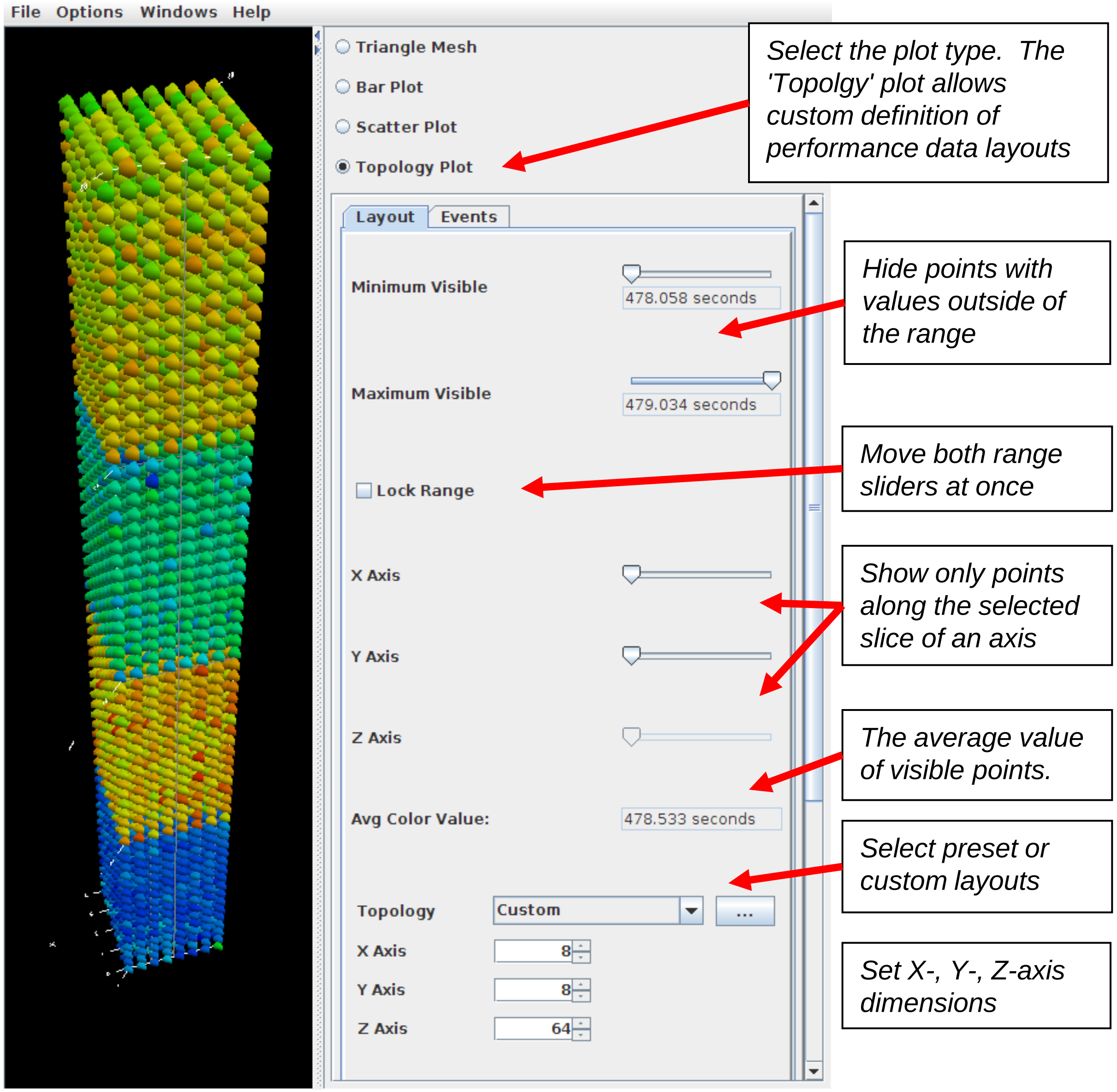


- Topology has direct effect on communication
- Visualization mapped to hardware topologies can suggest better node/rank mapping
- *MPI_AllReduce()* values for Sweep3D highlights waiting distribution from rank 0 (lower left) to the most distant rank (upper right)

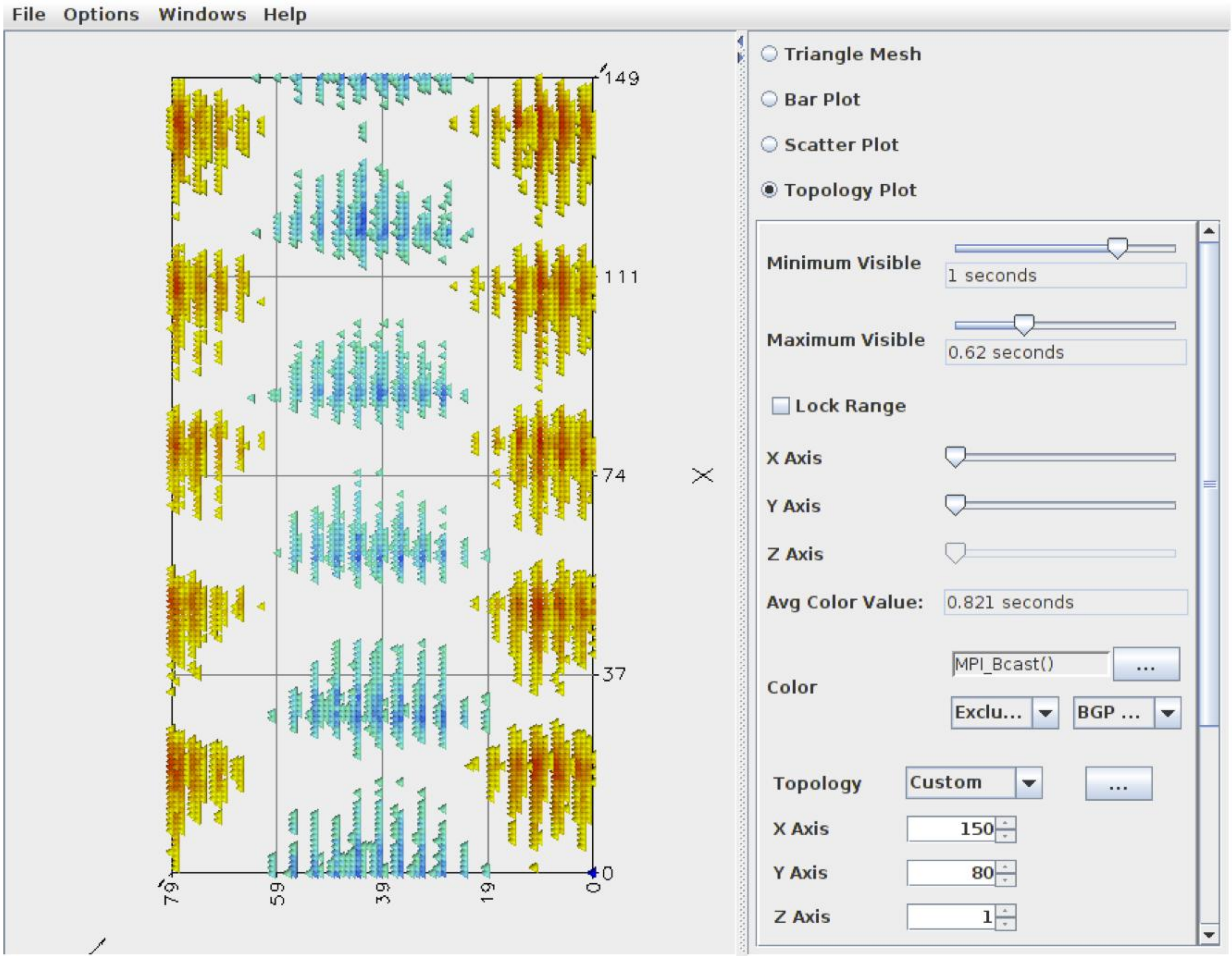


Topology Control UI

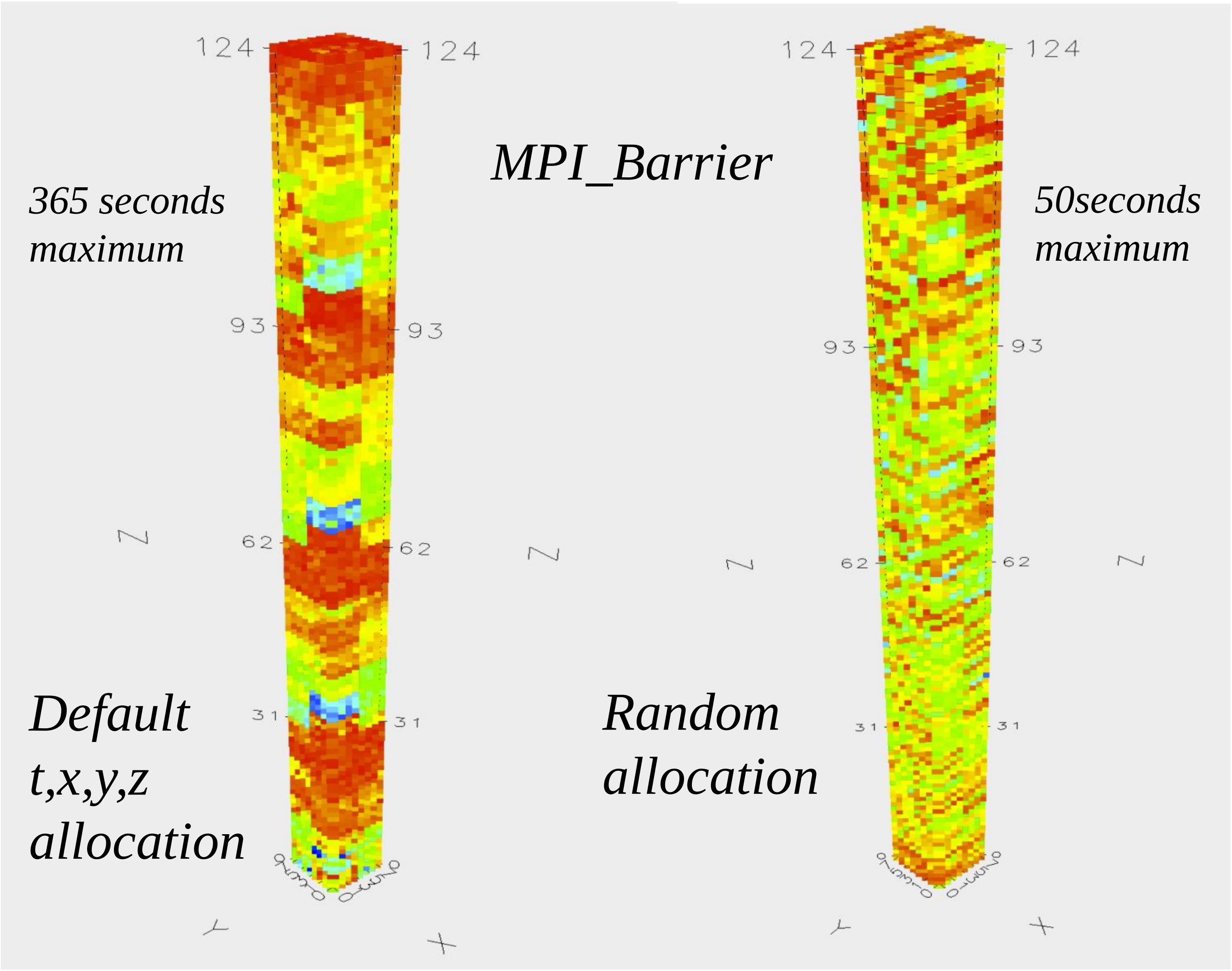
- *Layout* tab allows customization of the position and visibility of data points
- Performance event/metric data used to define color and position is selected in the *Event* tab
- Additional rendering options, such as color scale and point size are available
- 4k-core S3D run on IBM BG/P



S3D Topology View with Z Axis Collapsed



S3D on Intrepid with Different Allocations

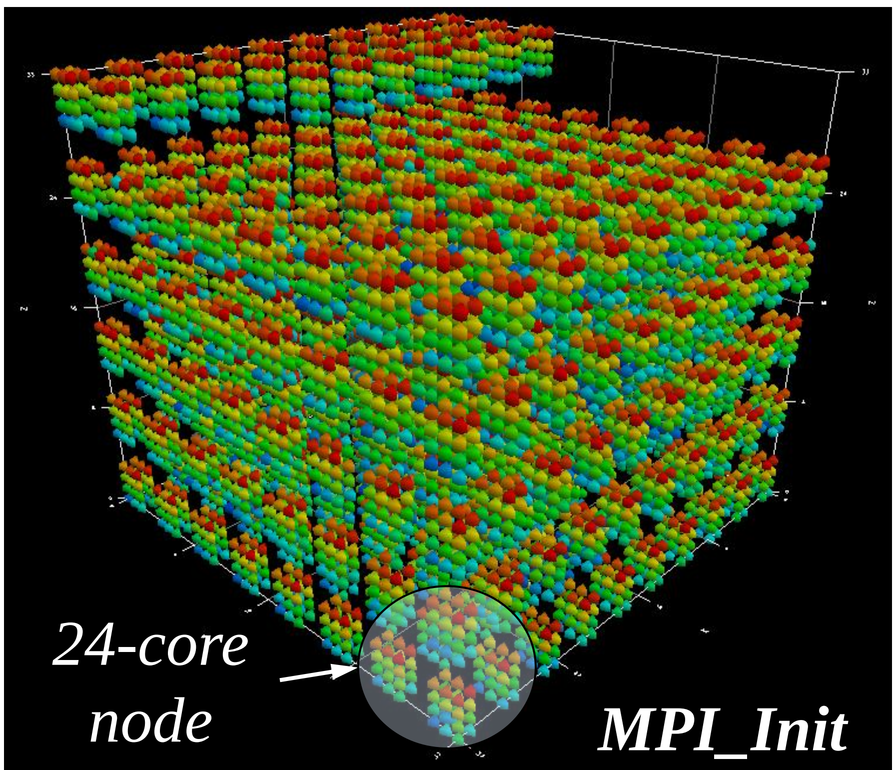


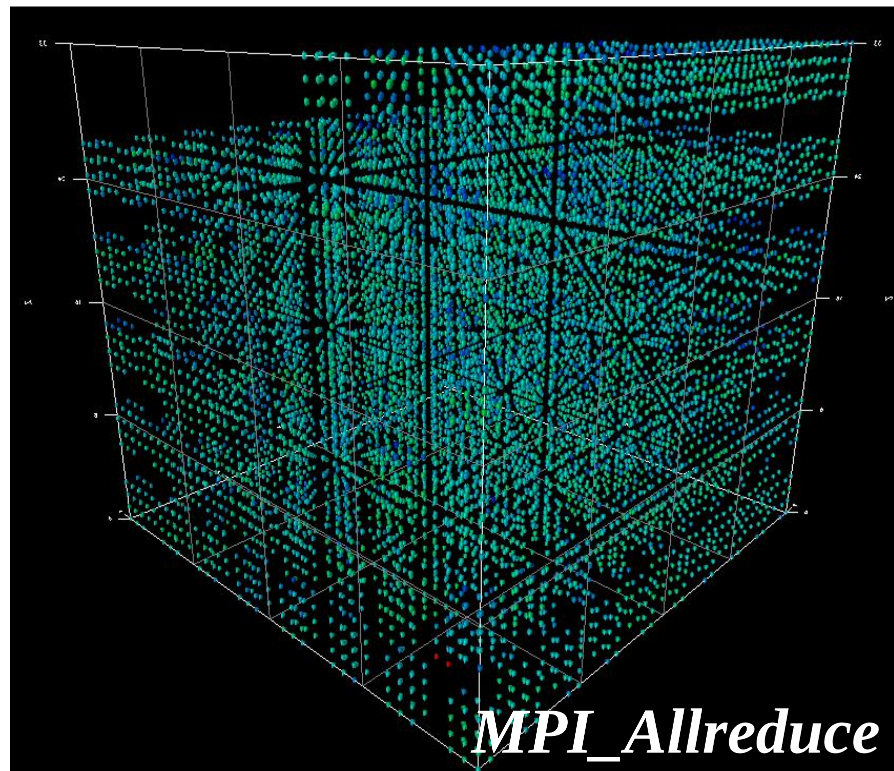
Cray XE6 Topology

- Just added ability to capture Cray XE6 topology metadata
- Cray RCA library on Cray maps MPI ranks to topology
  - NERSC uses this API to get the x/y/z node coordinates
  - However it does not provide the coordinates of cores
- A unique node id is provided by */proc/cray_xt/cname*
  - Contains node information (rack, chassis, …)
  - Position of chips on a node is provided if environment variable *MPICH_CPUMASK_DISPLAY* is set
- TAU collects all node and core location data this way
- Output is parsed by ParaProf
- Also can use technique from Brian Wylie from Juelich

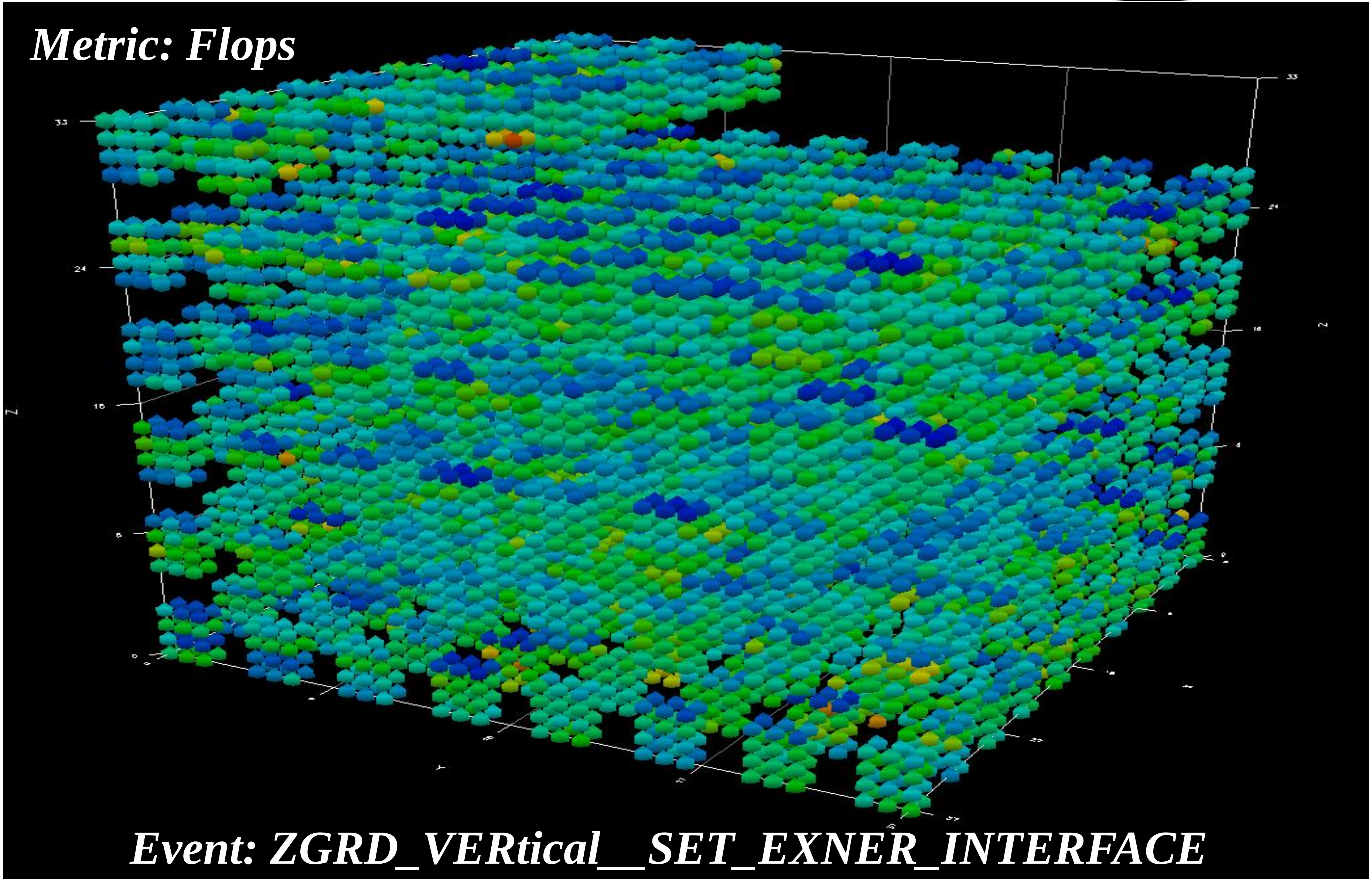
Cray XE6 Topology Visualization

- Example: GCRM global cloud resolving mode
- Visualization of 10K execution on Cray XE6
  - Topological view shows core distribution (24 cores per node)
  - Custom layout makes decision on node/core blocking

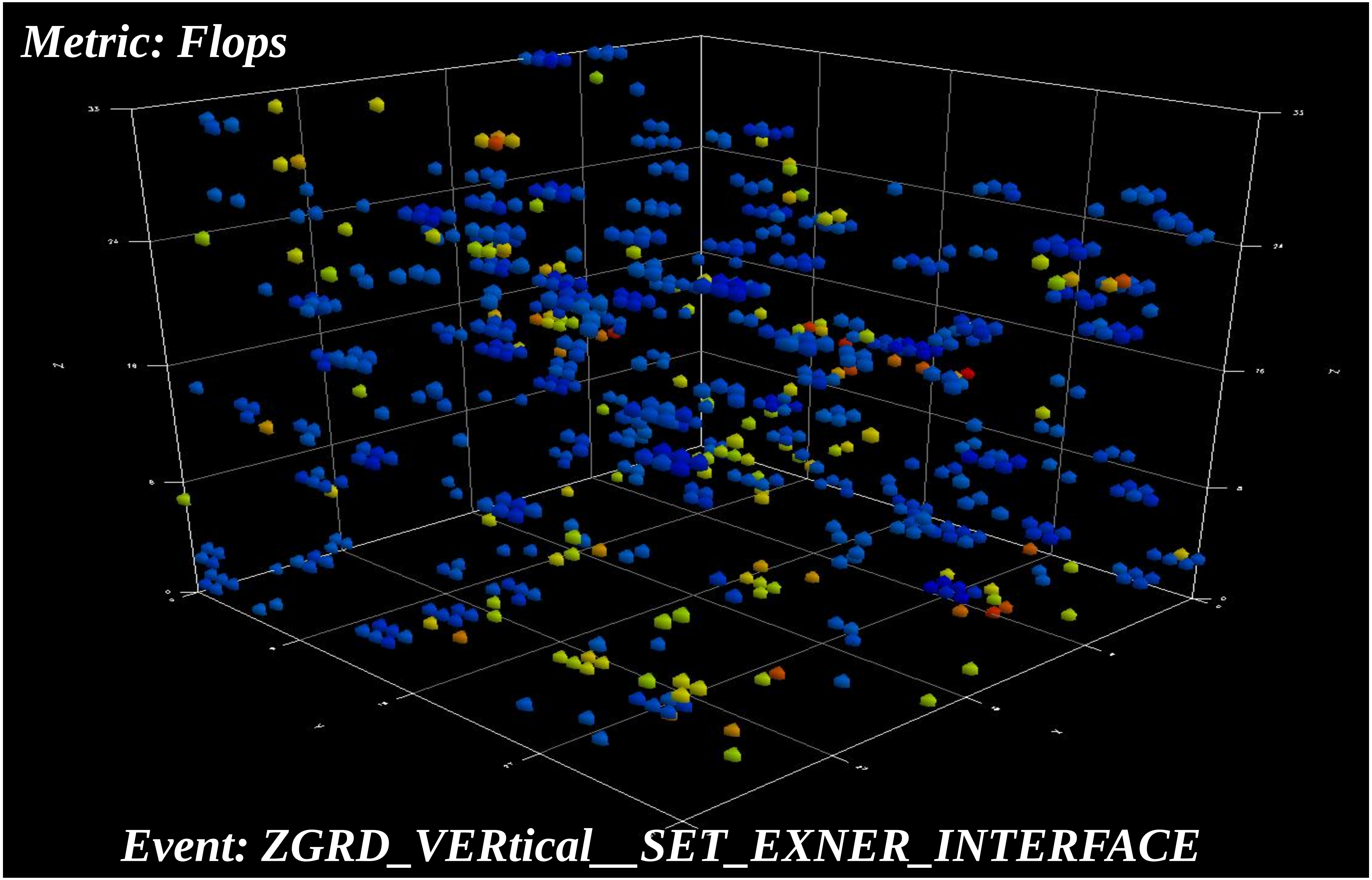




GCRM on Cray XE6 (10K Cores)



GCRM on Cray XE6 (10K Cores, Filtered)



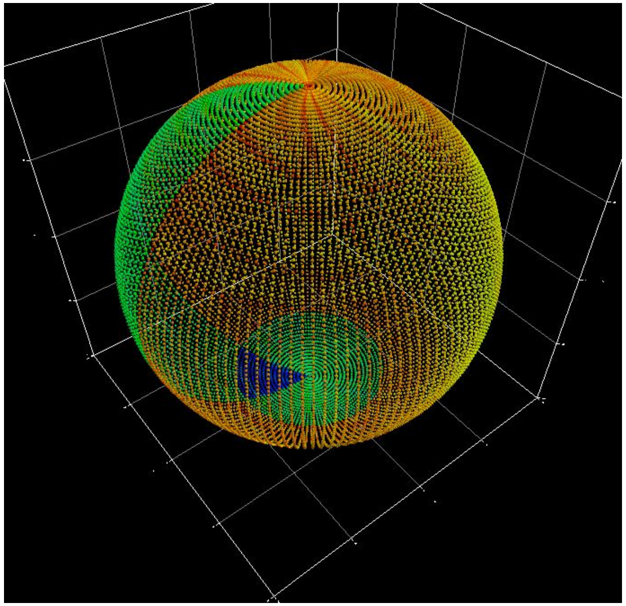
Visual Layout Specification

- Want to allow creation of explicit layouts by the user
- IDEA: Define a specification "language" that allows mathematical expressions to describe features of performance display
  - Equations define X, Y, Z coordinates and color per process
  - Event and metrics are seen as variables
    - *eventX.val* : value for *X*th specified event and metric
    - *eventX.{min,max,mean}* : global aggregate values
    - *atomicY* : *Y*th atomic event value
  - Intermediate variables can be used in the calculation
  - Defined global variables (e.g., *max rank*) are provided
- Specifications are loaded and processed by ParaProf
  - Use the MESP expression parser

Sphere Layout Specification

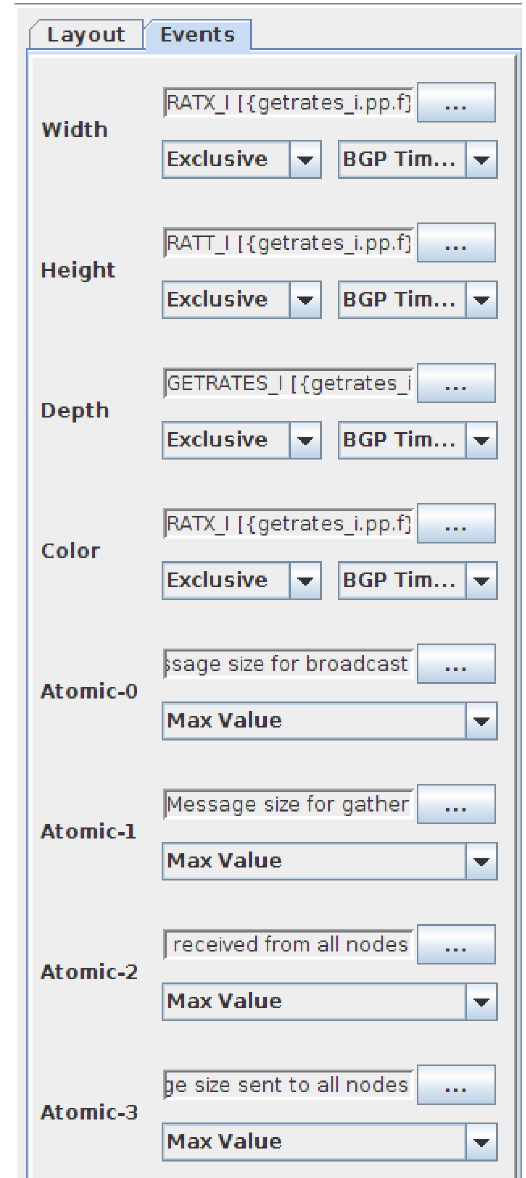
- Spatially mediated performance behavior may not be represented directly in topology metadata
  - Applications allocate resources with respect to a data-driven model
- Position of each point can be defined by custom equations in terms of event/metric, aggregate, atomic event and metadata
- Sweep3D profile mapped to a sphere

```
BEGIN_VIZ=Sphere
rootRanks=sqrt(maxRank)
theta=2*pi()/rootRanks*mod(rank,rootRanks)
phi=pi()/rootRanks*(ceil(rank/rootRanks))
x=cos(theta)*sin(phi)*100
y=sin(theta)*sin(phi)*100
z=cos(phi)*100
END_VIZ
```



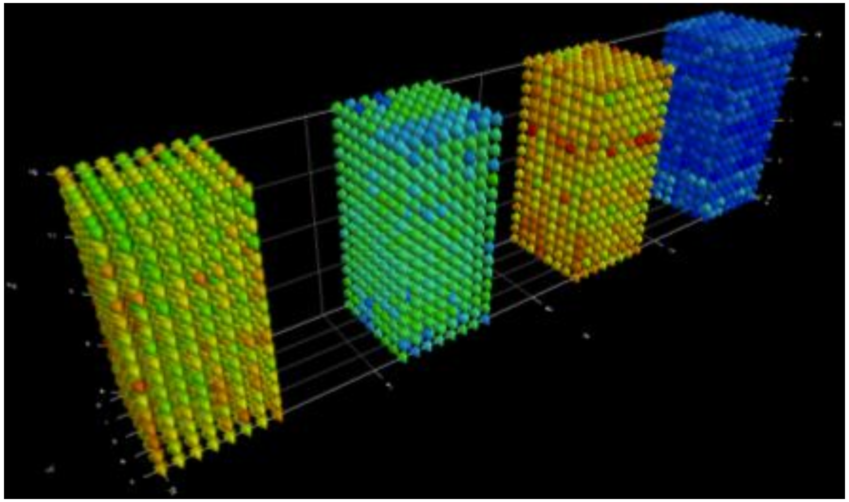
ParaProf Events Panel

- Events / metrics get bound in ParaProf UI
- Example:
  - *event0* is the FLOP count for function *foo*
  - *event1* is the time value for function *foo*
  - To set the X coordinate for each process point to the FLOPS for event *foo*:
    x = event0.val / event1.val
  - To set the Y coordinate for each process point to the global average FLOPS for event *foo*:
    y = event0.mean / event1.mean



Adding Dimensionality

- Topologies can involve more than three dimensions (e.g., intranode)
- Mirror actual machine layout to capture communication structure and cores
- Custom layouts allow specification of multiple points from a single process/rank
- 4K-core S3D run on BG/P
- Default topology only covers X, Y, Z coordinates
- A custom topology divides each *n*th core into its own block

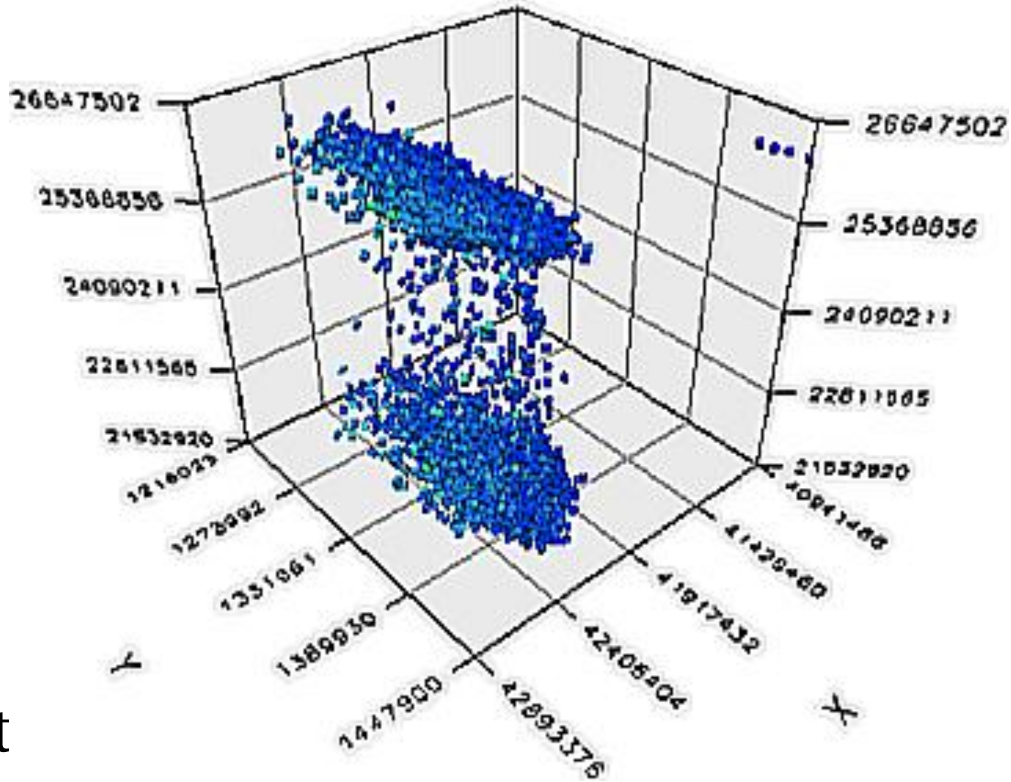


BEGIN_VIZ=4K_8x8x16Block
xdim=8
ydim=8
zdim=16

*x=mod(rank,xdim)+16*floor(rank/1024)*
y=mod(floor(rank/xdim),ydim)
z=mod(floor(rank/xdim/ydim),zdim)
END_VIZ

Non-Spatial Relationships

- Positioning of points needs not be with respect to physical or data topology
- Correlation of metrics within the same events or events between processes can indicate relevant performance effects
- Partitioning or clustering of different processes based on selected performance crit
- 3D scatterplot for 10240 core run of GCRM/ZGrd application
- Correlates four selected events, one for each spatial axis plus color



BEGIN_VIZ=ScatterTest
restrictDim=1
x=event0.val
y=event1.val
z=event2.val
END_VIZ

Conclusion and Next Steps

- Have extended TAU's ParaProf tool for visualization
  - Working with topology information
  - Provide custom topology views through UI
  - User-defined layouts
- Demonstrate on relatively large applications
- Future work
  - Expand UI for more general access to performance model
  - Allow independent manipulation of unconnected segments
  - Improve presentation of data values, ranks, and metrics
  - Better functionality for automatic higher-dimensional layouts
  - Build library of user-defined layouts for reuse

Support Acknowledgements

- Department of Energy (DOE)
  - Office of Science
  - ASC/NNSA
- Department of Defense (DoD)
  - HPC Modernization Office (HPCMO)
- NSF Software Development for Cyberinfrastructure (SDCI)
- Research Centre Juelich
- Argonne National Laboratory
- Technical University Dresden
- ParaTools, Inc.
- NVIDIA