Linking Performance Data into Scientific Visualization Tools

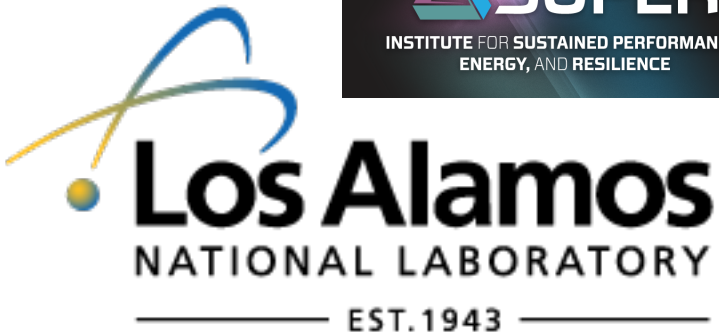
Kevin A. Huck, Kristin Potter, Doug W. Jacobsen, Hank Childs, Allen D. Malony

Workshop on Visual Performance Analysis

@ SC'14 New Orleans, Louisiana USA

November 21, 2014

UNIVERSITY OF OREGON

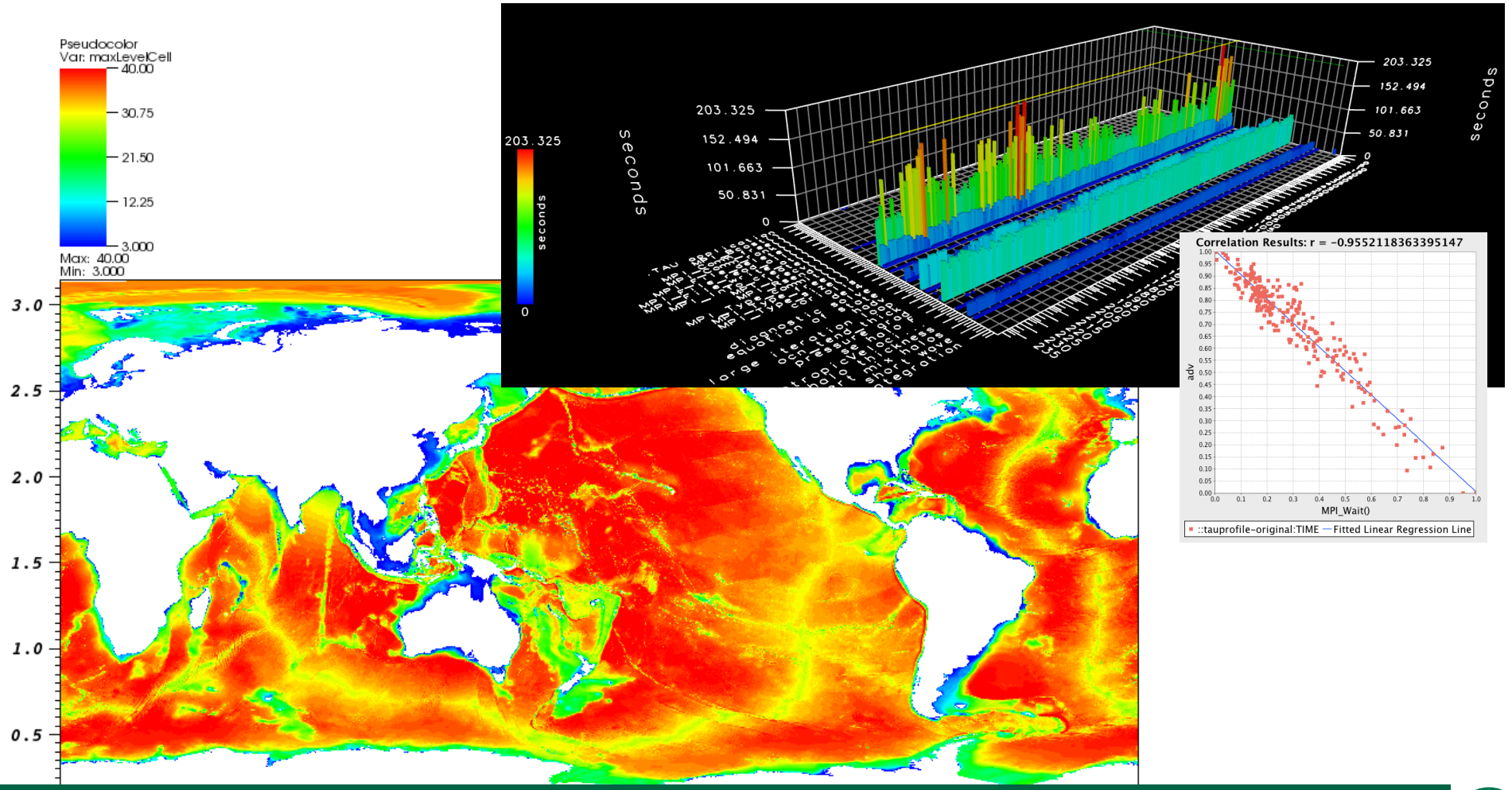


Motivation

- Evaluate the performance properties of different decomposition strategies for scientific simulation, in the physical domain
- Performance data is only analyzed, visualized with respect to node/process rank/id
- Currently, we can correlate physical properties with performance measurements, but we want to visualize those performance correlations
- Use case: load imbalance in ocean simulation

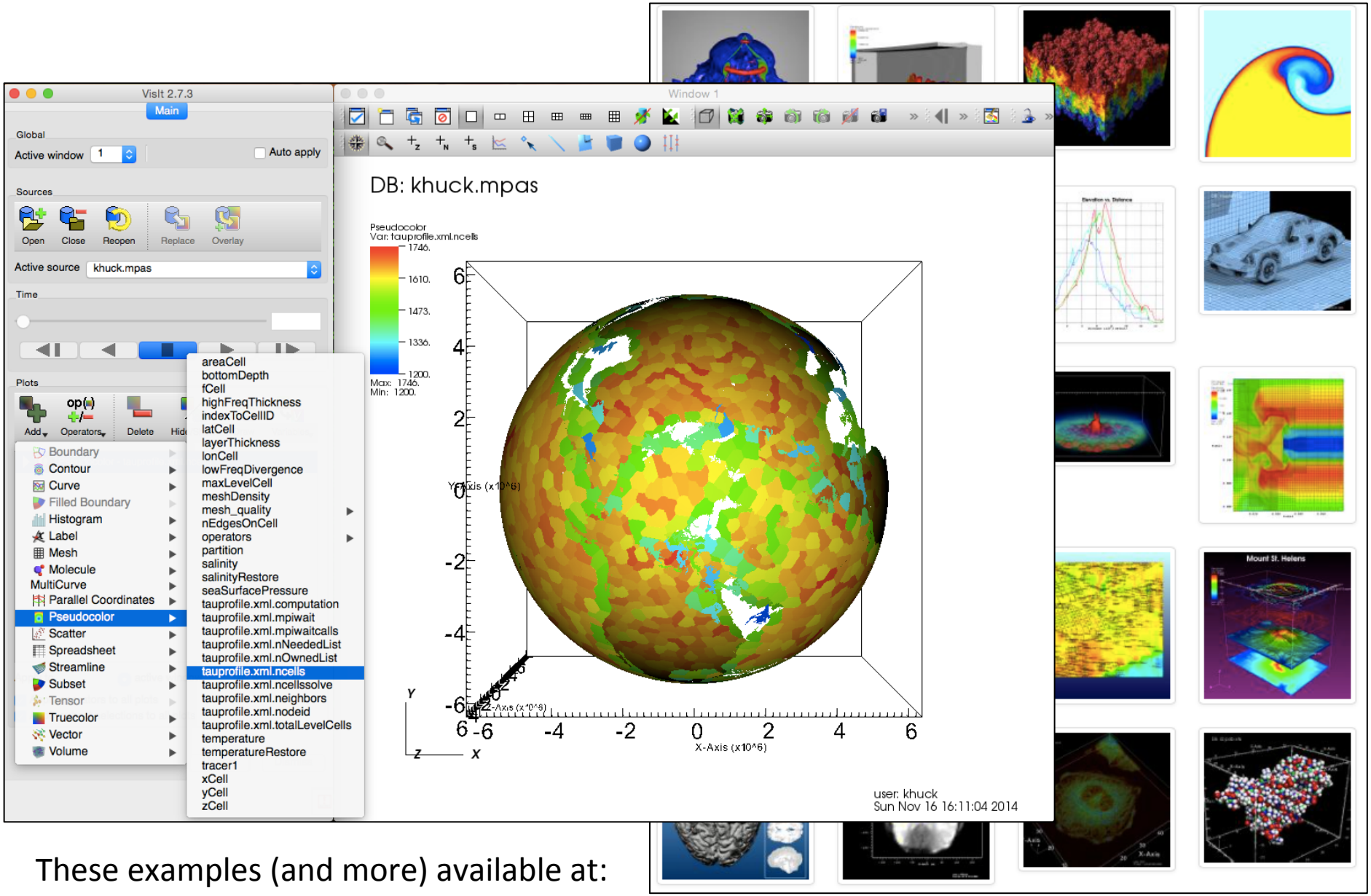
Visualization Perspectives

- Two separate(?) worlds



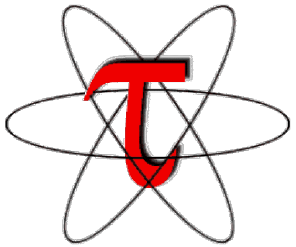
Background : VisIt

- Open-source visualization, animation, and analysis toolkit
- Interactive user interface
- Support for over 120 different scientific data formats
- The customizable plugin design allows for user-developed modifications to accommodate a large variety of scientific visualization data sets and display methods
- Many traditional visualization techniques supported
  - 2D data (pseudo-color, scatterplots, etc)
  - 3D data (volume rendering, isocontours, etc)
  - vector data (stream- lines, glyphs, etc)
- Runs on a variety of platforms
- Can be configured to post-process in parallel
- Scales from desktop machines to large-scale HPC
- http://visit.llnl.gov

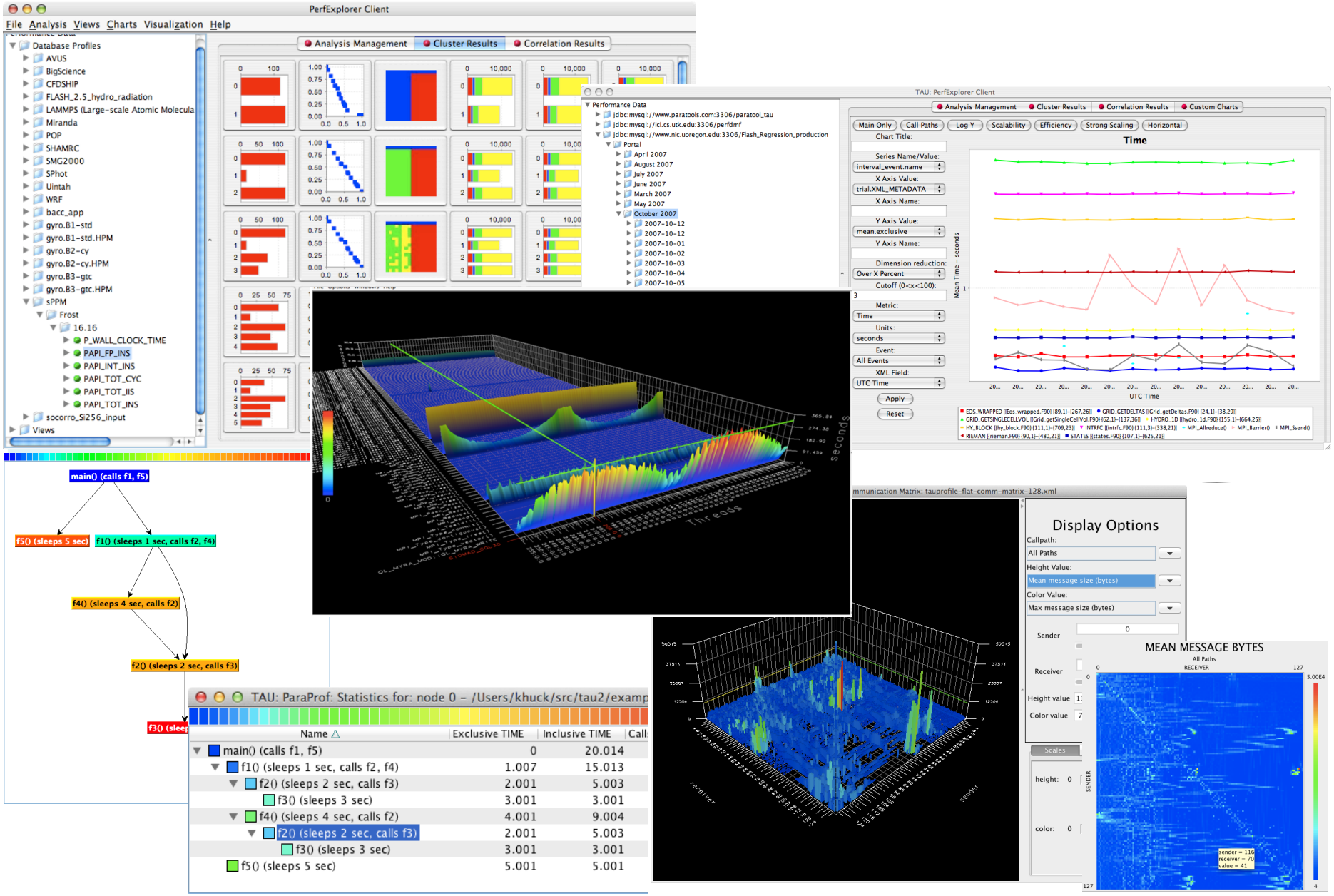


These examples (and more) available at:
https://wci.llnl.gov/simulation/computer-codes/visit/gallery

Background : TAU



- Tuning and Analysis Utilities (20+ year project)
- Performance problem solving framework for HPC
  - Integrated, scalable, flexible, portable
  - Target all parallel programming / execution paradigms
- Integrated performance toolkit
  - Multi-level performance instrumentation
  - Timers, counters, sampling, binary rewriting…
  - Flexible and configurable performance measurement
  - Widely-ported performance profiling / tracing system
  - Device support (CUDA, OpenCL, OpenACC, Intel Phi…)
  - Performance data management and data mining
  - Open source (BSD-style license)
- Broad use in complex software, systems, applications
- http://tau.uoregon.edu

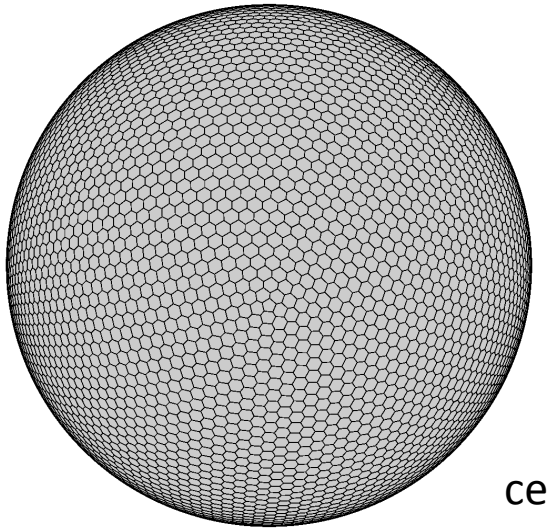


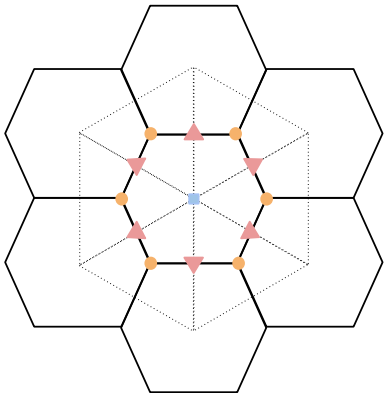
Background : MPAS-Ocean

- The Model for Prediction Across Scales (NCAR, LANL)
- Framework for rapid prototyping of single-component climate system models
- TAU was used to evaluate MPI approach, suggest improvements
- Linked with TAU, existing manual instrumentation mapped to TAU API
- http://mpas-dev.github.io
- Data domain organized as unstructured meshes, typically Spherical Centroidal Voronoi Tesselations (SCVTs).
- Mesh cells are arbitrary polygons, usually hexagons, decomposing the data in 2 dimensions.

Background : Load Imbalance

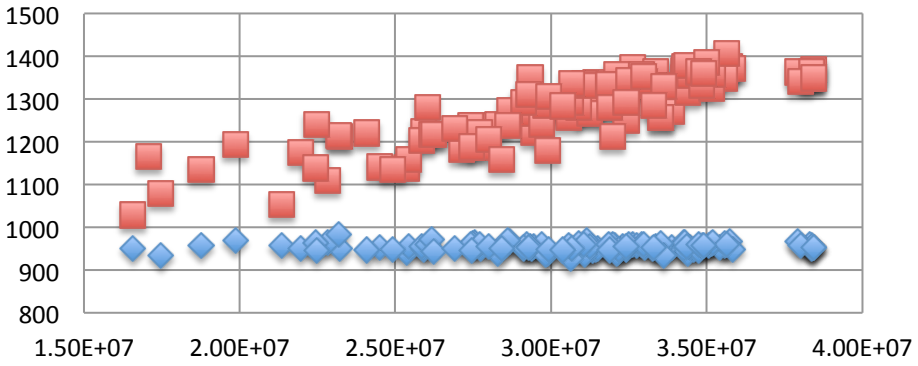
- Domain decomposition across MPI ranks: cells are organized into blocks, with (typically) ***three*** layers of halo cells from neighboring blocks.
- Blocks are partitioned ***only*** with respect to nCellsSolve (cells in block)
- TAU timers were integrated into MPAS using application timers and CAMTIMERS (CESM), and application metadata is also collected
- TAU PerfExplorer correlation analysis showed high correlation of computational imbalance and MPI synchronization with nCells (cells in block + halo cells) - each process has to solve for its explicitly assigned cells ***and*** its halo cells





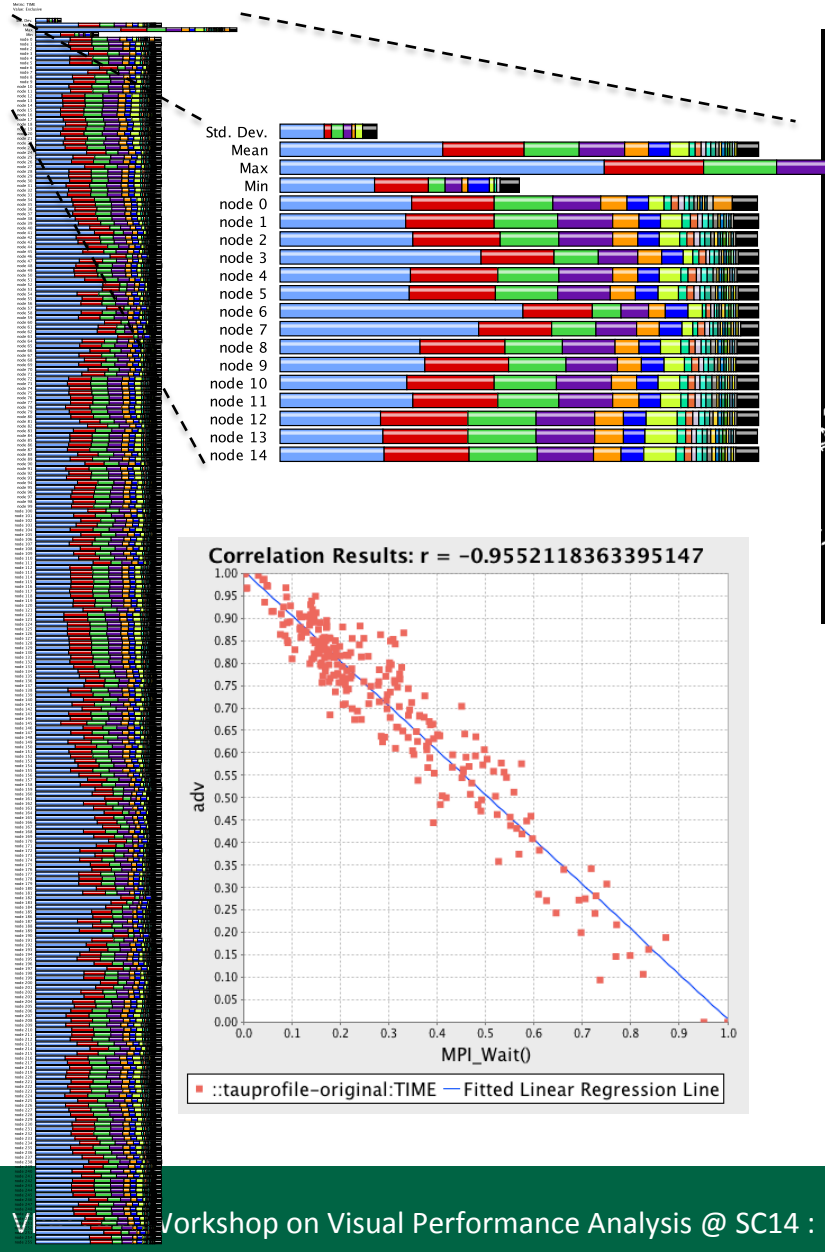
SCVT structure,
cell/mesh neighborhood

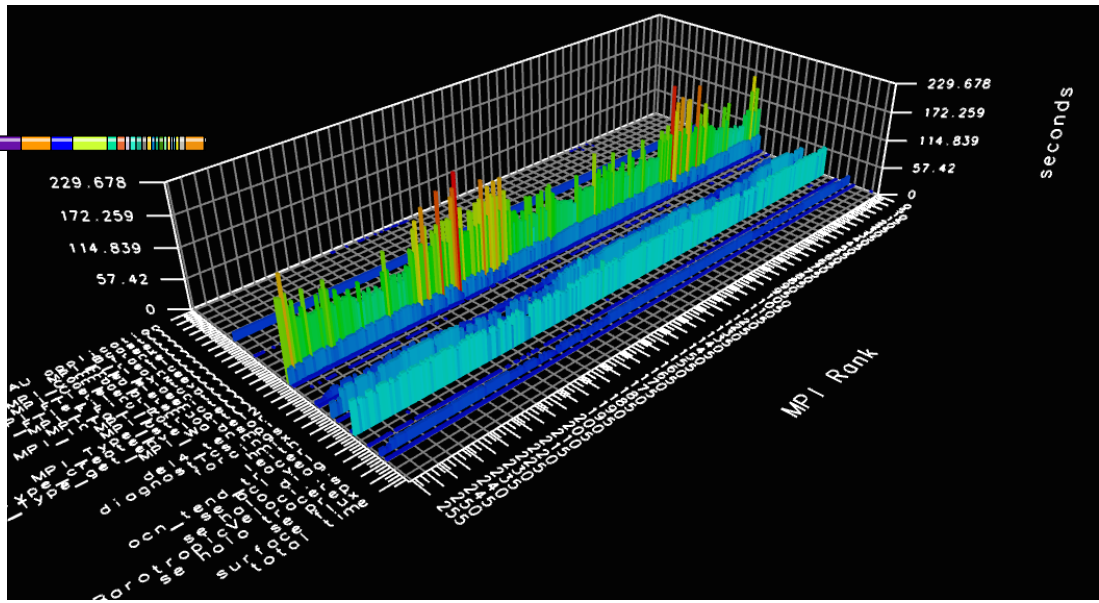
Metadata Correlated with Adv Timer



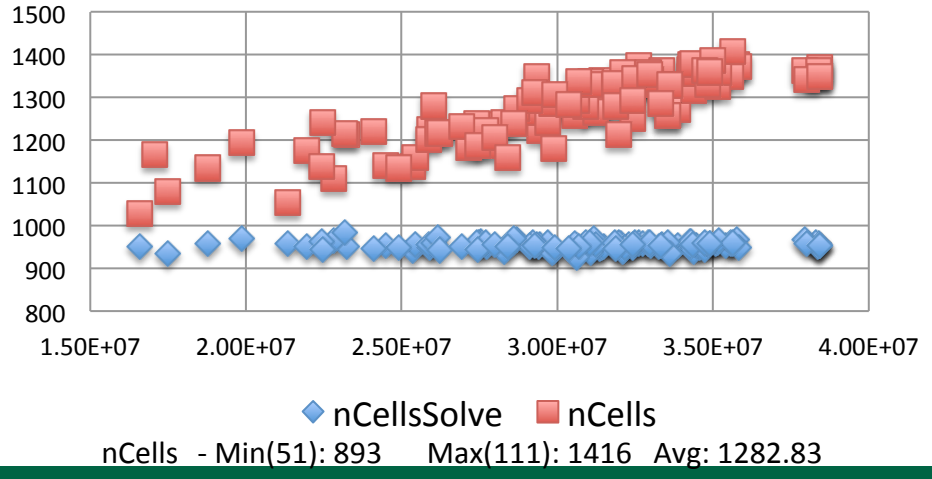
nCells - Min(51): 893 Max(111): 1416 Avg: 1282.83

Load Imbalance: TAU Visualizations





Metadata Correlated with Adv Timer

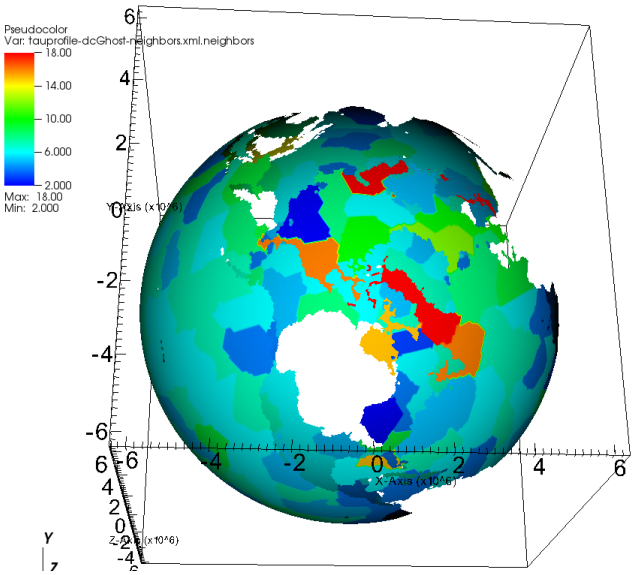


nCells - Min(51): 893 Max(111): 1416 Avg: 1282.83

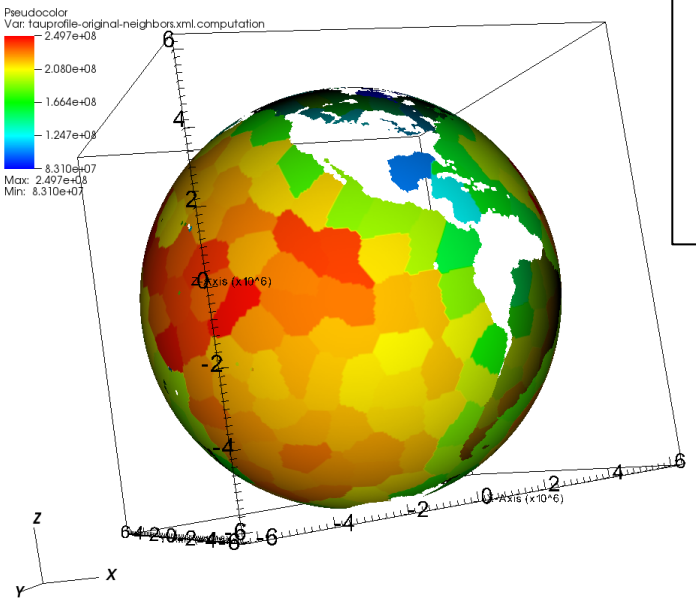
Approach

- New parser for VisIt: *.mpas index (meta) files
  - NetCDF input/output data file
    - n cell definitions, with simulation properties
  - Gpmetis partition file
    - n rows, one for each cell
    - Block ID for each cell in NetCDF data
    - Assume 1 block per process
  - Performance data summaries
    - p values, assume 1 value per process, to be matched up with each block
- PerfExplorer script to extract data from TAU profiles

DB: dcGhost-neighbors.mpas

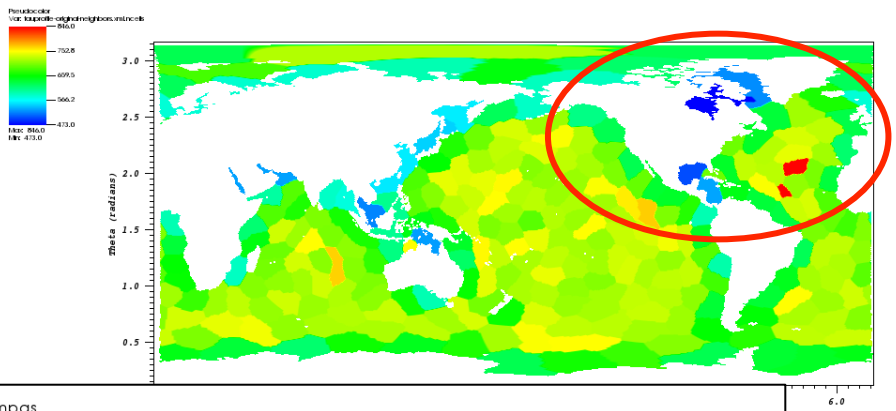


DB: 256-original.mpas



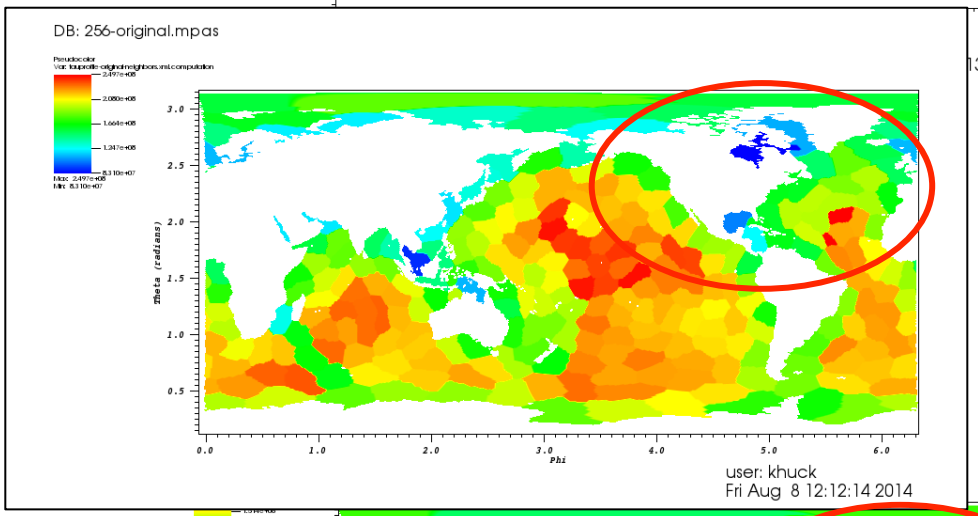
user: khuck
Fri Aug 8 11:55:31 2014

DB: 256-original.mpas

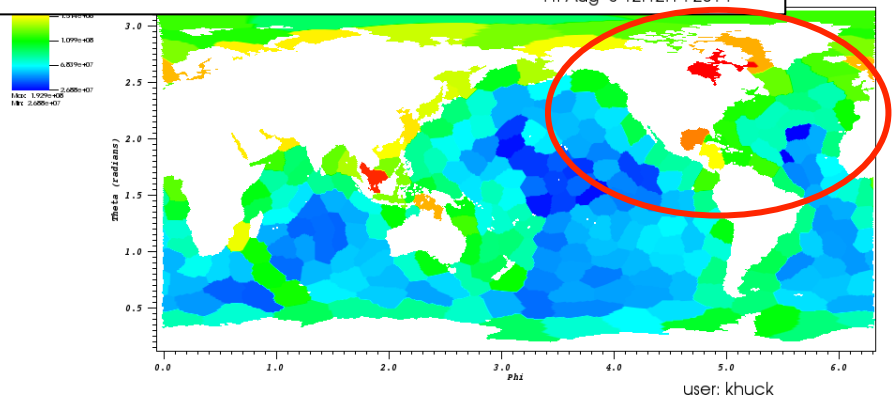


13:55 2014

DB: 256-original.mpas



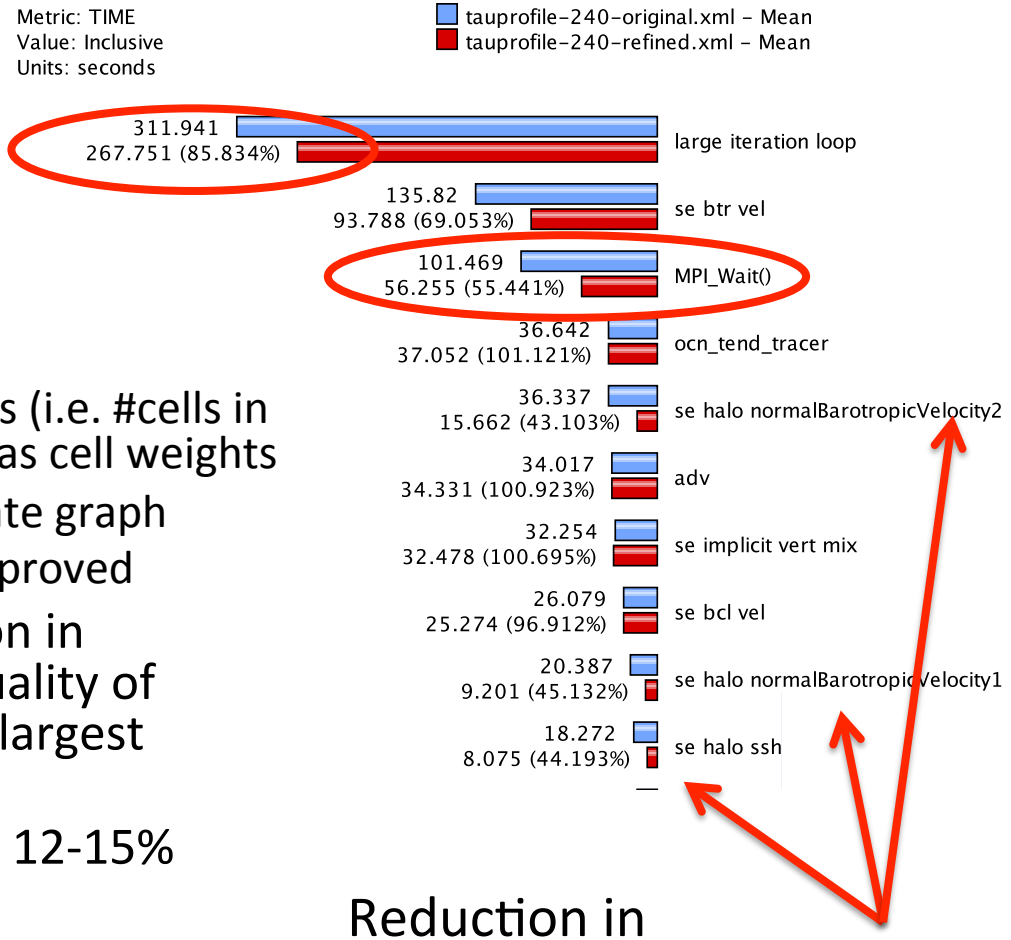
user: khuck
Fri Aug 8 12:12:14 2014



user: khuck
Fri Aug 8 12:12:51 2014

Experiments

- Visualize original partitioning of 60km data distributed into 240 blocks
- Evaluate *Hindsight* partitioner
- Optimizing Heuristic:
  - Partition with Metis
  - Compute properties of partitions (i.e. #cells in partition + #cells in halo) to use as cell weights
  - Assign weights to cells and update graph
  - Iterate until balance can't be improved
- Analysis suggests 2-12% reduction in execution time, depending on quality of initial partition (based on size of largest partition)
- TAU profile measurement shows 12-15% reduction in execution time
  - ~45% reduction in mean MPI_Wait() times for 240 process example on Edison



Reduction in communication routines due to better balance

Results Validation

Total cells per block

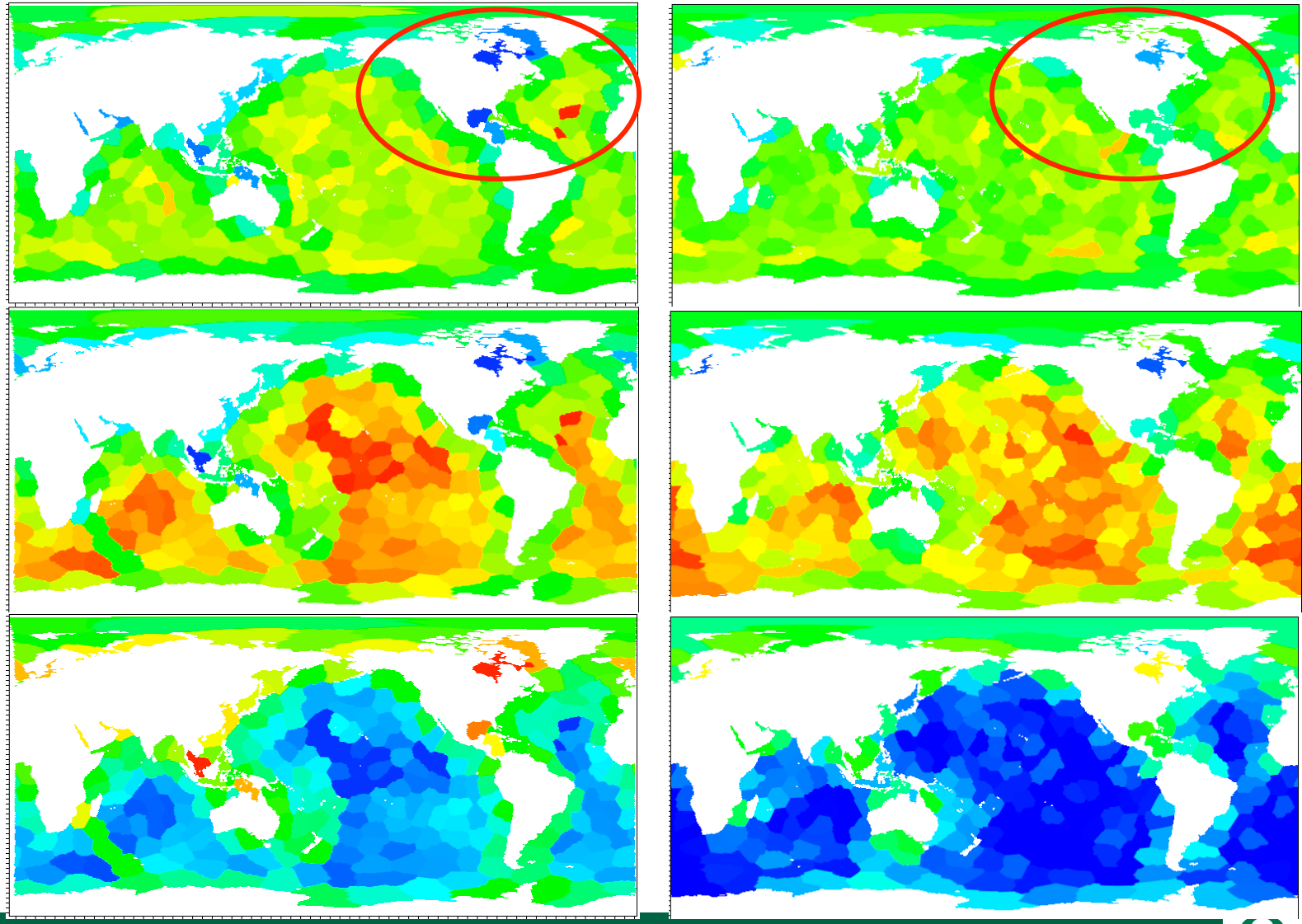
min:
473 ↗ 535
max:
846 ↘ 771

Time spent computing

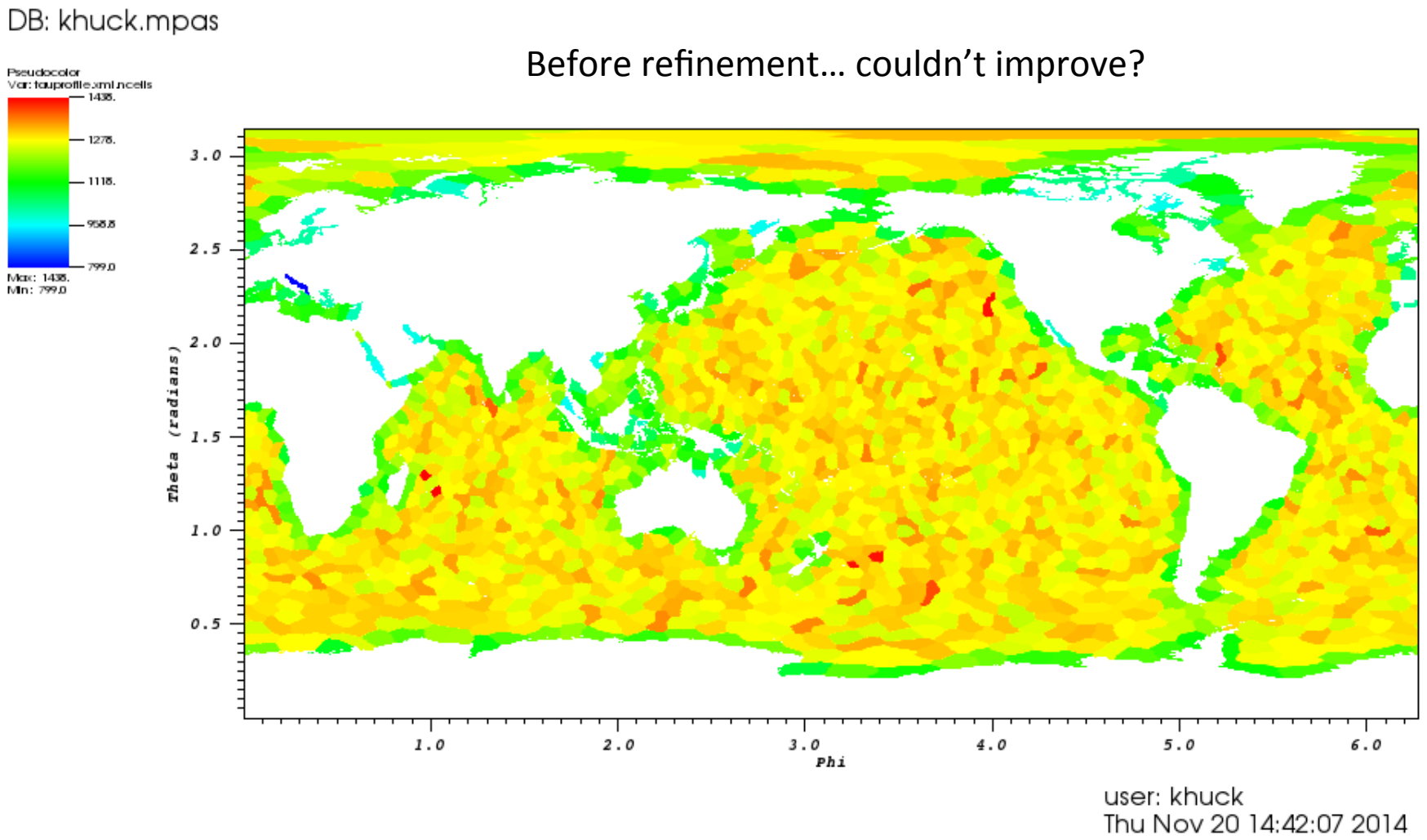
min:
8.3e7 ↗ 9.8e7
max:
2.5e8 ↘ 2.4e8

Time spent in MPI_Wait

min:
2.7e7 ↘ 9.0e6
max:
1.9e8 ↘ 1.5e8



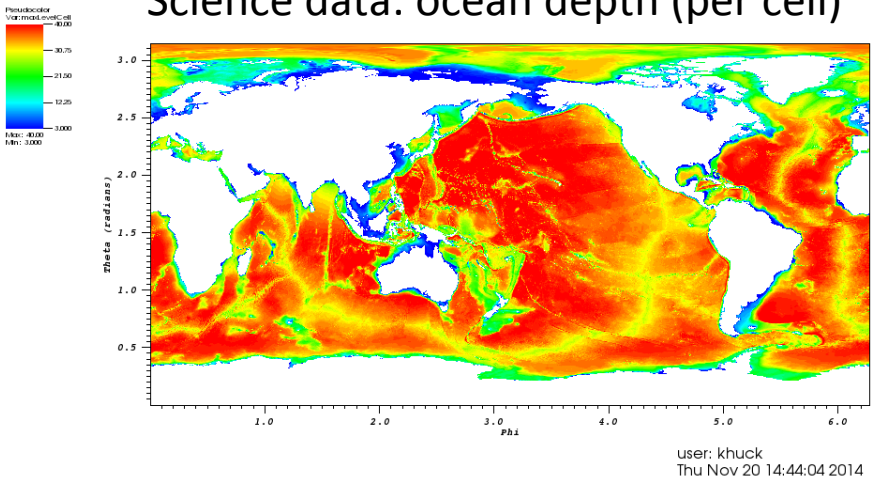
Problem Diagnosis : 2048 cells, 15km



2048/15km data: correlations

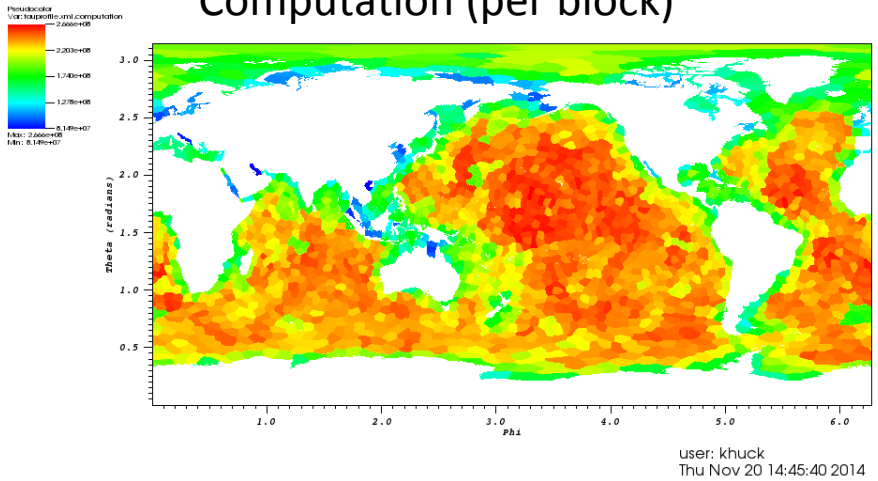
DB: khuck.mpas

Science data: ocean depth (per cell)



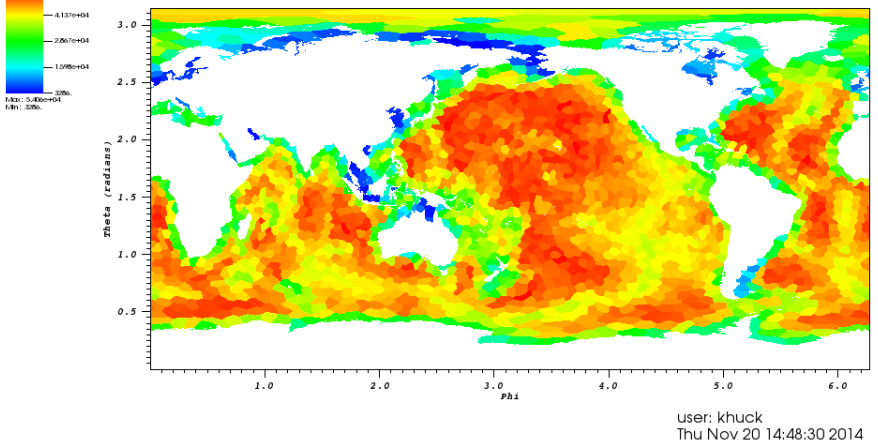
DB: khuck.mpas

Computation (per block)



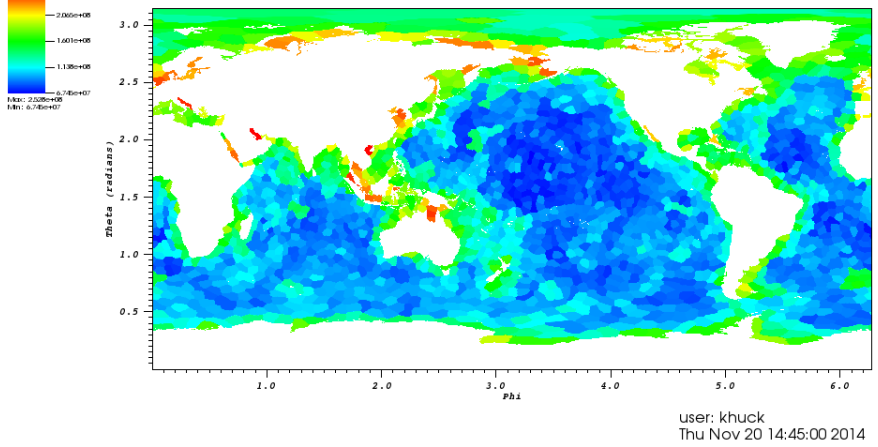
DB: khuck.mpas

Metadata: total_level_cells (per block)



DB: khuck.mpas

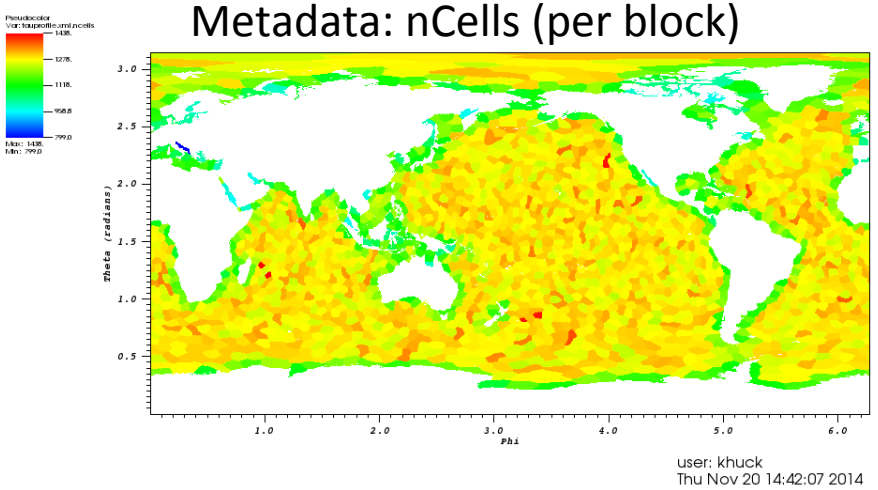
MPI_Wait (per block)



2048/15km data: original partitions

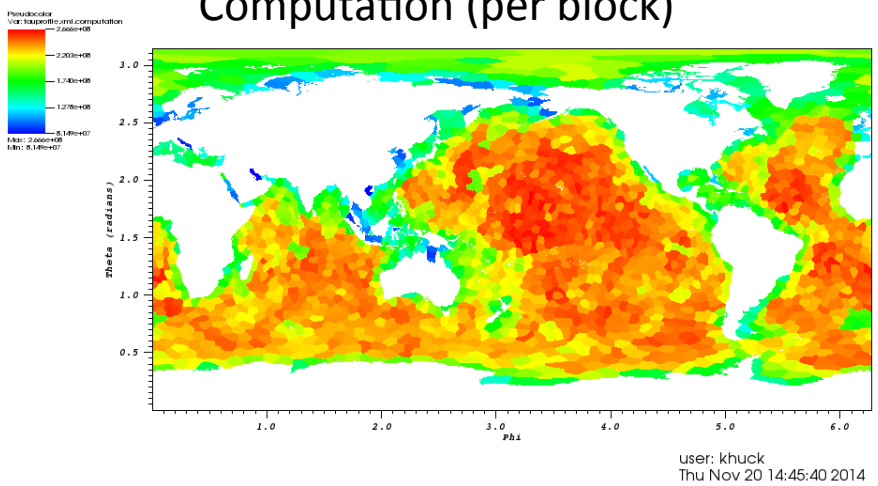
DB: khuck.mpas

Metadata: nCells (per block)



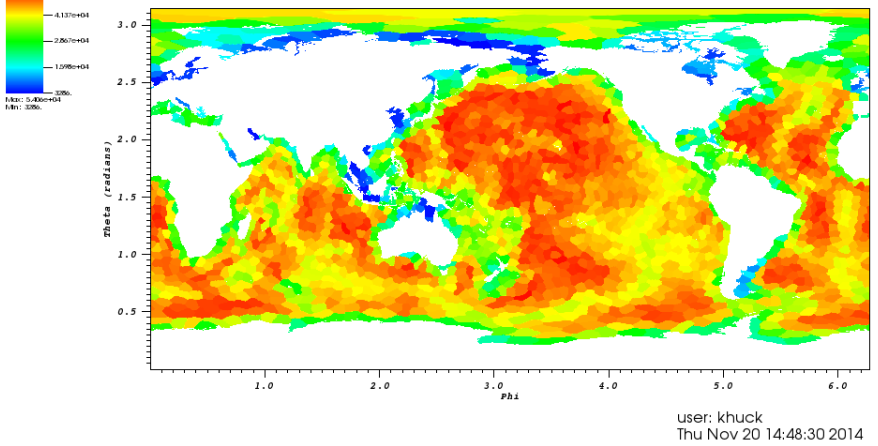
DB: khuck.mpas

Computation (per block)



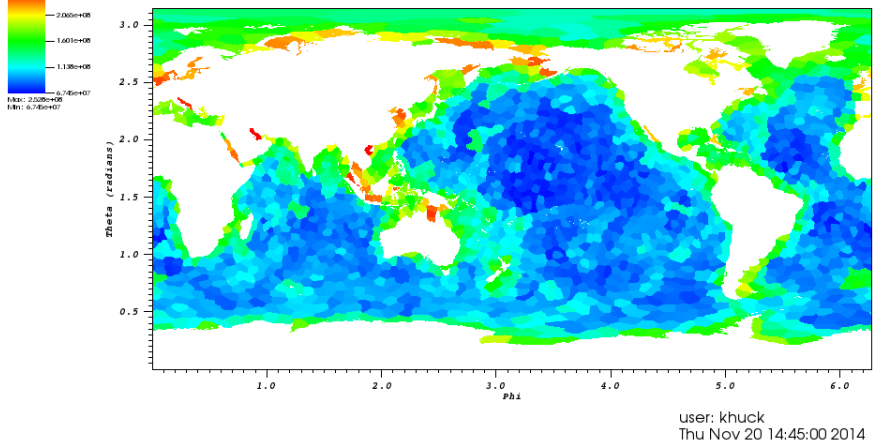
DB: khuck.mpas

Metadata: total_level_cells (per block)



DB: khuck.mpas

MPI_Wait (per block)

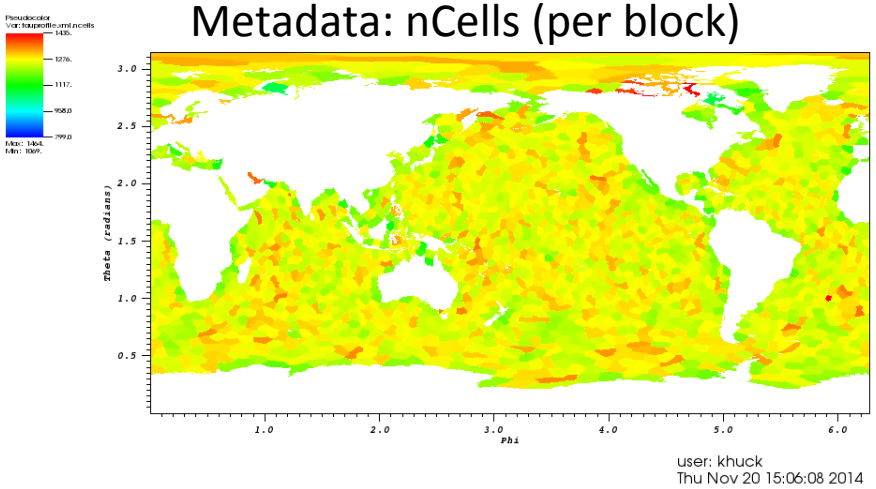


Modify Hindsight to bias on depth

- Partition using cell depth as initial weights
- bias weight with total cell depth in block

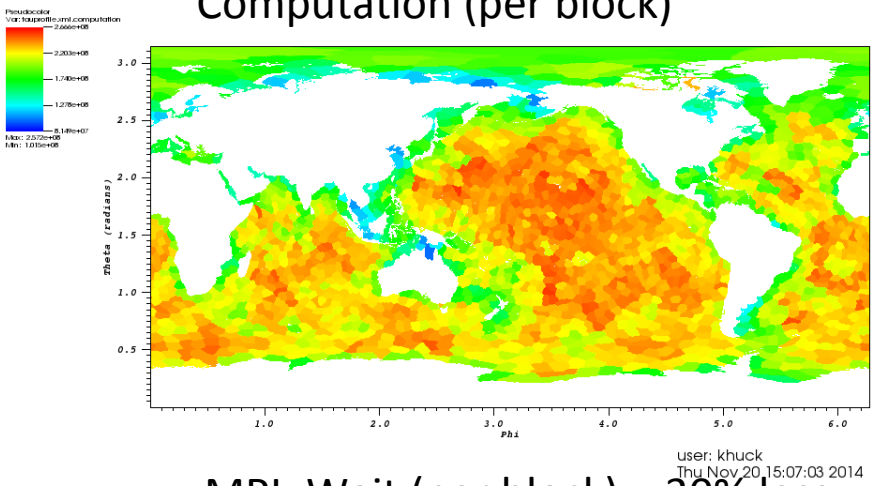
DB: khuck.mpas

Metadata: nCells (per block)



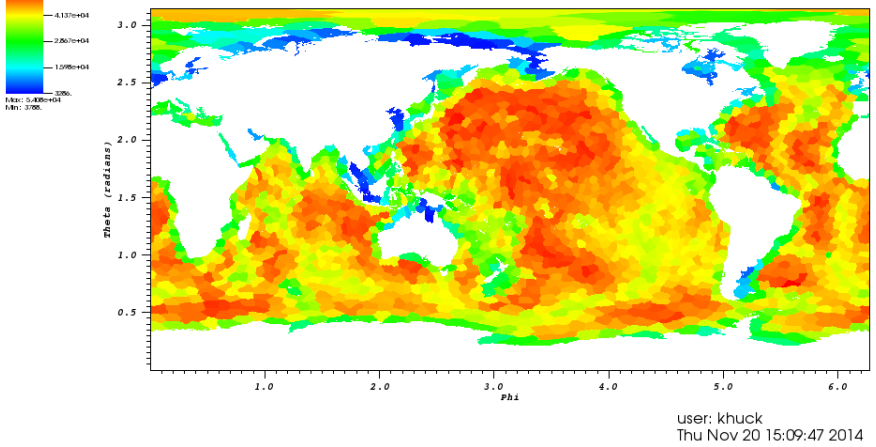
DB: khuck.mpas

Computation (per block)



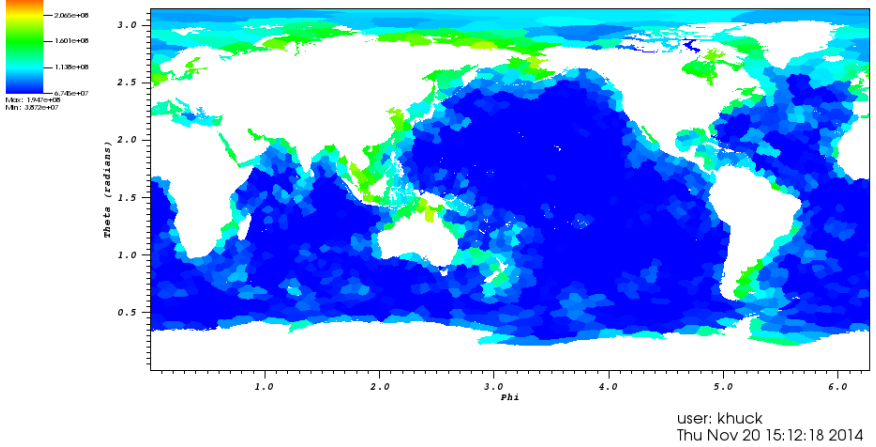
DB: khuck.mpas

Metadata: total_level_cells (per block)



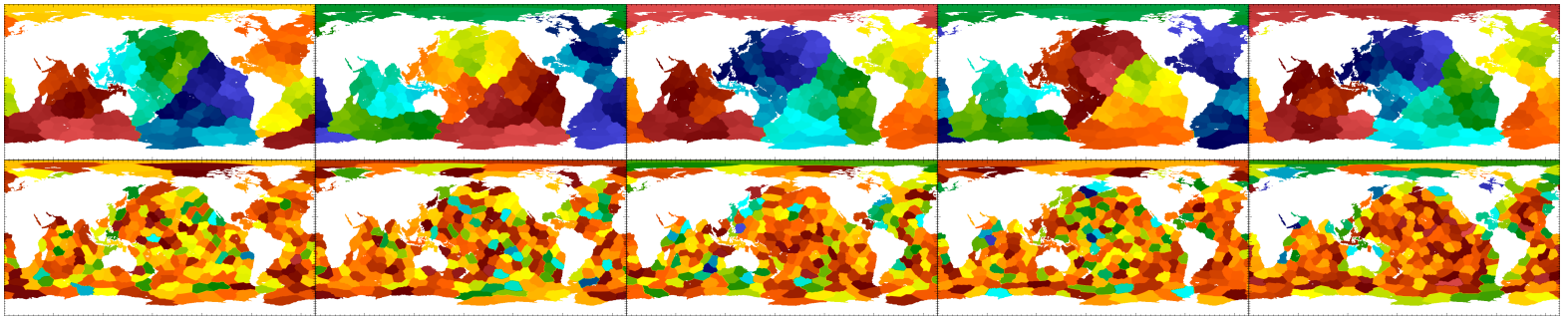
DB: khuck.mpas

MPI_Wait (per block) – 30% less, 11% reduction in execution time



Ensemble Visualization

- Time series data
- Parametric studies
- Example: evolution of block membership during 5 iterations of Hindsight



Summary

- Visualization in science domain helped understand and confirm load imbalances in MPAS-Ocean domain data
- Techniques ***broadly*** applicable to other applications (just need to add support)
- Any performance data & metadata can be mapped to the science domain – even could be (should be?) included in output file from simulation

Acknowledgements

- US DOE SciDAC
  - SUPER
  - SDAV
  - MULTISCALE
- US DOE Office of Science Biological and Environmental Research Program