Scientific Computing Group Support Model

Judith Hill

Liaison Task Lead, Scientific Computing Group

Oak Ridge Leadership Computing Facility (OLCF)

National Center for Computational Sciences (NCCS)

Lattice QCD Computational Science Workshop

April 29, 2013

Oak Ridge National Laboratory

U.S. DEPARTMENT OF ENERGY

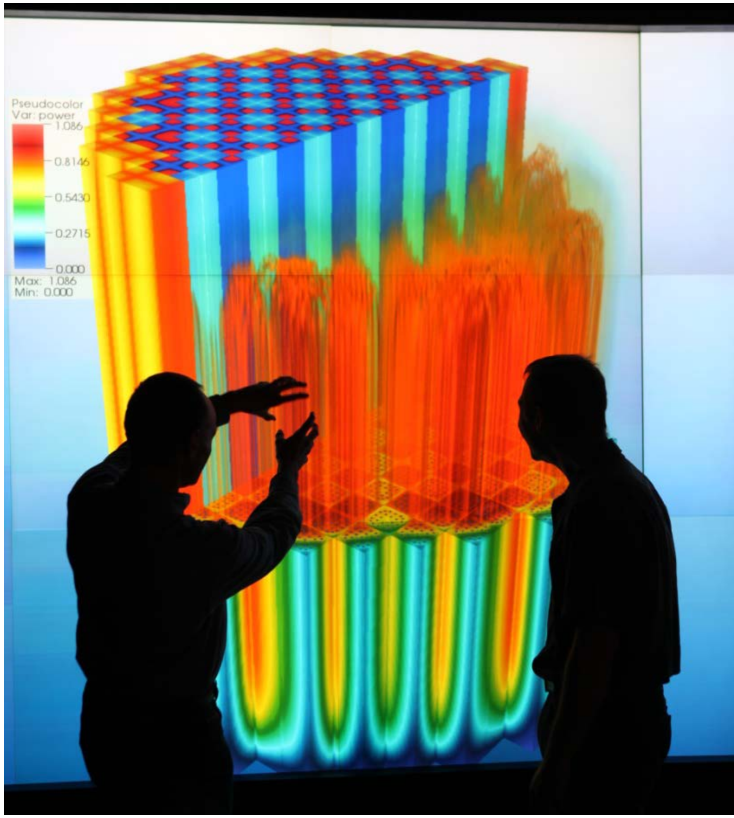
OAK RIDGE NATIONAL LABORATORY

MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY

Oak Ridge Leadership Computing Facility Mission

The OLCF is a DOE Office of Science National User Facility whose mission is to enable breakthrough science by:

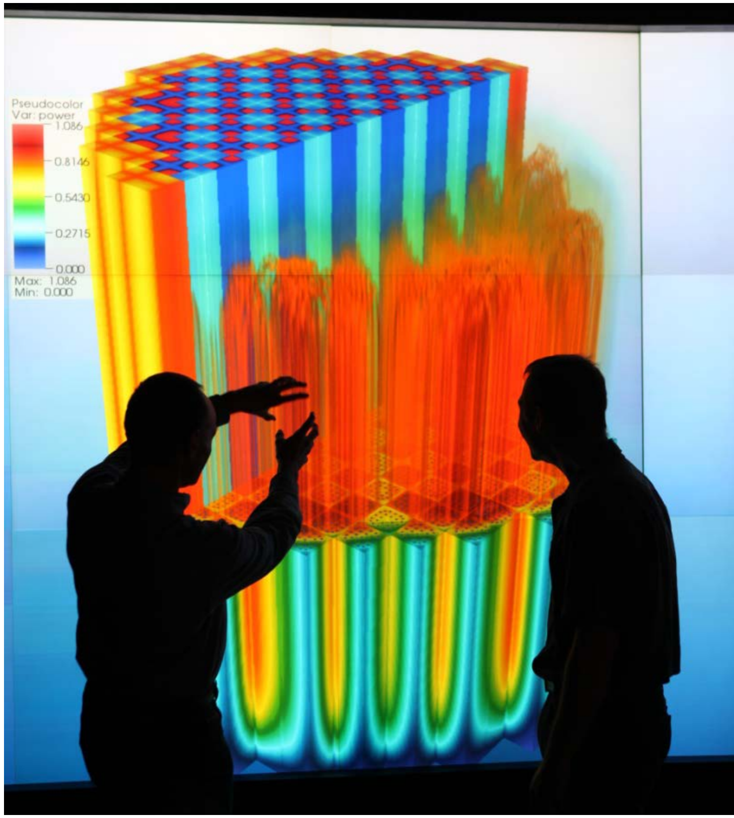
- Fielding the most powerful capability computers for scientific research,
- Building the required infrastructure to facilitate user access to these computers,
- Selecting a few time-sensitive problems of national importance that can take advantage of these systems,
- And partnering with these teams to deliver breakthrough science.



Oak Ridge Leadership Computing Facility Mission

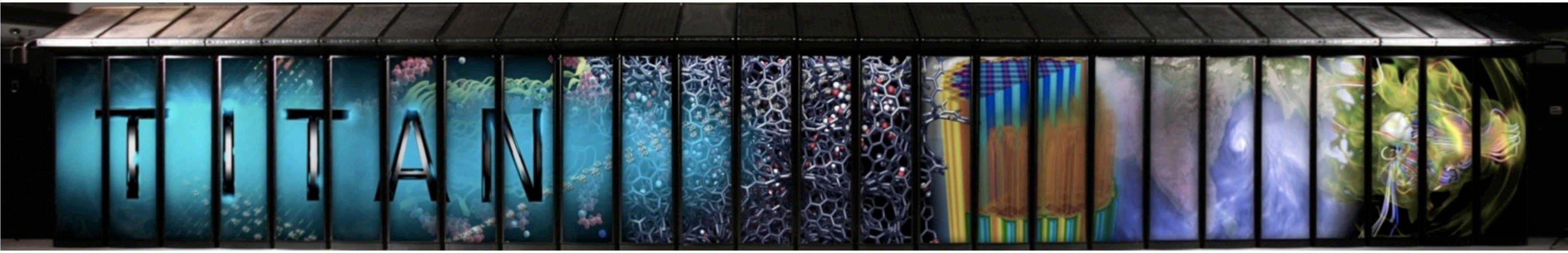
The OLCF is a DOE Office of Science National User Facility whose mission is to enable breakthrough science by:

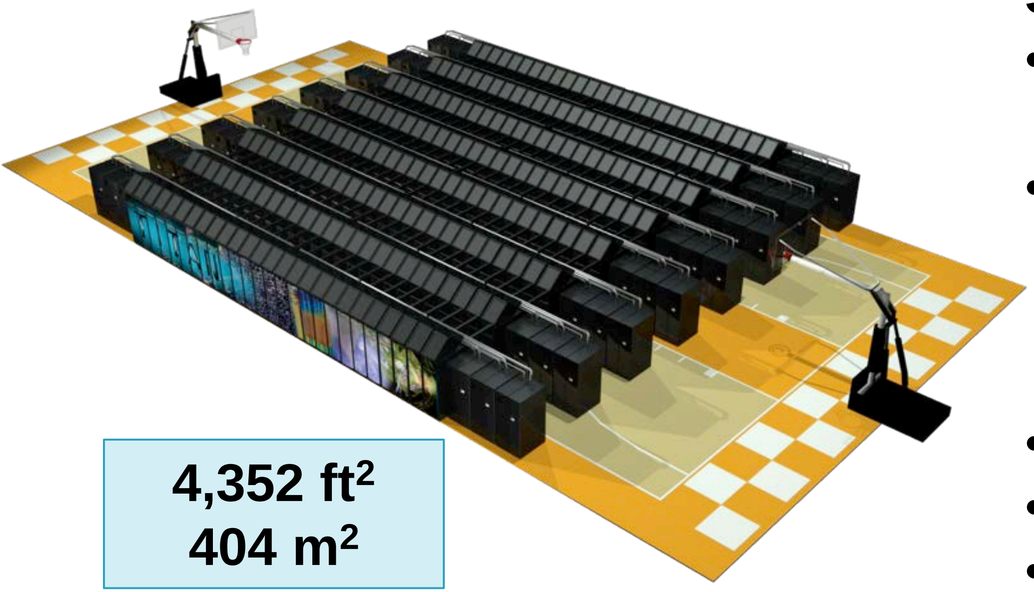
- Fielding the most powerful capability computers for scientific research,
- Building the required infrastructure to facilitate user access to these computers,
- Selecting a few time-sensitive problems of national importance that can take advantage of these systems,
- And partnering with these teams to deliver breakthrough science.



ORNL's "Titan" Hybrid System: World's Most Powerful Computer

#1 TOP500 SUPERCOMPUTER SITES





4,352 ft²
404 m²

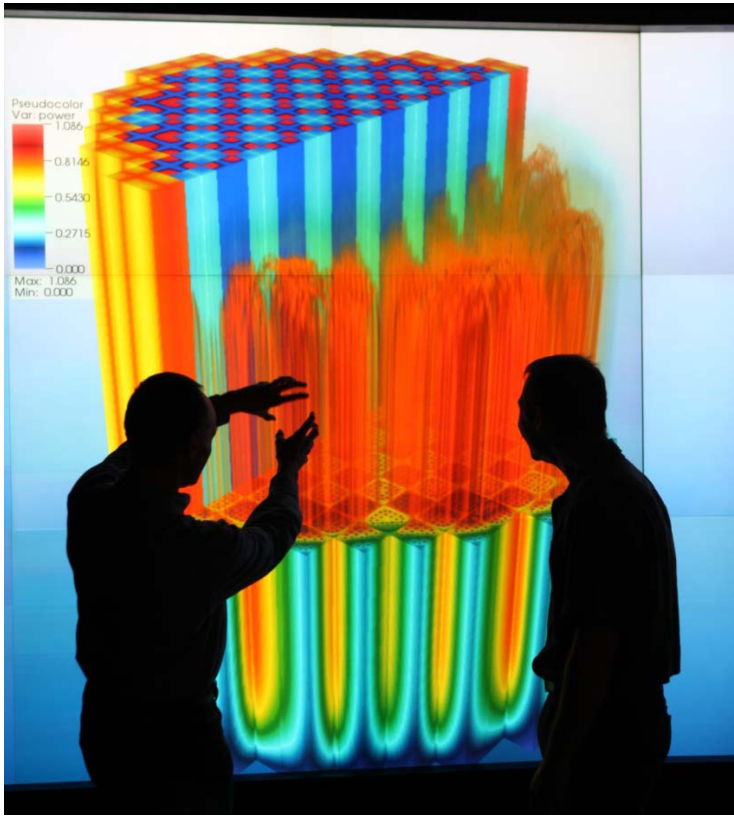
SYSTEM SPECIFICATIONS:

- Peak performance of 27.1 PF
  - 24.5 GPU + 2.6 CPU
- 18,688 Compute Nodes each with:
  - 16-Core AMD Opteron CPU
  - NVIDIA Tesla "K20x" GPU
  - 32 + 6 GB memory
- 512 Service and I/O nodes
- 200 Cabinets
- 710 TB total system memory
- Cray Gemini 3D Torus Interconnect
- 8.9 MW peak power

Oak Ridge Leadership Computing Facility Mission

The OLCF is a DOE Office of Science National User Facility whose mission is to enable breakthrough science by:

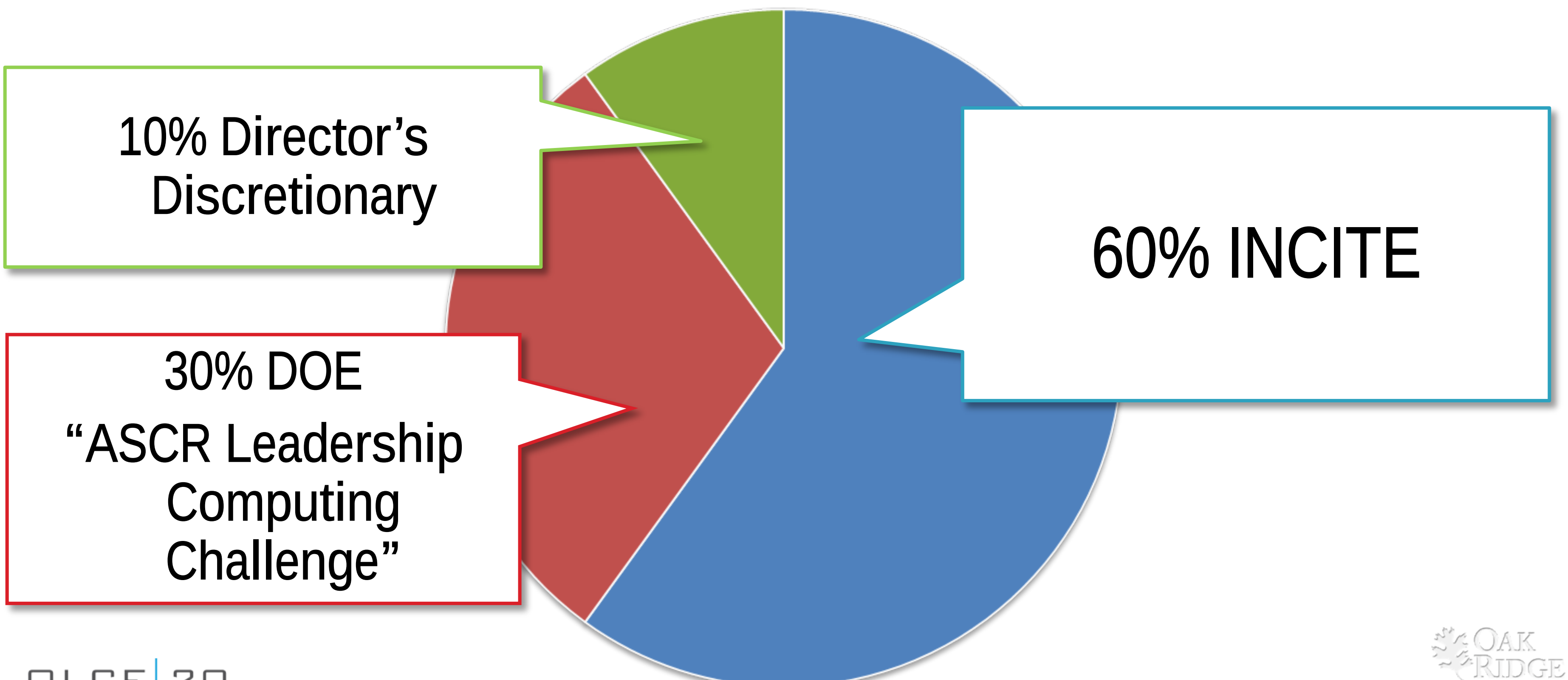
- Fielding the most powerful capability computers for scientific research,
- Building the required infrastructure to facilitate user access to these computers,
- Selecting a few time-sensitive problems of national importance that can take advantage of these systems,
- And partnering with these teams to deliver breakthrough science.



DOE Computational Facilities Allocation Policy for Leadership Facilities

Primary Objective:

- *"Provide substantial allocations to the open science community through an peered process for a small number of high-impact scientific research projects"*



Oak Ridge National Laboratory

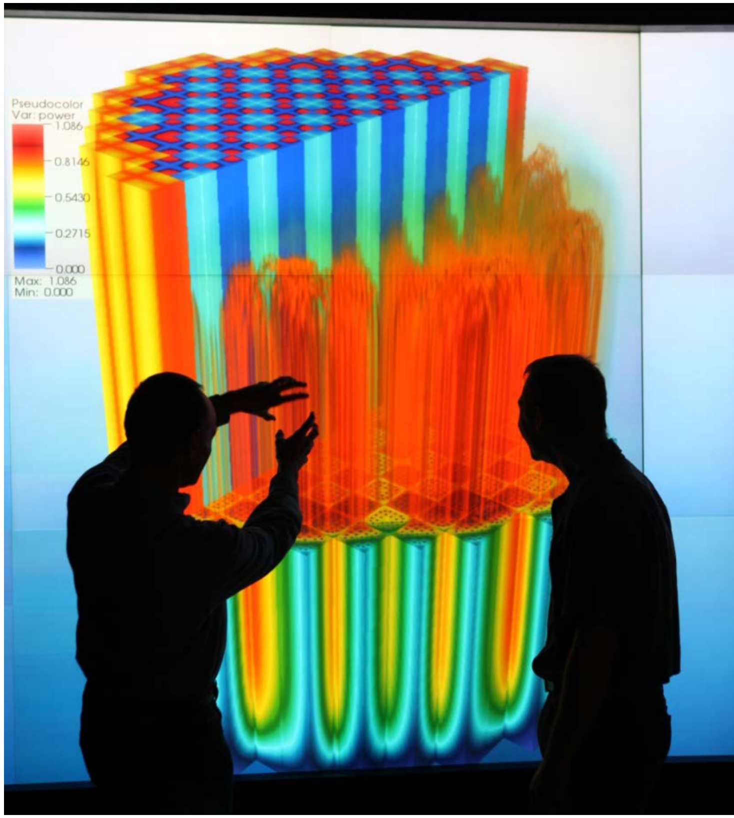
OLCF allocation programs
Selecting applications of national importance

	60% INCITE		30% ALCC		10% Director's Discretionary	
Mission	High-risk, high-payoff science that requires LCF-scale resources		High-risk, high-payoff science aligned with DOE mission		Strategic LCF goals	
Call	1x/year – (Closes June)		1x/year – (Closes February)		Rolling	
Duration	1-3 years, yearly renewal		1 year		3m,6m,1 year	
Typical Size	30 - 40 projects	20M - 100M core-hours/yr.	5 - 10 projects	1M – 75M core-hours/yr.	100s of projects	10K – 1M core-hours
Review Process	Scientific Peer-Review	Computational Readiness	Scientific Peer-Review	Computational Readiness	Strategic impact and feasibility	
Managed by	INCITE management committee (ALCF & OLCF)		DOE Office of Science		OLCF management	
Availability	Open to all scientific researchers and organizations including industry					

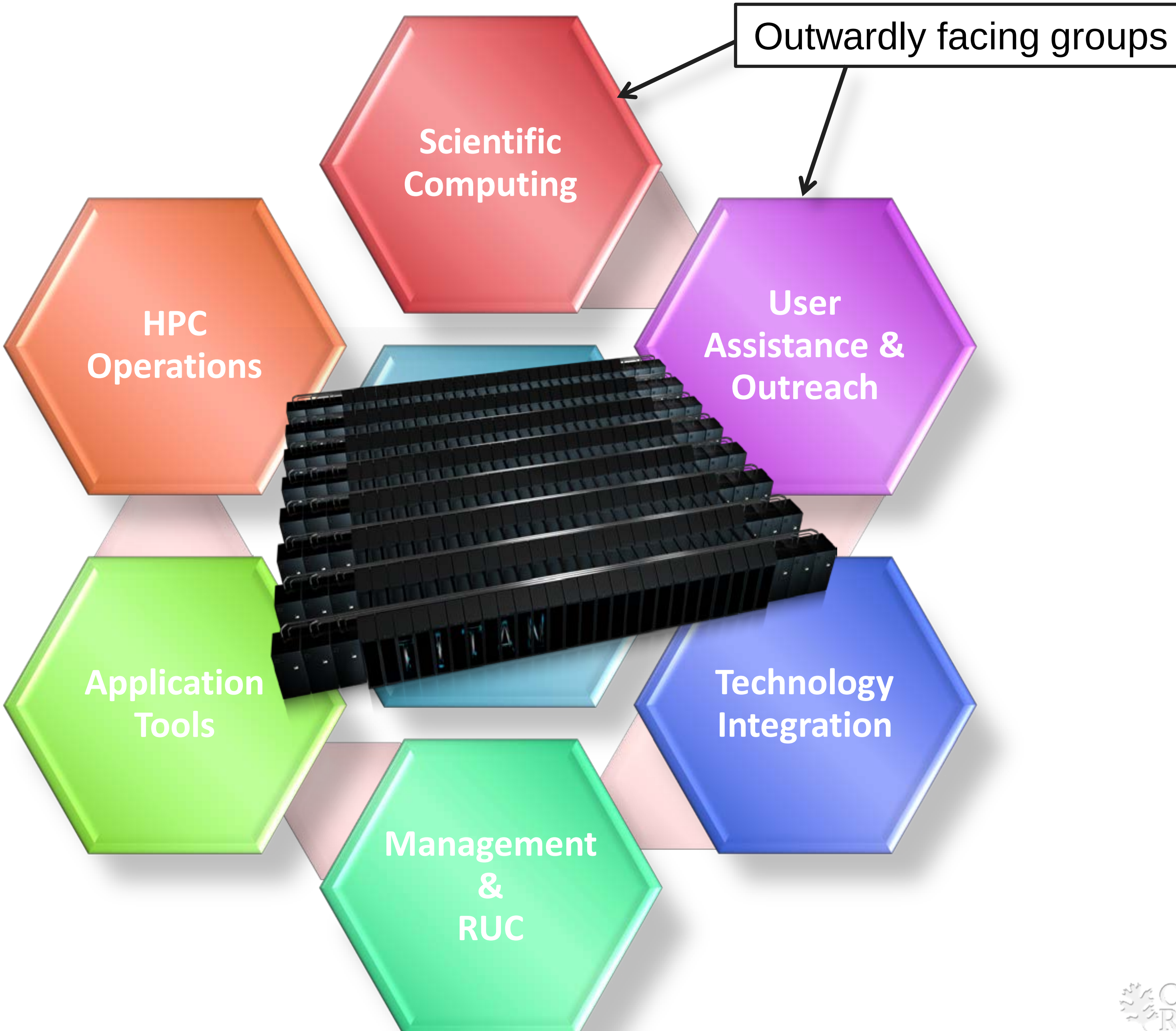
Oak Ridge Leadership Computing Facility Mission

The OLCF is a DOE Office of Science National User Facility whose mission is to enable breakthrough science by:

- Fielding the most powerful capability computers for scientific research,
- Building the required infrastructure to facilitate user access to these computers,
- Selecting a few time-sensitive problems of national importance that can take advantage of these systems,
- And partnering with these teams to deliver breakthrough science.



The OLCF


Outwardly facing groups
Scientific Computing
HPC Operations
User Assistance & Outreach
Application Tools
Technology Integration
Management & RUC

User Assistance and Outreach

User Assistance & Outreach

A. Carlyle	L. Moore[1]
A. DiGirolamo	S. Parete-Koon
F. Foertter	L. Rael
C. Fuson	S. Ray
B. Gajus[5]	B. Renaud
M. Griffith	D. Rose
J. Hines[#]	J. Rumsey[1]
S. Jones[5]	A. Simpson
M. Keele[1]	J. Smith
D. Levy[5]	R. Whitten[3,5]
M. Miller	L. Williams[5]

- Often your first stop when you enter the OLCF
  - Account setup
  - Training events
  - Website documentation
  - Basic technical support
  - Communications

help@olcf.ornl.gov

SciComp Group

Scientific Computing

M. Abbasi[5]
S. Ahern[3,5]
V. Anantharaj
M. Brown
J. Daniel
M. Eisenbach
M. Fahey[3,5]
J. Hill
W. Joubert[#]
S. Klasky[5]
Y. Li[7]
A. Lopez-Bezanilla[7]
M. Matheson
B. Messer
K. Mu[8]
H. Nam
T. Nguyen[7]
M. Norman
S. Pannala[5]
N. Podhorszki[5]
D. Pugmire
D. Rosenberg
R. Sankaran
R. Tchoua[5]
A. Tharrington[#]
Y. Tian[5]
R. Toedte
Z. Wang[7]

- ≥ 20 staff members
  - Some matrixed into our group
- ~ 4 post-docs or postmasters
- Expertise across a breadth of application domains
  - Common denominator is HPC

SciComp Group Tasks/Leads

Scientific Computing

M. Abbasi[5]
S. Ahern[3,5]
V. Anantharaj
M. Brown
J. Daniel
M. Eisenbach
M. Fahey[3,5]
J. Hill
W. Joubert[#]
S. Klasky[5]
Y. Li[7]
A. Lopez-Bezanilla[7]
M. Matheson
B. Messer
K. Mu[8]
H. Nam
T. Nguyen[7]
M. Norman
S. Pannala[5]
N. Podhorszki[5]
D. Pugmire
D. Rosenberg
R. Sankaran
R. Tchoua[5]
A. Tharrington[#]
Y. Tian[5]
R. Toedte
Z. Wang[7]

- Liaison (Judy Hill)
- Visualization (Dave Pugmire)
- End-to-End (Scott Klasky)

- Acceptance (Arnold Tharrington)
- Performance (Wayne Joubert)

Inwardly-Focused Tasks

- Acceptance Task
  - *Ensuring that new machines satisfy requirements*
  - Team across OLCF groups, including ~5 SciCompers
  - Very intensive testing schedule during acceptance period

- Performance Task
  - *Understanding how to use the machine and how the machine is used*
  - Improve performance of code
  - Study usage of machine
  - Inform next-generation architectures

Outwardly-Focused Tasks

- Liaison Task
  - *Furthering scientific accomplishments*
  - Collaborate with users to make codes run more efficiently, and produce more science
  - Advocate for users on center policies
- Visualization Task
  - *Enabling scientific discovery through visualization*
  - Creating tools for visualization
  - Developing visualizations in collaboration with users
- End-to-End Task
  - *Managing the computational inquiry lifecycle*
  - Inputting jobs on the machine, Managing jobs, Dealing with job output

Support Model for INCITE projects

- Liaison and Visualization tasks most outward-facing
  - End-to-end also involves users
  - Outward-facing performance tasks performed in liaison task
- Liaison and Visualization follow similar model

U.S. DEPARTMENT OF ENERGY
INCITE
LEADERSHIP COMPUTING

About Liaisons

- PhD-level scientists with expertise in computation
  - Astrophysics, biology, chemistry, climate, computer science, engineering, materials science, mathematics, nuclear physics, plasma physics, etc.
  - Experienced computational scientists with one thing in common
- Liaisons matched with INCITE projects based on science, mathematical, and algorithmic expertise
  - Can't always match for science first, e.g., I am not a chemist, but I am generally familiar with their math and algorithms
- Our motto: Whatever it takes!

Liaison Role

- Liaisons are collaborators whose unique expertise with leadership-level computers can enhance your experience and help you get more science done
- Levels of liaison support
  - Level 1: User support +
  - Level 2: Paratrooper – fix a specific problem in your code, O(1 month)
  - Level 3: Embedded member of code development team and science collaborator

Opportunities for Engagement

- Typical liaison activities
  - Profile code performance, providing feedback to code team
  - Code porting
  - Implement solutions to problems experienced by application scientists
  - Advocate for users regarding tools, libraries, etc.
  - Collaborate scientifically

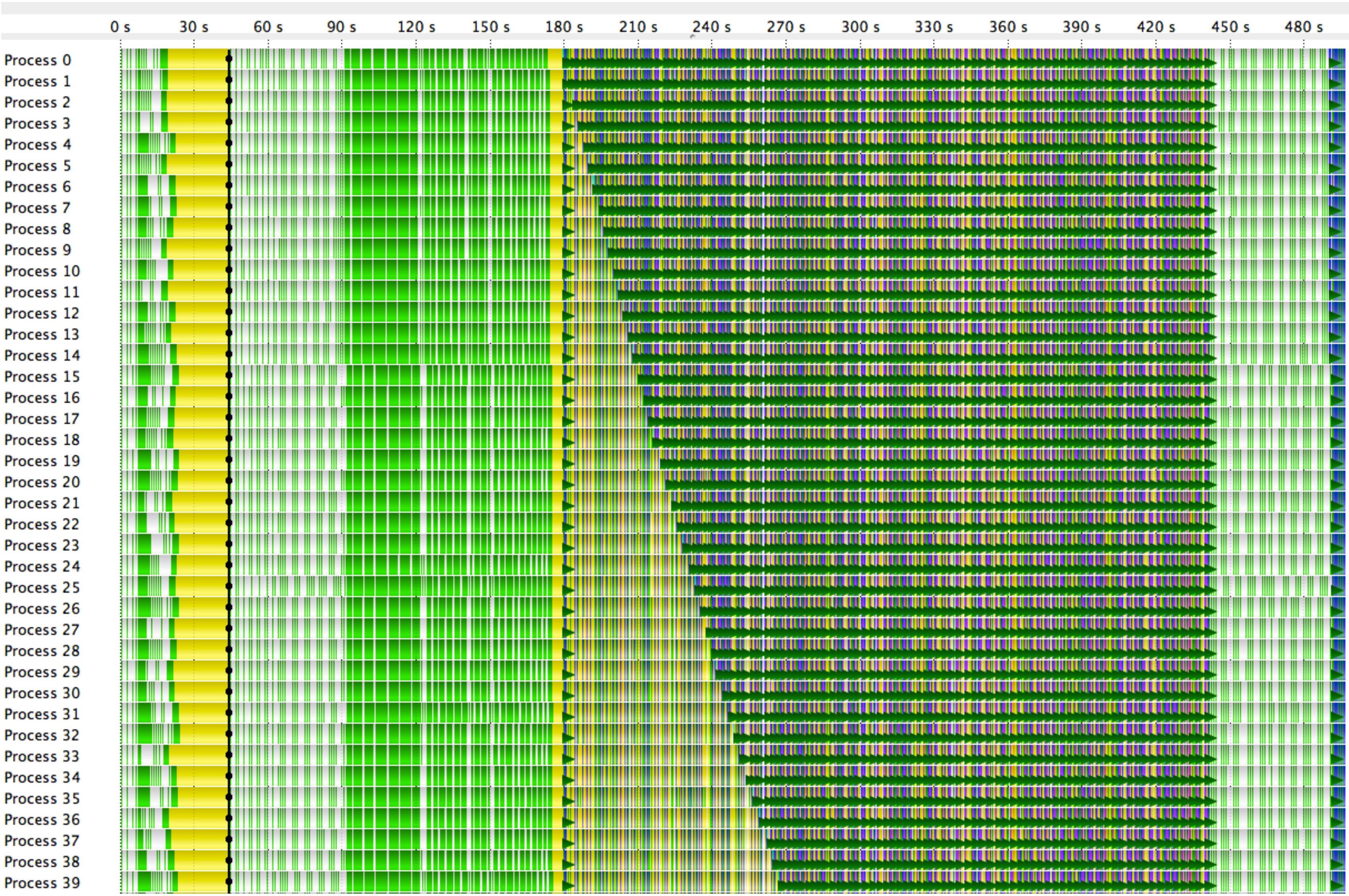
Opportunities for Engagement

- Typical liaison activities
  - Profile code performance, providing feedback to code team
  - Code porting
  - Implement solutions to problems experienced by application scientists
  - Advocate for users regarding tools, libraries, etc.
  - Collaborate scientifically

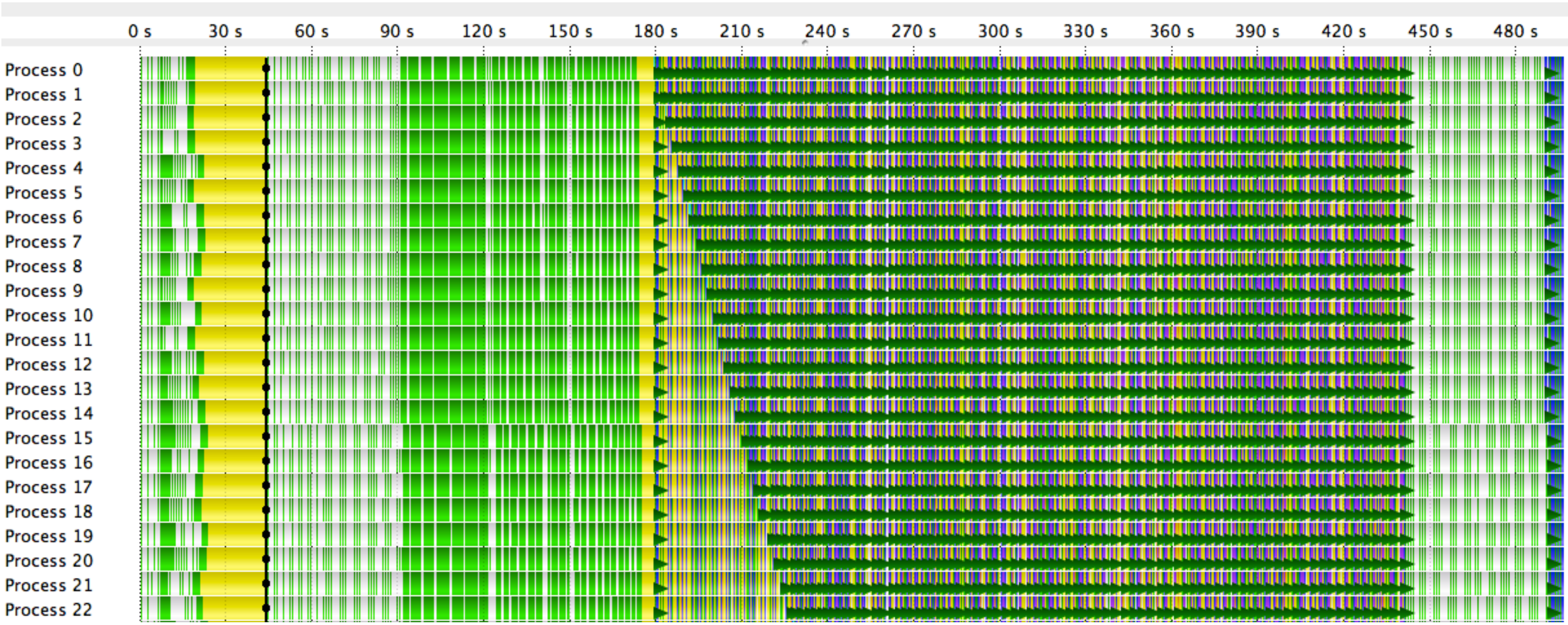
Example 1: BigStick (R. Hartman-Baker)

- R. H-B profiled Bigstick, a configuration interaction nuclear physics code using Vampir/VampirTrace to create a graphical representation of code performance
  - Promising for large-scale nuclear configuration calculations, because it does not require as much memory
  - ➢ Challenge: It doesn't scale well

Top-Level Overview



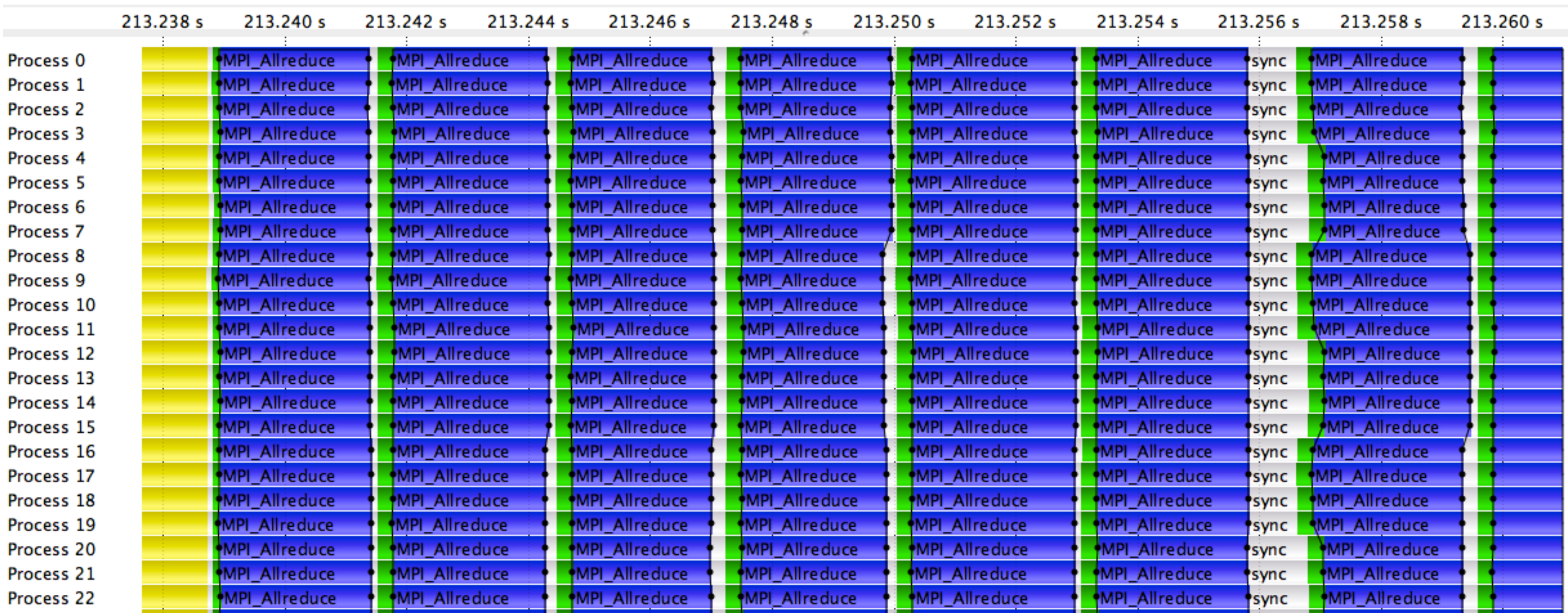
Top-Level Overview



- Triangular pattern reminiscent of sequential algorithm applied across processors
  - Digging deeper shows in orthogonalization phase, processors held up by single processor writing to Lanczos vector file
  - *Suggestion: reduce amount of orthogonalization performed*

Block Reduce Phase

Block Reduce Phase



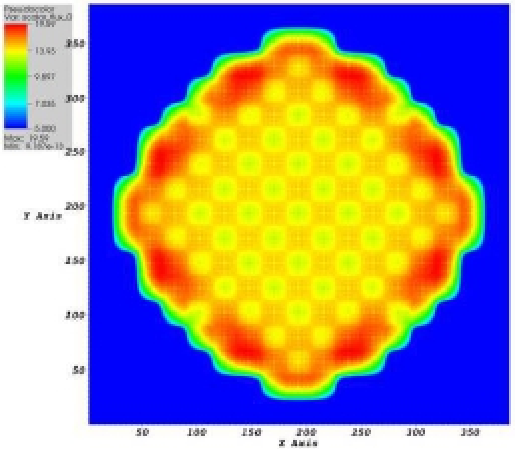
- Majority of time in block reduce phase spent in MPI_Allreduce
  - *Suggestion: Combining Allreduces could improve performance*

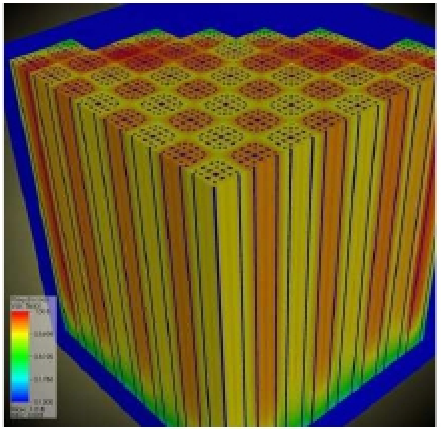
Opportunities for Engagement

- Typical liaison activities
  - Profile code performance, providing feedback to code team
  - Code porting
  - Implement solutions to problems experienced by application scientists
  - Advocate for users regarding tools, libraries, etc.
  - Collaborate scientifically

Example 2: Denovo (W. Joubert)

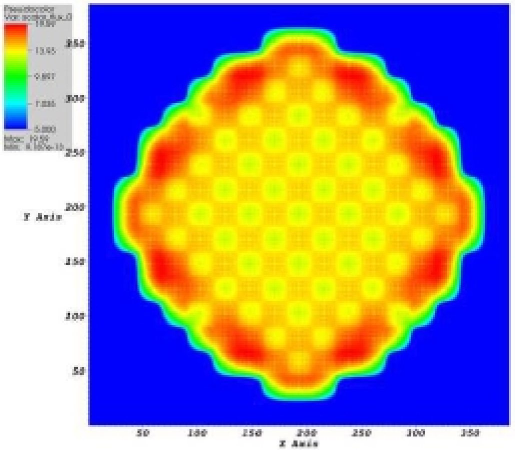
- W.J. developed and implemented a new, efficient sweep algorithm for Denovo that scales on Titan
- Denovo is a radiation transport code used in advanced nuclear reactor design
  - It solves for the density of particle flux in a 3-D spatial volume such as a reactor
  - In particular, it solves the six-dimensional linear Boltzmann equation (3-space, 2-angle, 1-energy)
- Denovo scaled up to 200K cores on ORNL's 2.3PF Jaguar system.
  - It was selected as an early port code for Titan

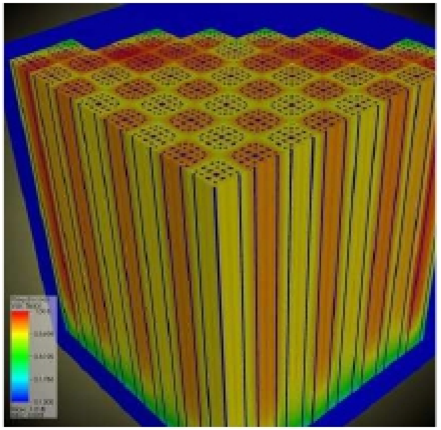




Example 2: Denovo (W. Joubert)

- Primary algorithms: the discrete ordinates method, 3-D sweep, GMRES linear solver and various eigensolvers, e.g., Arnoldi
- The execution time profile has a very prominent peak: nearly all the execution time (80-99%) is spent in a 3-D sweep algorithm.
- Because of this, the 3-D sweep must be the central focus of any effort to port Denovo to an accelerator-based system
- However, the sweep is a complex algorithm that is difficult to parallelize efficiently.



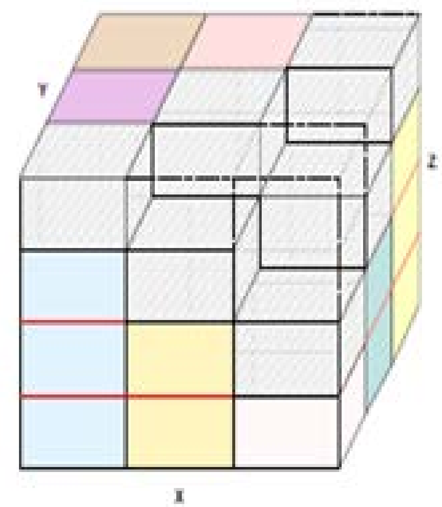


How to Program the Sweep on the GPU?

- Decide what language / parallel API to use to program the GPU.
  - 1. CUDA: a minor extension of C/C++ for GPU thread programming, also available for Fortran 90
  - 2. OpenCL: a multi-vendor standard similar to CUDA
    - NVIDIA support OpenCL, but going forward CUDA will be better supported and more in-sync with new hardware features.
  - 3. Compiler directives: similar to OpenMP (PGI, CAPS, Cray, ...)
    - Sweep is a complex algorithm, with many dimensions. Directives may not be flexible enough or expose enough hardware functionality to get the needed performance.

✓ *Thus use CUDA. Use C++ for consistency with Denovo base language.*

Mapping the Algorithm to the GPU



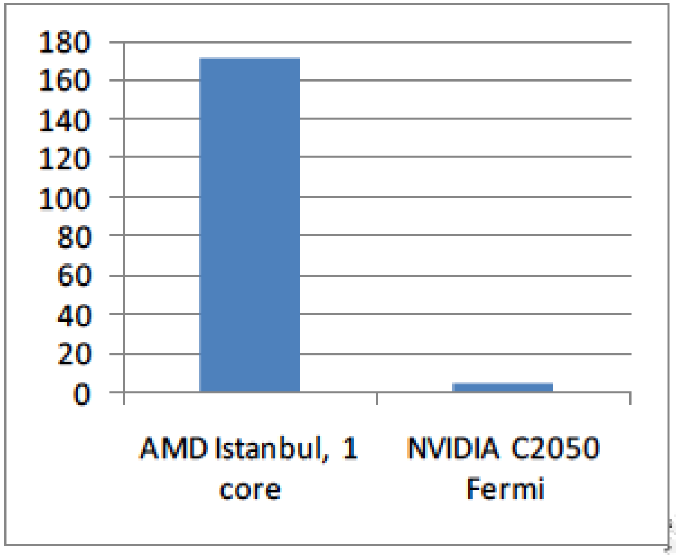
- Denovo has many candidate dimensions for parallelism: space (3), energy, moment/angle, octant, and also unknown (4 unknowns per gridcell for this discretization).
- NVIDIA suggests 4K-8K threads for the GPU to keep all GPU streaming multiprocessors busy and cover various latencies.
- Need the right kind of parallelism – *proper decoupling of data.*
- Need good memory access patterns (reuse of data loaded from global memory, coalesced stride-1 memory references, good use of registers, shared memory, caches on the GPU).
- *Approach: explore each problem dimension for potential thread parallelism.*

Results: Sweep GPU Performance

- Single core (AMD Istanbul) / single GPU (Fermi C2050) comparison

	AMD Istanbul 1 core	NVIDIA C2050 Fermi	Ratio
Kernel compute time	171 sec	3.2 sec	**54X**
PCIe-2 time (faces)	--	1.1 sec	
TOTAL	**171 sec**	**4.2 sec**	**40X**

NVIDIAFermi is 40X faster than single Opteron core



Opportunities for Engagement

- Typical liaison activities
  - Profile code performance, providing feedback to code team
  - Code porting
  - Implement solutions to problems experienced by application scientists
  - Advocate for users regarding tools, libraries, etc.
  - Collaborate scientifically

Example 3: VPIC (V. Anantharaj)

- VPIC: 3-D electromagnetic relativistic particle-in-cell simulation code
  - Application scientists reporting failure at high core counts for unknown reasons
- *Problem: Libraries were incompatible because of environment version changes*
  - Found because of liaison in collaboration with colleages H. Nam and R. Hartman-Baker
- *Solution: recompiling with fresh environment and proper modules loaded eliminated the error*

Opportunities for Engagement

- Typical liaison activities
  - Profile code performance, providing feedback to code team
  - Code porting
  - Implement solutions to problems experienced by application scientists
  - Advocate for users regarding tools, libraries, etc.
  - Collaborate scientifically

Examples of User Needs

- Queue Priority
  - Liaisons have science-area expertise, and understand unique needs of users
  - Advocate and explain why exception is needed
  - Advocate for users whose allocations have been exhausted

- Software Needs
  - Advocate for users' software needs
  - Serve as translator between projects and Resource Utilization Council (RUC) and software committee

Opportunities for Engagement

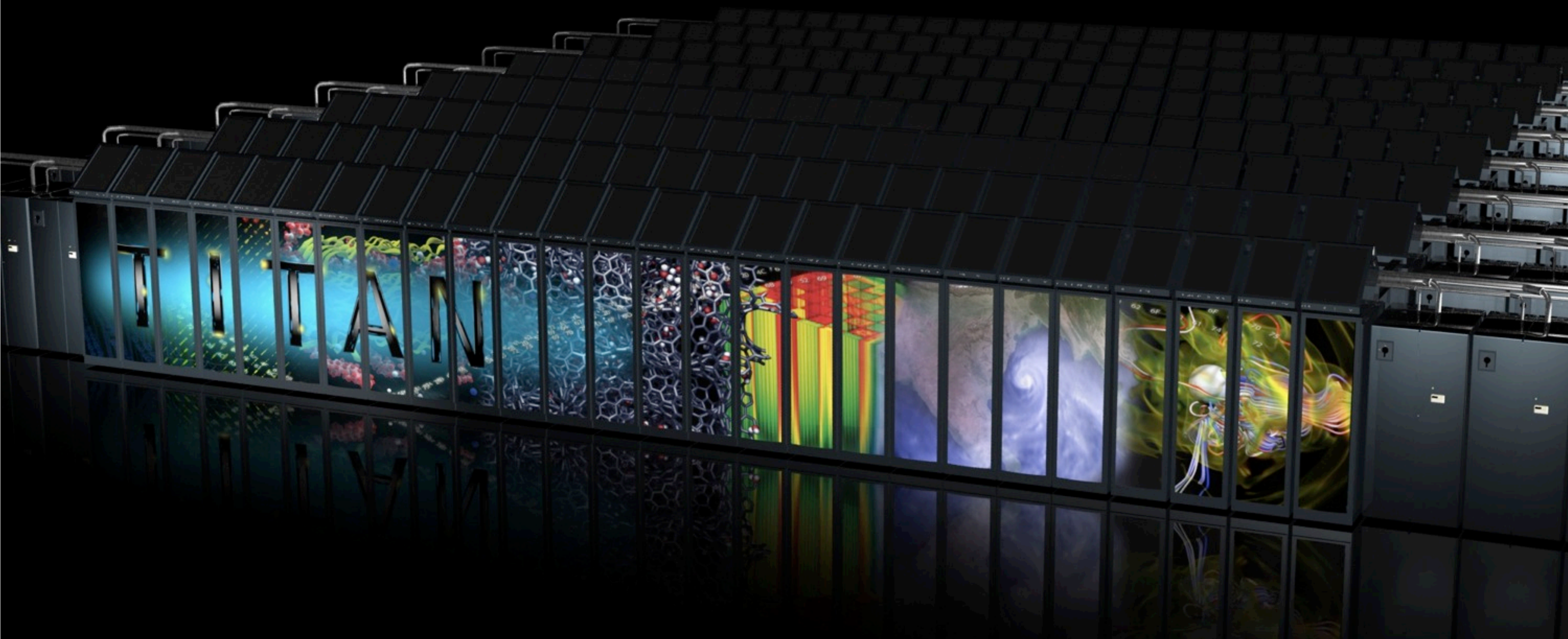
- Typical liaison activities
  - Profile code performance, providing feedback to code team
  - Code porting
  - Implement solutions to problems experienced by application scientists
  - Advocate for users regarding tools, libraries, etc.
  - Collaborate scientifically
    - *Too numerous to count (in the interest of time), but H. Nam (workshop co-organizer) exemplifies this*

Conclusions

- Liaisons are a valuable resource for INCITE projects
- Liaisons have unique HPC skills that projects can take advantage of
- *Whatever it takes!*

- Acknowledgements:
  - H. Nam and W. Joubert
  - Scientific Computing group

Questions?



Judith Hill
hilljc@ornl.gov