# NRC Workshop on NASA's Modeling, Simulation, and Information Systems and Processing Technology


**Bronson Messer**

**Director of Science**

**National Center for Computational Sciences**

**&**

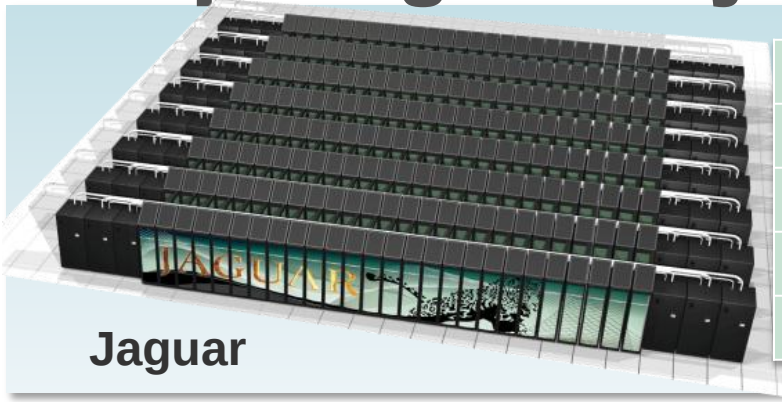
**Senior R&D Staff**

**Oak Ridge National Laboratory**

# Outline

- What are the top technical challenges in the area of your presentation topic?
- What are technology gaps that the roadmap did not cover?
- What are some of the high priority technology areas that NASA should pursue?
- Do the high priority areas align well with the NASA's expertise, capabilities, facilities and the nature of the NASA's role in developing the specified technology?
- In your opinion, how well is NASA's proposed technology development effort competitively placed?
- What specific technology can we call a "Game Changing Technology"?
- Is there a technology component near the tipping point? (Tipping point: large advance in technology readiness is possible with a relatively small additional investment.)
- In your opinion, what is the time horizon for the technology to be ready for insertion (5-30 years)?
- Provide a sense of value in terms of payoffs, risk, technical barriers and chance of success.

# Today, ORNL is the world's most powerful computing facility

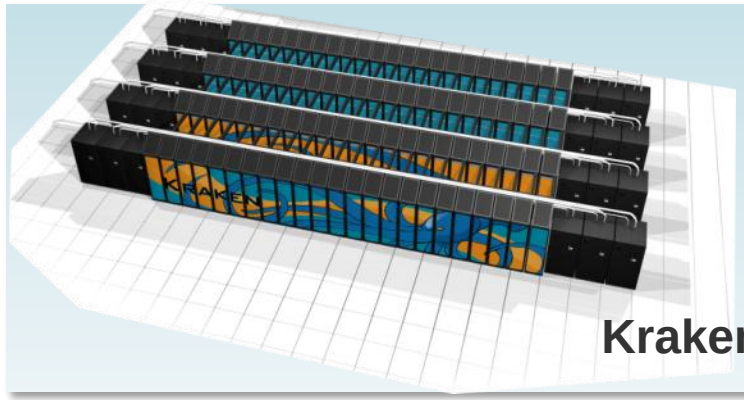



Jaguar

| | |
|---|---|
| Peak performance | 2.33 PF/s |
| Memory | 300 TB |
| Disk bandwidth | > 240 GB/s |
| Square feet | 5,000 |
| Power | 7 MW |

TOP500 SUPERCOMPUTER SITES #2

Dept. of Energy's most powerful computer

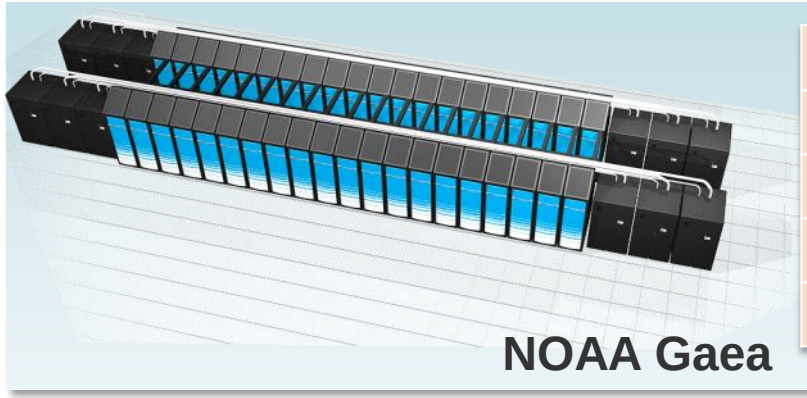



Kraken

| | |
|---|---|
| Peak performance | 1.03 PF/s |
| Memory | 132 TB |
| Disk bandwidth | > 50 GB/s |
| Square feet | 2,300 |
| Power | 3 MW |

TOP500 SUPERCOMPUTER SITES #8

National Science Foundation's most powerful computer



NOAA Gaea

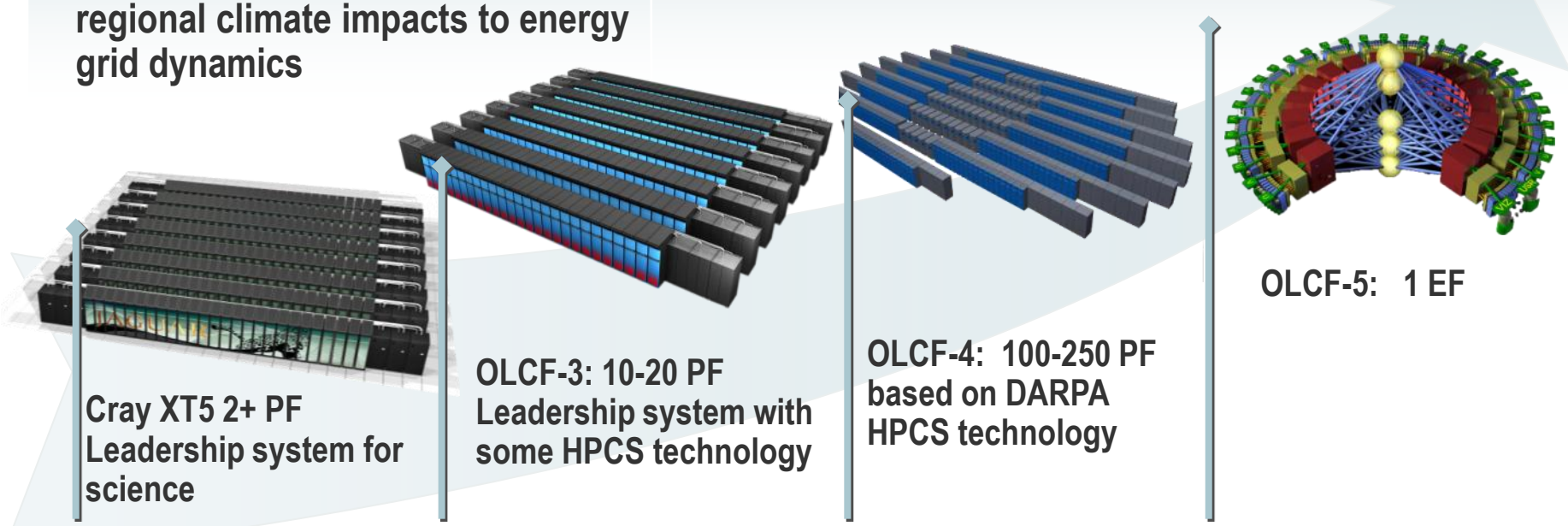
| | |
|---|---|
| Peak Performance | 1.1 PF/s |
| Memory | 248 TB |
| Disk Bandwidth | 104 GB/s |
| Square feet | 1,600 |
| Power | 2.2 MW |

TOP500 SUPERCOMPUTER SITES #32

National Oceanic and Atmospheric Administration's most powerful computer

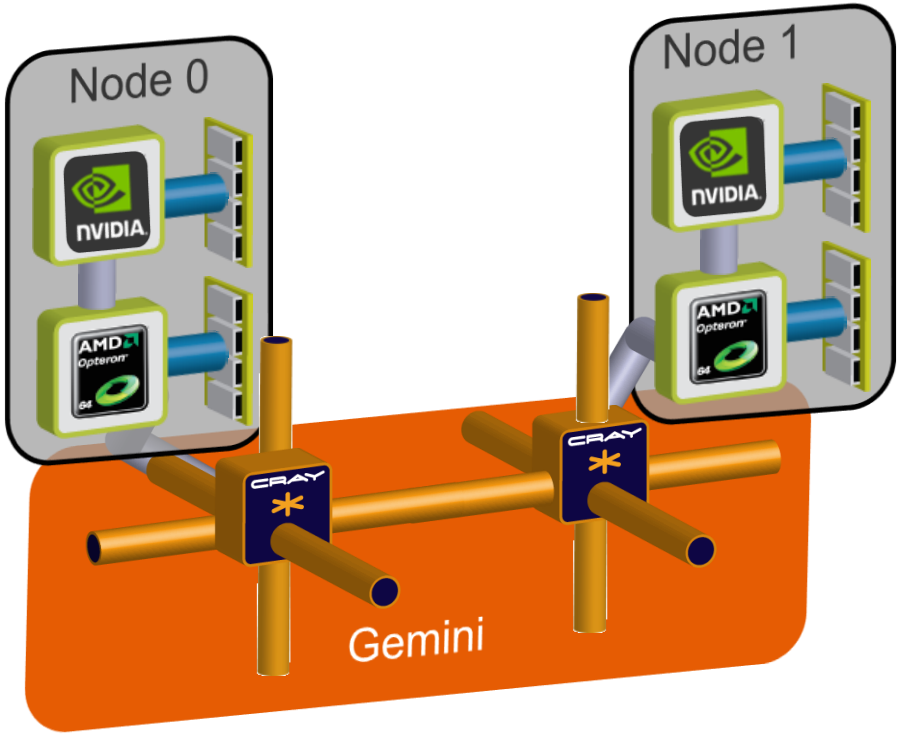
# Our science requires that we advance computational capability 1000x over the next decade

| Mission: Deploy and operate the computational resources required to tackle global challenges | Vision: Maximize scientific productivity and progress on the largest scale computational problems |
|---|---|
| • Deliver transforming discoveries in climate, materials, biology, energy technologies, etc.<br>• Ability to investigate otherwise inaccessible systems, from regional climate impacts to energy grid dynamics | • Providing world-class computational resources and specialized services for the most computationally intensive problems<br>• Providing stable hardware/software path of increasing scale to maximize productive applications development |



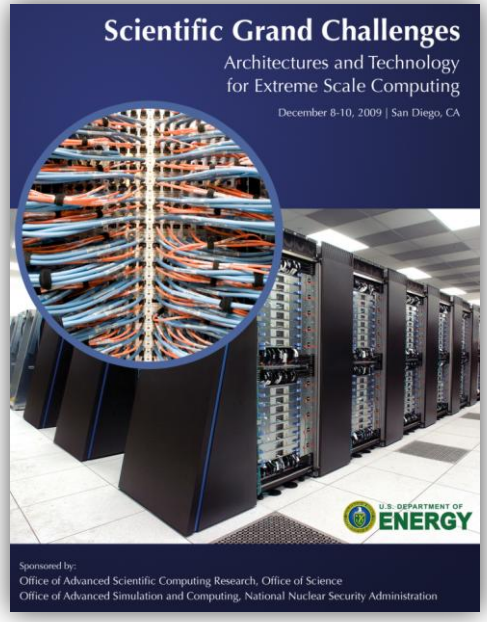
# OLCF-3 node description

- **New node for "Cray XE" infrastructure**
  - Gemini interconnect
  - AMD Socket G34 processor
- **1 AMD socket G34 processor and 1 NVIDIA GPU per node**
- **Interlagos uses AMD socket G34 and new "Bulldozer" core**
  - DDR3-1600 memory
  - HyperTransport version 3
- **NVIDIA "Kepler" accelerator**
  - Successor to Fermi




| | Jaguar's XT5 node | OLCF-3 node |
|---|---|---|
| Opteron sockets | 2 | 1 |
| Opteron memory (GB) | 16 | 32 |
| Interconnect | Seastar2 | Gemini |
| Node peak GFLOPS | 110 | >1500 |

# What will the exascale look like?

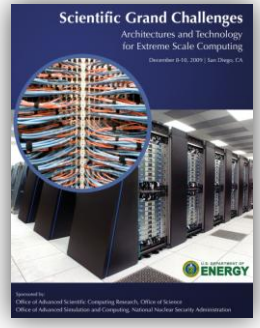



- "Node architectures are expected to change dramatically in the next decade, becoming more hierarchical and heterogeneous."
- ". . . computer companies are dramatically increasing on-chip parallelism to improve performance. The traditional doubling of clock speeds every 18 to 24 months is being replaced by a doubling of cores or other parallelism mechanisms."
- "Systems will consist of one hundred thousand to one million nodes and perhaps as many as a billion cores."

Architectures and Technology for Extreme Scale Computing, Workshop Report, 2009;
http://www.er.doe.gov/ascr/ProgramDocuments/Docs/Arch-TechGrandChallengesReport.pdf

| Systems | 2009 | 2015 +1/-0 | 2018 +1/-0 |
|---|---|---|---|
| **System peak** | **2 Peta** | **100-200 Peta** | **1 Exa** |
| **Power** | **6 MW** | **~15 MW** | **~20 MW** |
| System memory | 0.3 PB | 5 PB | 64 PB (+) |
| Node performance | 125 GF | 0.5 TF or 7 TF | 1,2 or 15TF |
| Node memory BW | 25 GB/s | 1-2TB/s | 2-4TB/s |
| Node concurrency | 12 | O(100) | O(1k) or 10k |
| Total Node Interconnect BW | 3.5 GB/s | 100-200 GB/s<br>10:1 vs memory bandwidth<br>2:1 alternative | 200-400GB/s<br>(1:4 or 1:8 from memory BW) |
| System size (nodes) | 18,700 | **50,000 or 500,000** | **O(100,000) or O(1M)** |
| Total concurrency | 225,000 | O(100,000,000) *O(10)-O(50) to hide latency | O(billion) * O(10) to O(100) for latency hiding |
| Storage | 15 PB | 150 PB | 500-1000 PB (>10x system memory is min) |
| IO | 0.2 TB | 10 TB/s | 60 TB/s (how long to drain the machine) |
| MTTI | days | O(1day) | O(0.1 day) |

# What does this say about the programming model?

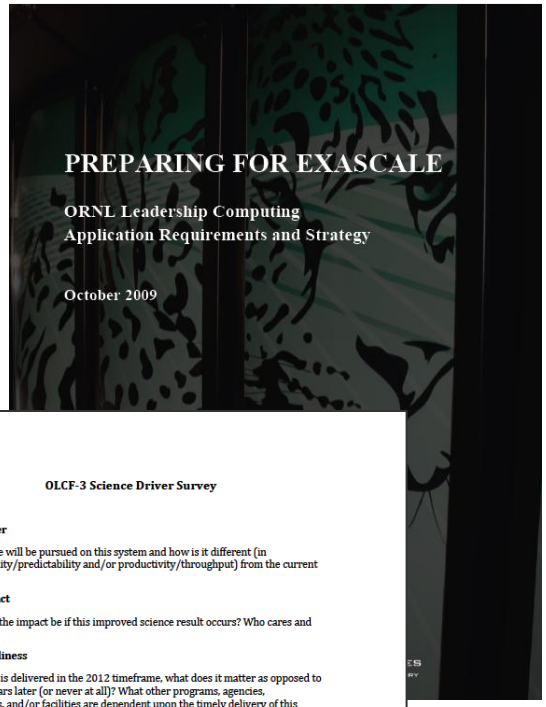


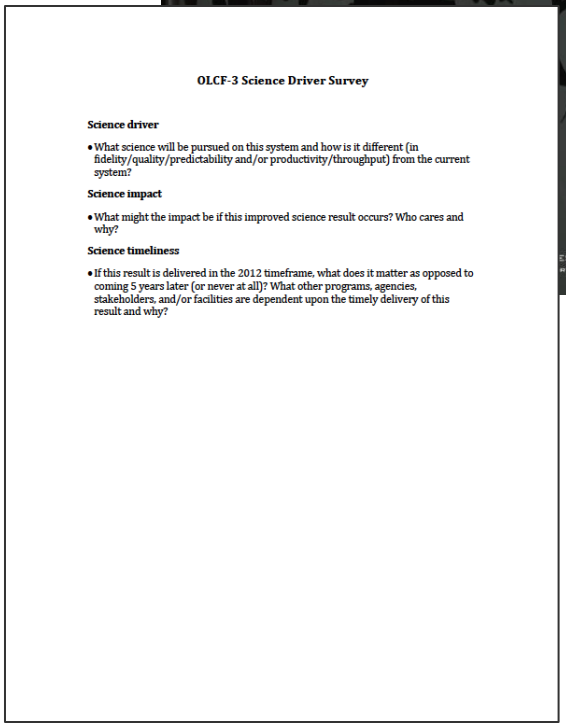

- **"The principal programming environment challenges will be on the exascale node: concurrency, hierarchy and heterogeneity."**
  - An "exascale node" will also be the workgroup/departmental-scale computing resource
- **". . . more than a billion-way parallelism to fully utilize an exascale system"**
- **"Portability will be a significant concern . . . In order to improve productivity a programming model that abstracts some of the architectural details from software developers is highly desirable."**

Architectures and Technology for Extreme Scale Computing, Workshop Report, 2009;
http://www.er.doe.gov/ascr/ProgramDocuments/Docs/Arch-TechGrandChallengesReport.pdf

# OLCF-3 Applications Analysis
## informed by two requirements surveys

- Project application requirements
  - Elicited, analyzed, and validated using a new comprehensive requirements questionnaire
  - Project overview, science motivation & impact, application models, algorithms, parallelization strategy, S/W, development process, SQA, V&V, usage workflow, performance
  - Results, analysis, and conclusions documented in 2009 OLCF application requirements document
- OLCF-3 baseline plan developed in consultation with 50+ leading scientists in many domains
  - What are the science goals and does OLCF-3 enable them?
  - What might the impact be if the improved science result occurs?
  - What does it matter if this result is delivered in the 2012 timeframe?



# PF Survey Findings

- **Algorithm development is evolutionary**
- **No algorithm "sweet spots"**
  - But algorithm footprints share characteristics
- **No one is clamoring for new languages**
- **MPI until the water gets too hot (frog analogy)**
- **Apps lifetimes are >3-5x machine lifetimes**
  - Refactoring is already a way of life
- **Fault tolerance via defensive checkpointing de facto standard**
  - Won't this eventually bite us? Artificially drives I/O demands
- **Weak or strong scale or both (no winner)**

# What kind of software infrastructure do we want?

- **inter-node layer is "straightforward"**
  - MPI, SHMEM, Global Arrays, Co-Array Fortran, maybe UPC
- **intra-node layer that allows us to easily move identified kernels to the accelerator**
  - it should be as facile as OpenMP
  - directive-based where accelerator regions are bounded
  - work with C/C++/Fortran
- **single compiler handles all aspects of the system intra-node architecture**
- **integrated libraries for BLAS/FFT/LAPACK**
- **Where do HPCS languages (e.g., Chapel) "sit"**
  - The original view might have have been at the inter-node layer
  - Incremental, evolutionary introduction might demand at the intra-node level

# What should the programming model look like?

1. MPI or Global Address Space languages across nodes
2. Within the very powerful nodes, use OpenMP, or other threads package to exploit the large number of cores
3. In each thread, use directives to invoke vector, SIMD, or SSE style instructions in the processor or accelerator to maximize performance
4. Explicitly manage data movement to minimize power
5. Describe the parallelism in the high-level language in a portable way, then let the compiler and libraries generate the best code for the architecture

**We are implementing this programming model for Titan, but this model works on current and future systems**

# Tools can enable more effective application development

- **pre-processing technology to manage complexity**
  - ROSE (http://rosecompiler.org/ )
- **performance hints, including opportunities for buffering**
- **frameworks that generate code**
  - MADNESS
  - Tensor Contraction Engine (TCE), http://www.csc.lsu.edu/~gb/TCE/
  - MAGMA (Atlas+ for GPUs)

- **build application-centric functionality into compiler/tools chain**
- **encapsulate appropriate prescribed tasks for accelerator work**
- **– similar to evolution of vectorizing or OpenMP compilers & technologies**
- **IPORT Scidac Institute proposed to build integrated, production-level, user-friendly refactoring toolchain from extant tools and new tools (PI's: R. Graham and B. Messer)**