Cray Architecture and Software Support for PGAS Languages

HPDC 2009 - München

June 12, 2009

Wilfried Oed

wko@cray.com

Cray Today

- **Nasdaq: CRAY**
  - Formed on April 1, 2000 as Cray Inc.
  - Headquartered in Seattle, WA
  - Roughly 850 employees across 30 countries
- **Four Major Development Sites**

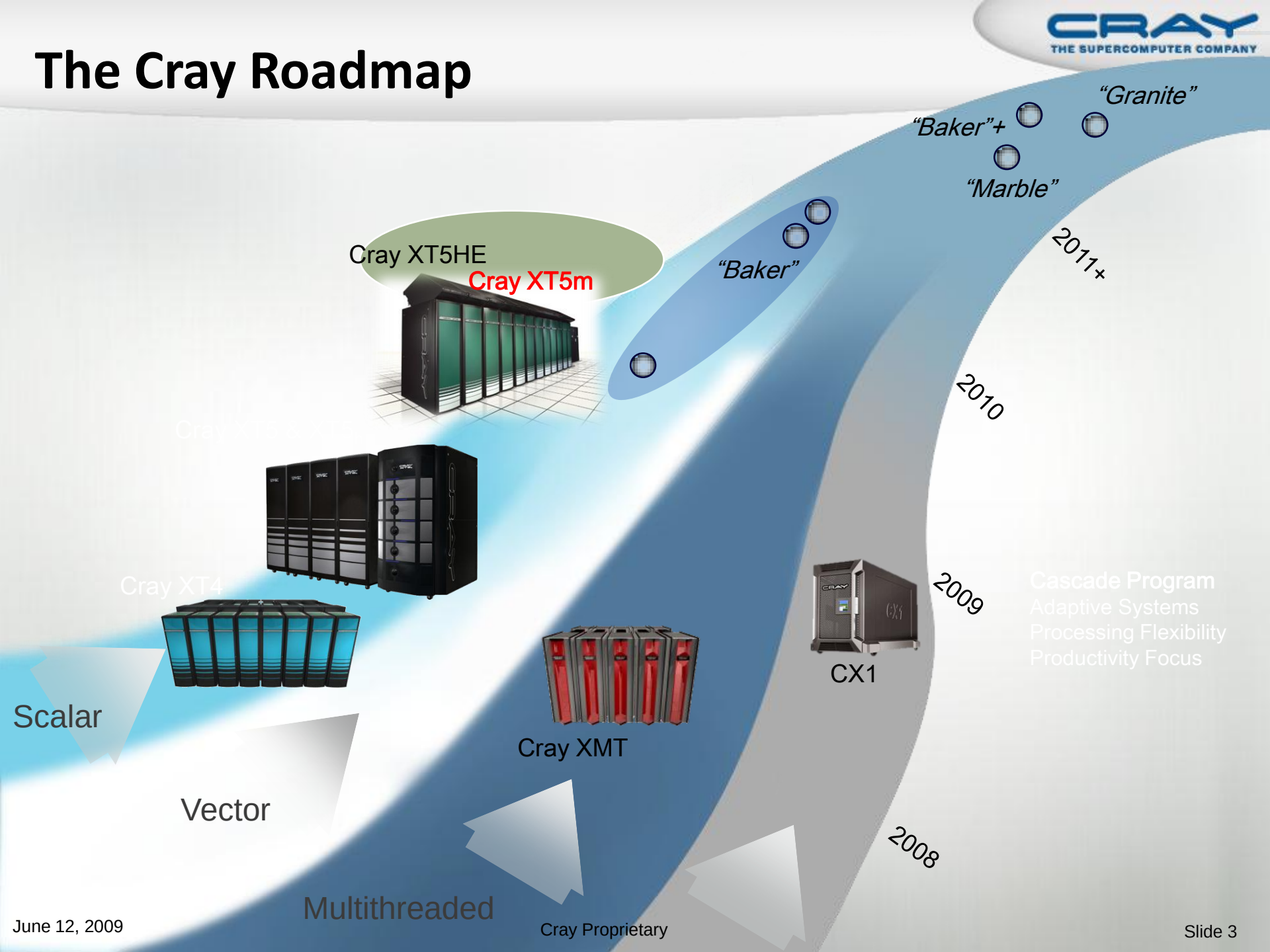
Seattle, Washington

Chippewa Falls, Wisconsin

Minneapolis, Minnesota

Austin, Texas

The Cray Roadmap



Cray XT Systems – Designed to Scale

JAIST
JAPAN ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY
1990

OAK RIDGE National Laboratory

THE UNIVERSITY OF WESTERN AUSTRALIA

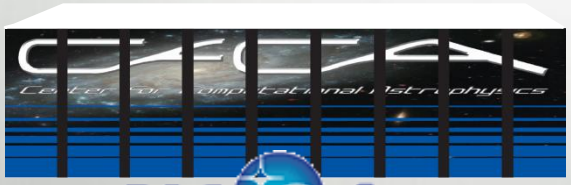
CSC

AWE

PITTSBURGH SUPERCOMPUTING CENTER

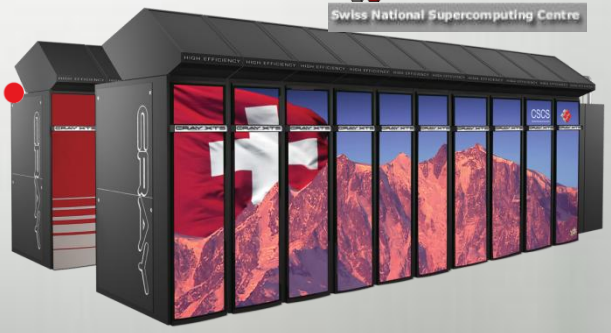
Sandia National Laboratories

CSCS
Swiss National Supercomputing Centre

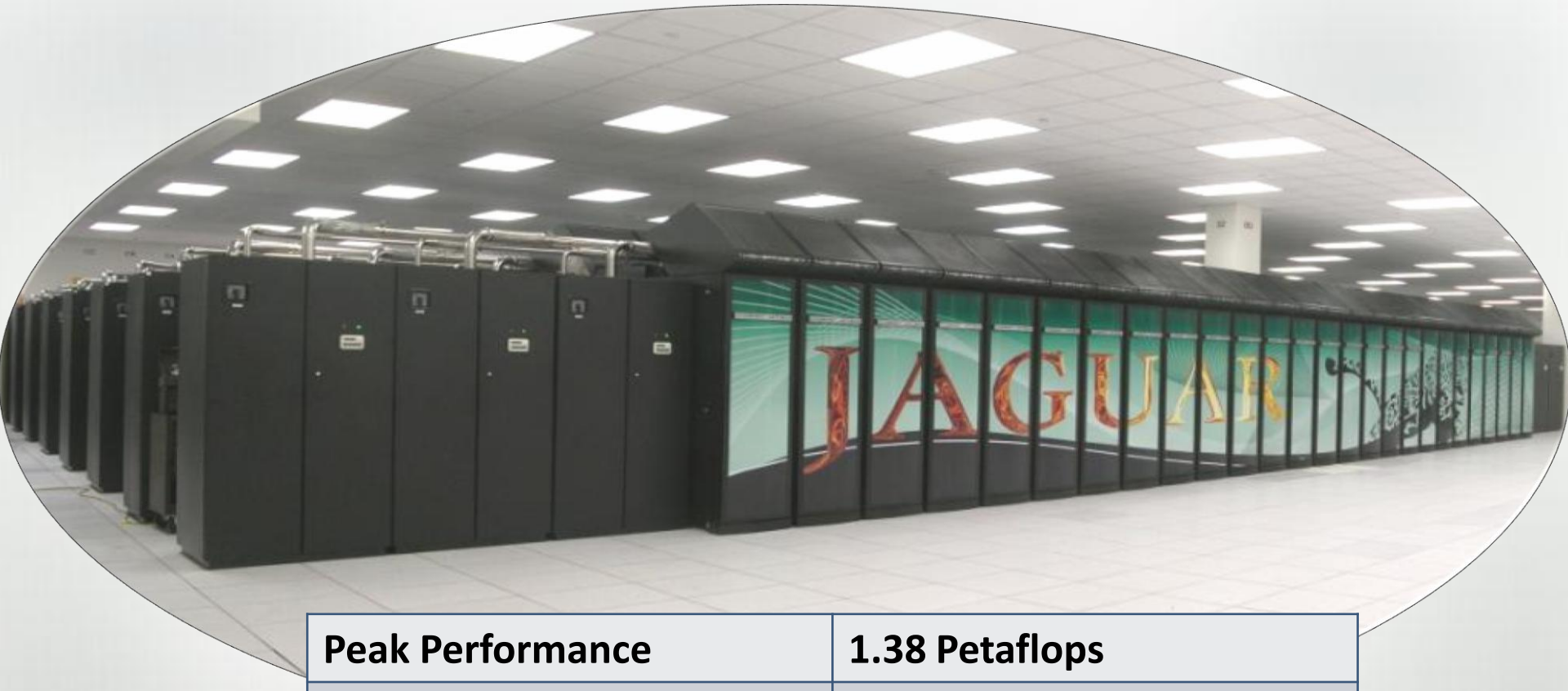


NAOJ
National Astronomical Observatory of Japan

JST Japan Science and Technology Agency



Jaguar: World's most powerful computer
Designed for science from the ground up



Peak Performance	**1.38 Petaflops**
System Memory	300 Terabytes
Disk Space	10.7 Petabytes
Disk Bandwidth	240+ Gigabytes/second
Processor Cores	150,000

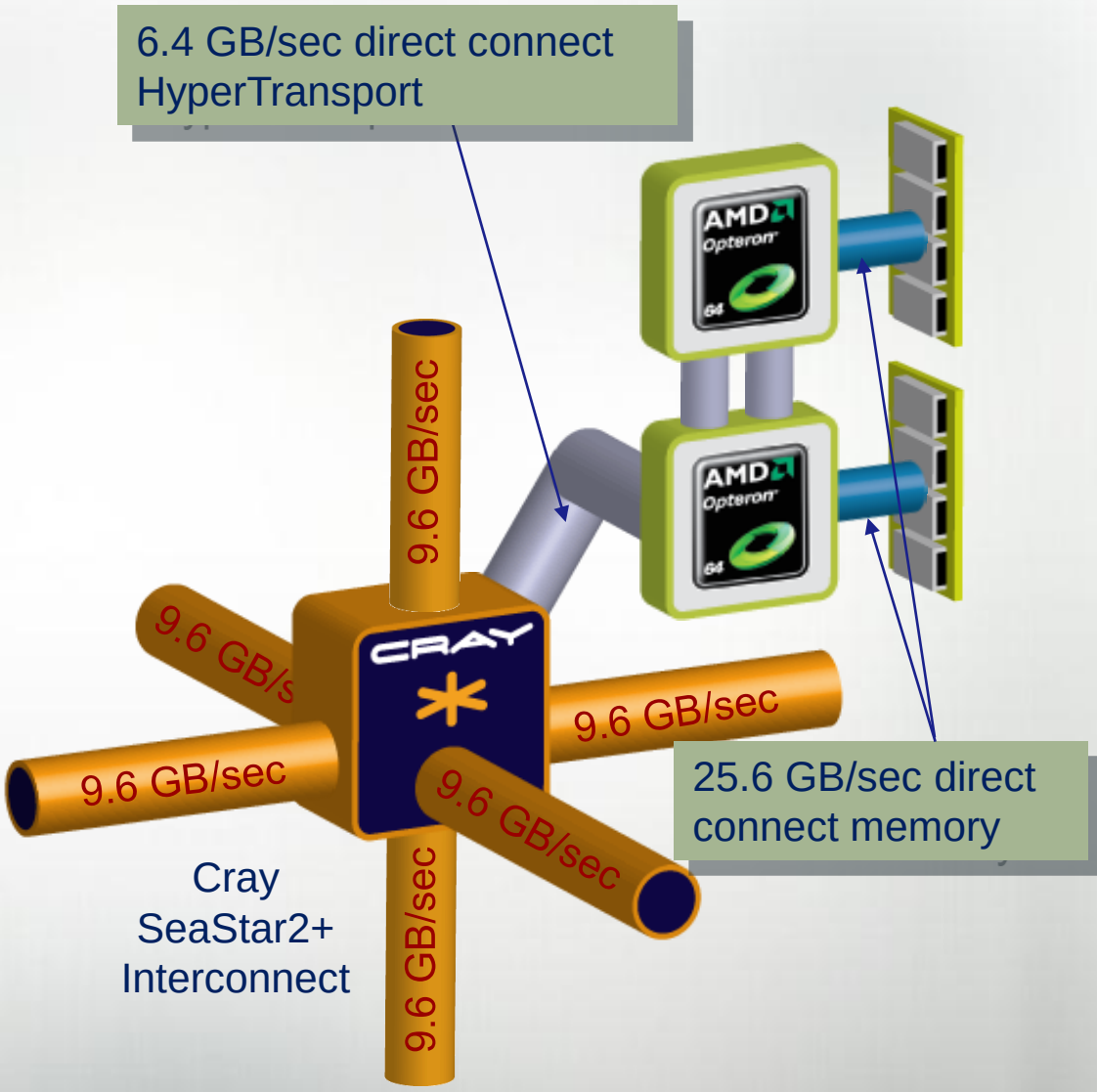
Early Science Applications

Science Area	Code	Contact	Cores	Total Perf	Notes	Scaling
Materials	DCA++	Schulthess	150,144	1.3 PF*	Gordon Bell Winner	Weak
Materials	LSMS/WL	ORNL	149,580	1.05 PF	64 bit	Weak
Seismology	SPECFEM3D	UCSD	149,784	165 TF	Gordon Bell Finalist	Weak
Weather	WRF	Michalakes	150,000	50 TF	Size of Data	Strong
Climate	POP	Jones	18,000	20 sim yrs/ CPU day	Size of Data	Strong
Combustion	S3D	Chen	144,000	83 TF		Weak
Fusion	GTC	UC Irvine	102,000	20 billion Particles / sec	Code Limit	Weak
Materials	LS3DF	Lin-Wang Wang	147,456	442 TF	Gordon Bell Winner	Weak

Cray XT5
Key Features

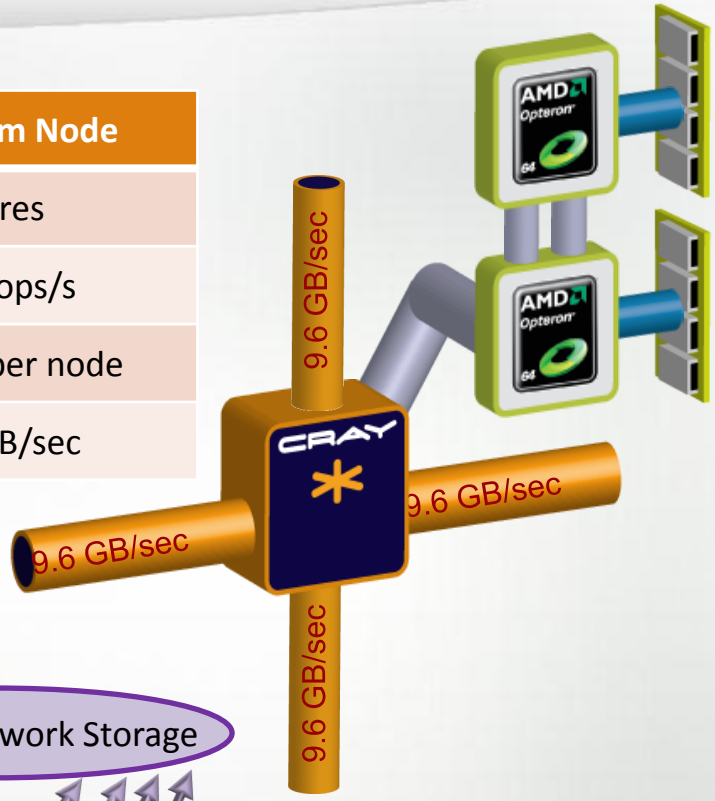
Cray XT5 Node

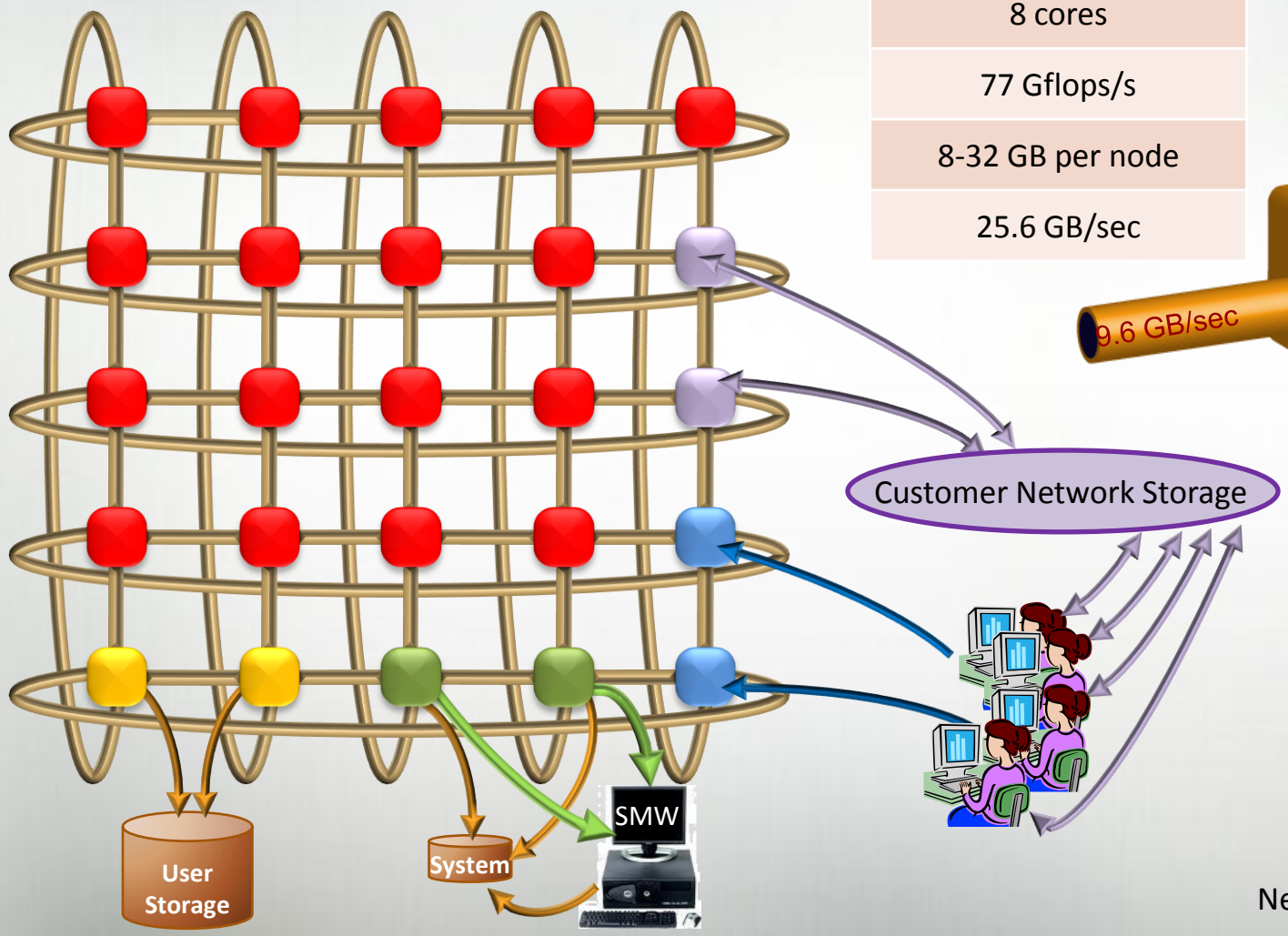
Characteristics	
Number of Cores	**8 or 12**
Peak Performance Shanghai (2.7)	**86 Gflops/sec**
Peak Performance Istanbul (2.6)	**124 Gflops/sec**
Memory Size	**8-32 GB per node**
Memory Bandwidth	**25.6 GB/sec**



Cray XT5m Architecture

Cray XT5m Node
8 cores
77 Gflops/s
8-32 GB per node
25.6 GB/sec





Compute Node

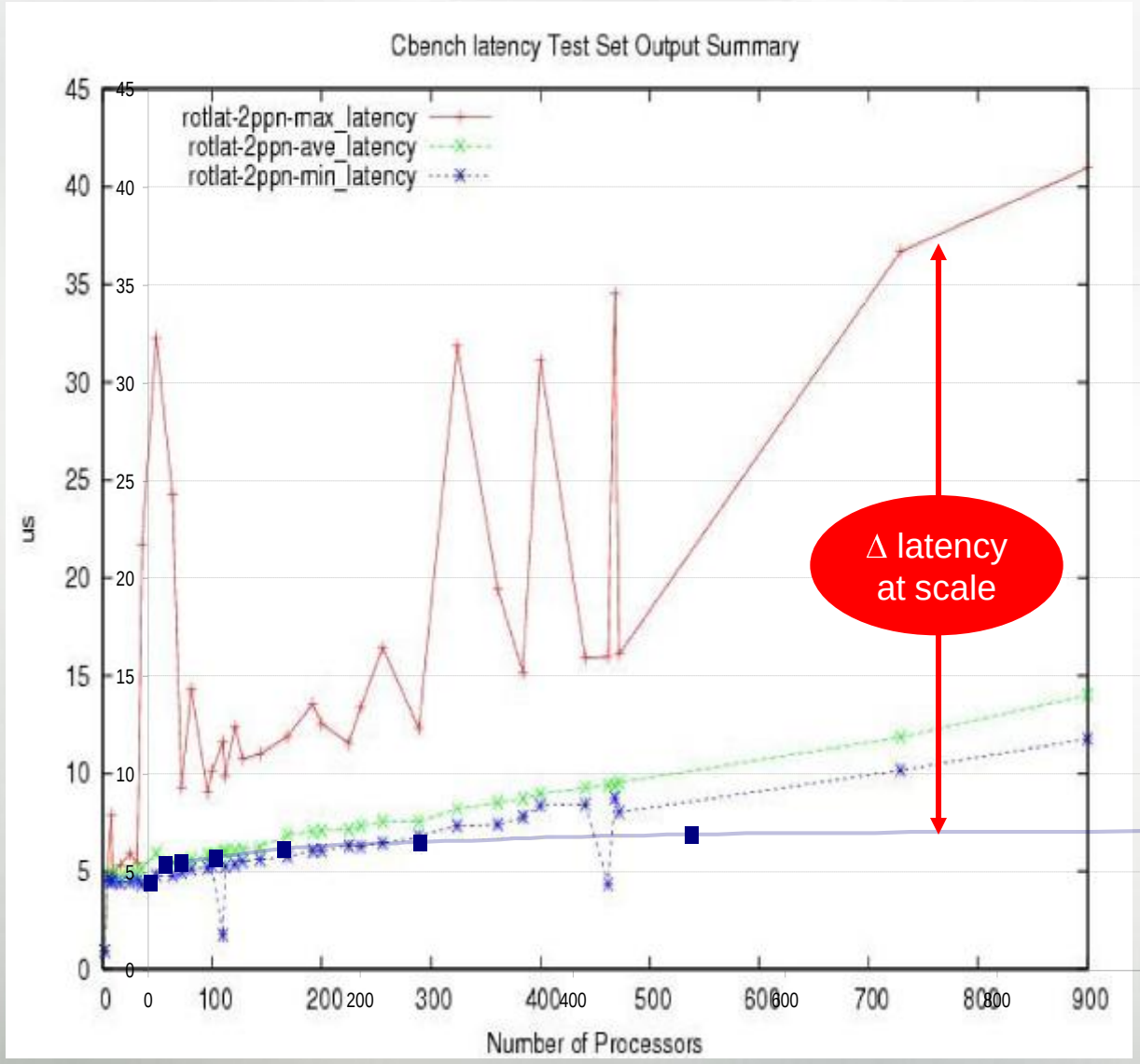
Login Node

System Node

I/O Node

Network Node (Optional)

Why Custom Interconnects ?
IB Cbench Latency



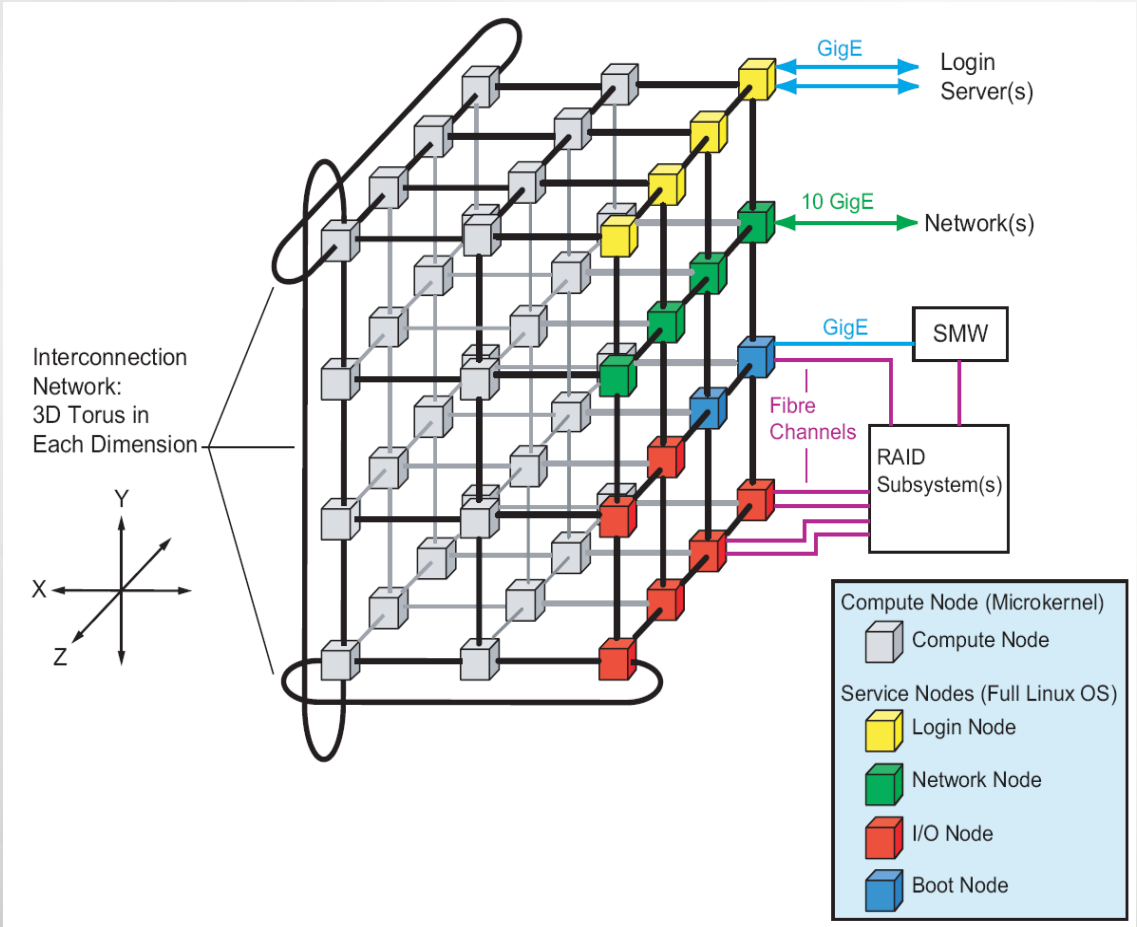
- IB shows a large spread best and worst case
- In MPP computing, we always wait for the slowest processor, so the *worst case figures are most important*
- Solutions include over-provisioning the interconnect and adaptive routing

Δ latency at scale

Cray SeaStar Maximum Latency

Source: Presentation by Matt Leininger & Mark Seager, OpenFabrics Developers Workshop, Sonoma, CA, April 30th, 2007

Scalable Software Architecture: CLE



- Lightweight kernel on Compute PEs, full featured Linux on Service PEs
- Contiguous memory layout used on compute processors to streamline communications
- Service PEs specialized by function
- Software Architecture eliminates OS "Jitter"
- Software Architecture enables reproducible run times
- Large machines boot in under 20 minutes, including filesystem
- Few seconds job launch time on 1000s of PEs

Partitioned Global Address Space (PGAS) Languages

PGAS programming

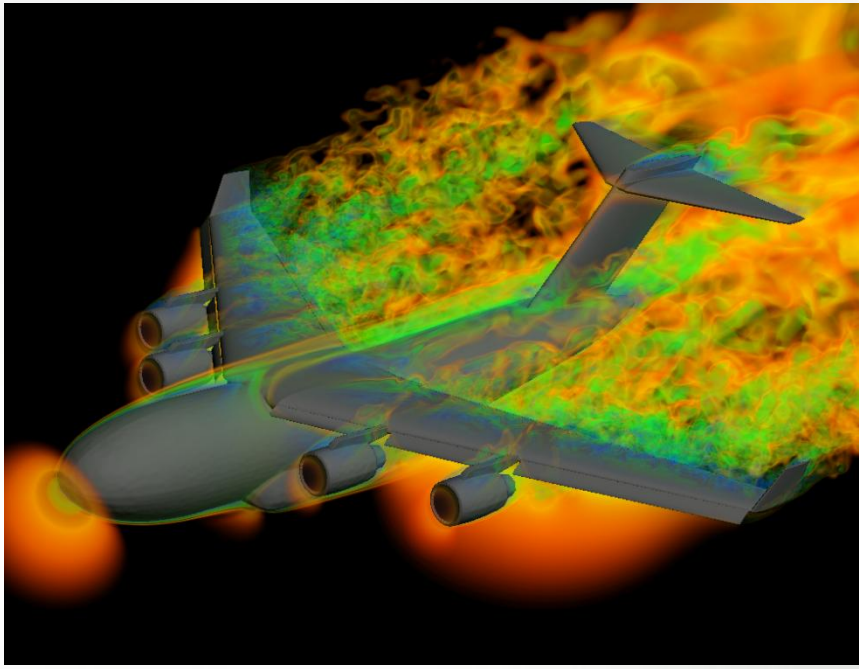
- **Productivity**
  - Global address space supports construction of complex shared data structures
  - High level constructs (e.g., multidimensional arrays) simplify programming
  - Language level parallelism as opposed to library calls
    - Extension to C – Unified Parallel C (UPC)
    - Extension to Fortran – Coarray Fortran (CAF)
  - Many algorithms are very naturally expressed using one-sided language level parallelism
    - Halo exchanges
    - Mesh manipulation and movement
- **Performance**
  - PGAS Languages are faster than two-sided MPI
  - Compilers can optimize parallel constructs
- **Portability**
  - These languages are nearly ubiquitous

PGAS and Cray

- Cray has been supporting Co-Array Fortran (CAF) and UPC since the beginning
  - Original support on the Cray T3E
  - Supported in software and hardware on the Cray X1/X2
  - Supported in software on the Cray XT
  - Cray XT hardware support in next generation "Gemini" interconnect
  - Can be mixed with MPI and OpenMP
- Full PGAS support on the Cray XT
  - Cray Compiling Environment (CCE) 7.0 – Dec 08
  - Full UPC 1.2 specification
  - Full CAF support – CAF proposed for the Fortran 2008 standard
  - Hybrid MPI/PGAS codes supported – very important!
- Fully integrated with the Cray software stack
  - Same compilers, job launch tools, libraries
  - Contrast with Berkeley UPC

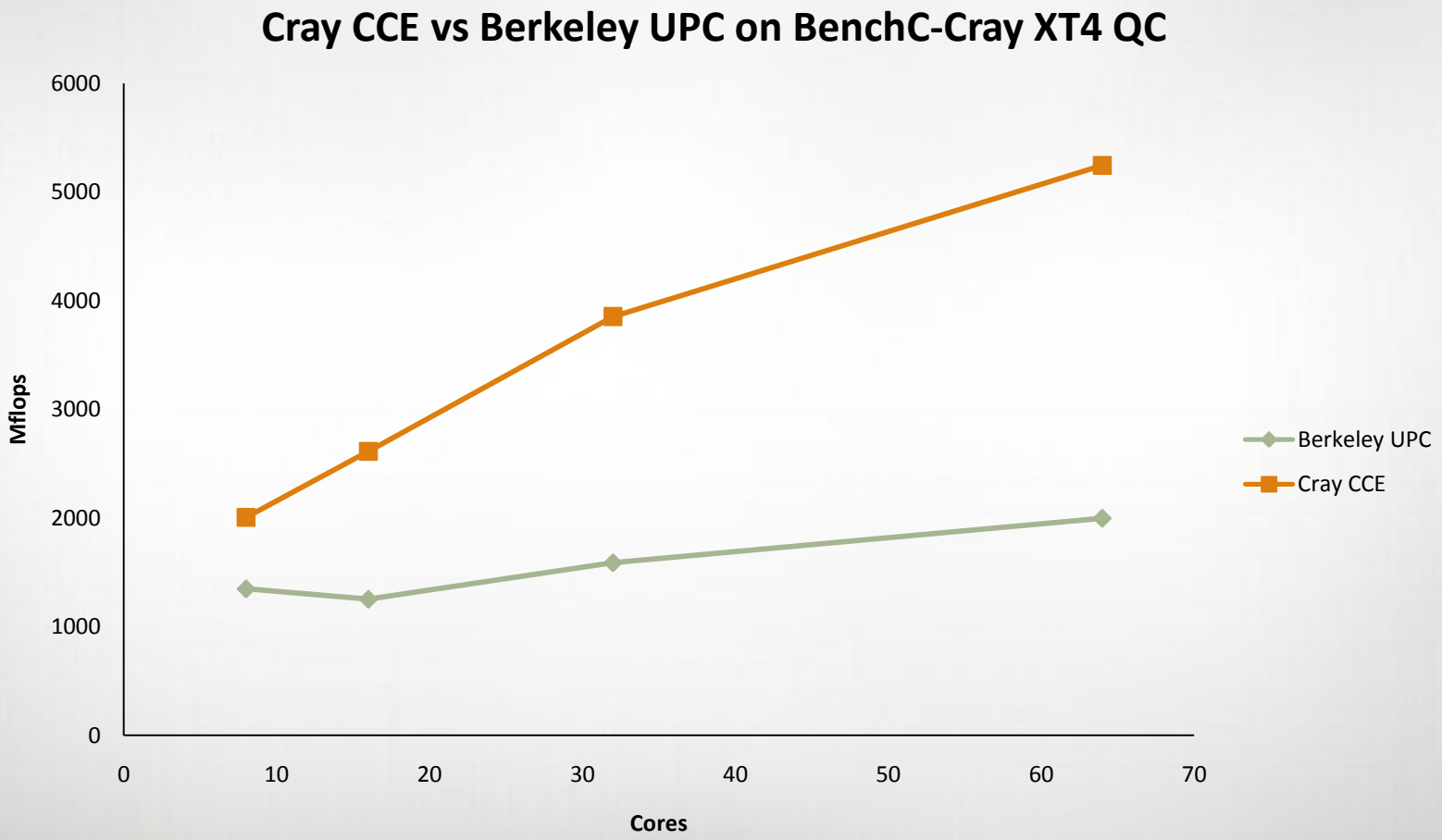
BenchC

- Unstructured finite element CFD solver
- Extensive and detailed timing routines
  - I/O
  - Problem setup and initialization
  - Communication routines
  - Matrix solver routines
  - “Number crunching” routines
- Extensively validated
- Developed at AHPCRC



- Communication in timestep loop can be either 100% MPI or 100% UPC
  - Compile-time option
  - UPC version still uses MPI during set-up stages (not tracked in performance measurements)
  - Original target was Cray X1/X2
  - Work ongoing to improve Cray XT performance

PGAS Performance on the Cray XT



PGAS performance on the Cray XT

- The SeaStar in the Cray XT is an MPI engine – designed with nonblocking MPI in mind
  - Does not support in hardware the single-sided comms used in PGAS
  - Next generation "Gemini" will support this in hardware (AMOs)!
  - Current Cray XT systems can be upgraded to Gemini!
- Cray Compiling Environment (CCE) 7.0 gives a reference implementation of UPC and CAF on the XT
  - CCE 7.1 emphasizes performance with UPC or CAF
- How does this compare to other implementations?
  - Only implementation we can test is Berkeley UPC
  - No CAF XT options other than CCE
- Berkeley UPC
  - Open source and cross-platform
  - Supports a portals conduit for the Cray XT

Thank You
For Your Attention !