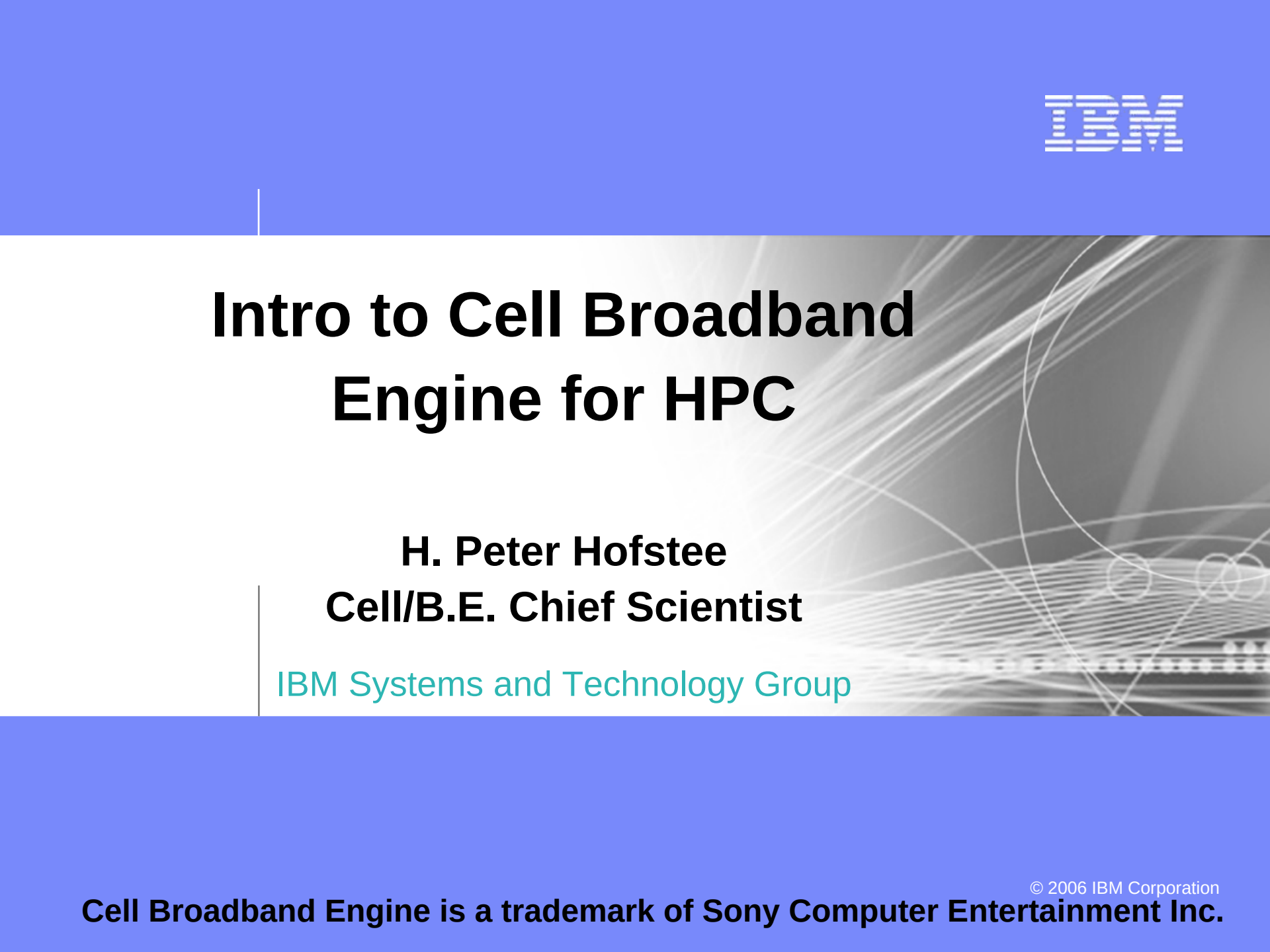


Intro to Cell Broadband Engine for HPC

H. Peter Hofstee
Cell/B.E. Chief Scientist

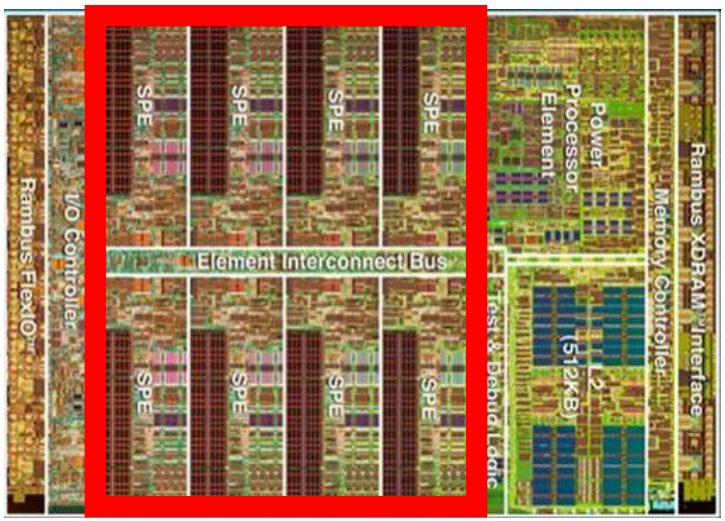
IBM Systems and Technology Group

Dealing with the Memory Wall in the Compute Node

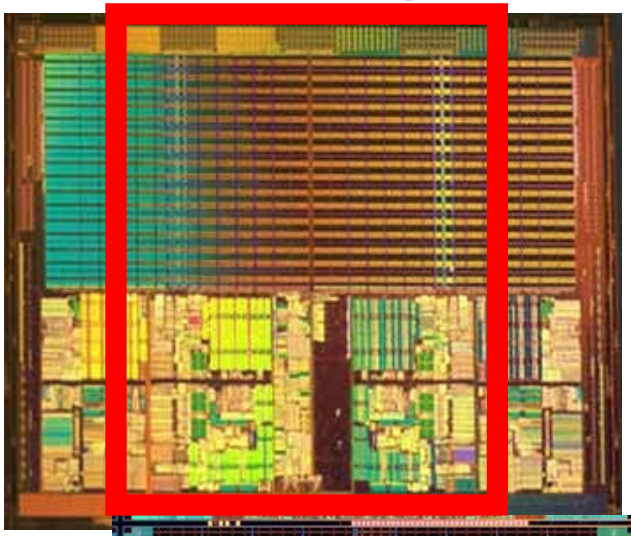
- Manage locality
  - Cell/B.E. does this explicitly, but applies to nearly every processor once you tune
- Go very (thread) parallel in the node
  - Many relatively slow threads so that memory appears closer
  - IBM BlueGene, Sun Niagra, CRAY XMT, …
- Prefetch
  - Generalization of long-vector (compute is the easy part)
  - Cell/B.E. (code and data), old-style CRAY
- All place a burden on programmers
  - Automatic caching has its limits
  - Auto-parallelization has its limits
  - Automatic pre-fetching / Deep auto-vectorization has its limits
- All have proven efficiency benefits
  - BGP and RoadRunner have about same GFlops/W
  - Both have significantly improved application efficiency on a variety of applications over clusters of conventional processors.

Memory Managing Processor vs. Traditional General Purpose Processor

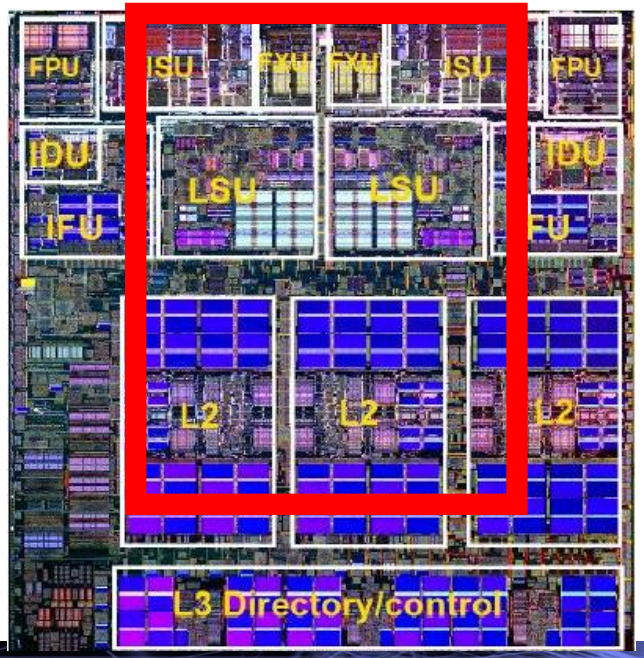
Cell BE



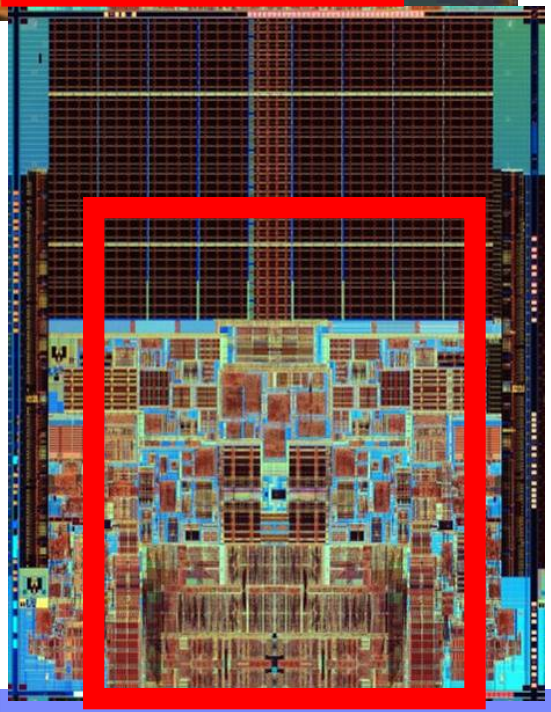
AMD



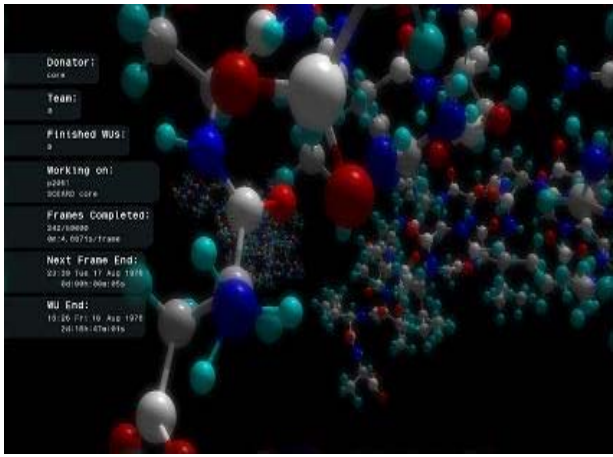
IBM



Intel



Folding@home distributed computing



Client statistics by OS

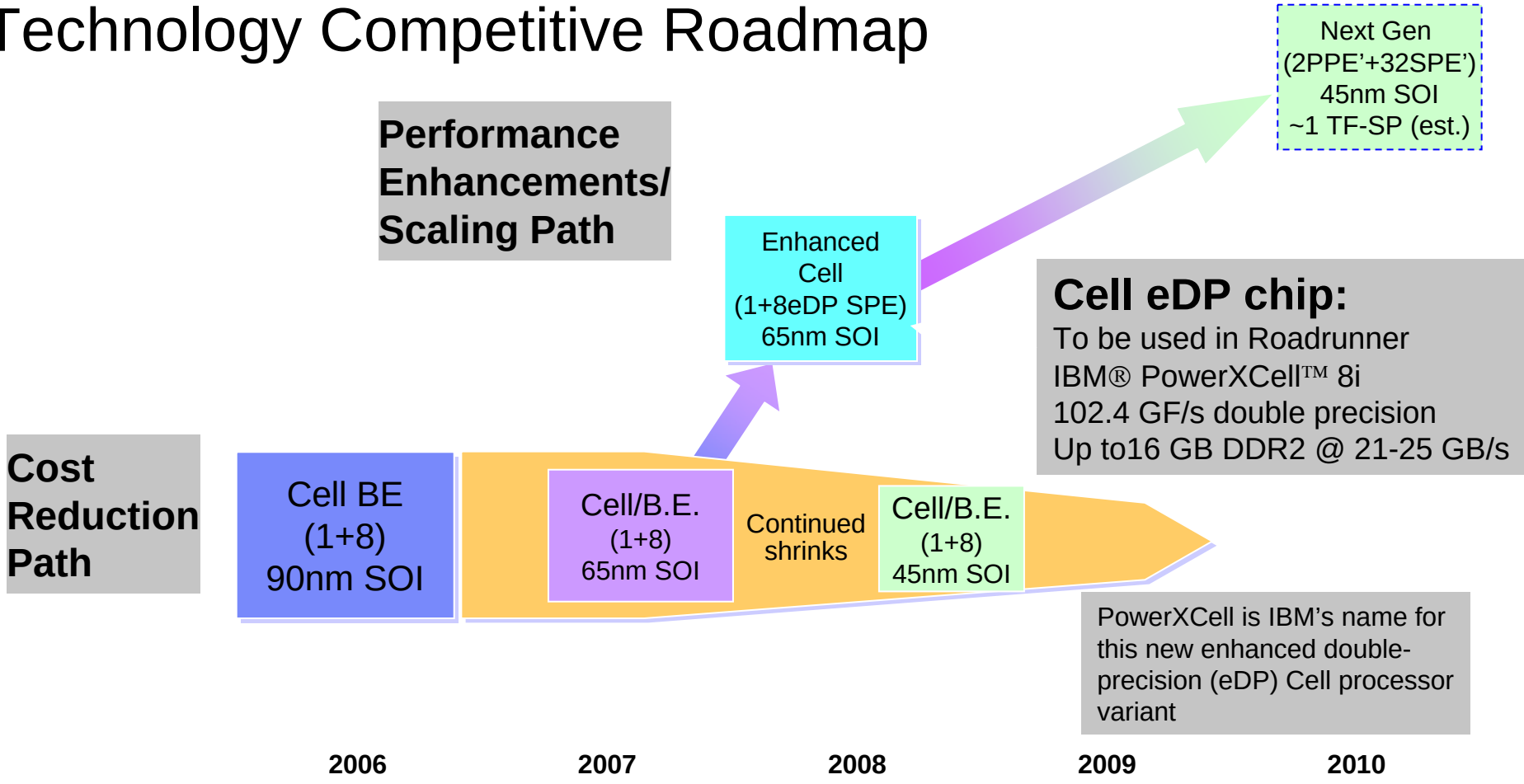
OS Type	Current TFLOPS*	Active CPUs	Total CPUs
Windows	164	172703	1798091
Mac OS X/PowerPC	8	9391	105647
Mac OS X/Intel	13	4188	23712
Linux	36	21449	245348
GPU	43	732	4246
PLAYSTATION®3	1000	40305	255338
Total	1264	248768	2432382

Total number of non-Anonymous donators = 808390

Last updated at Sun, 23 Sep 2007 09:49:22

DB date 2007-09-23 09:51:47

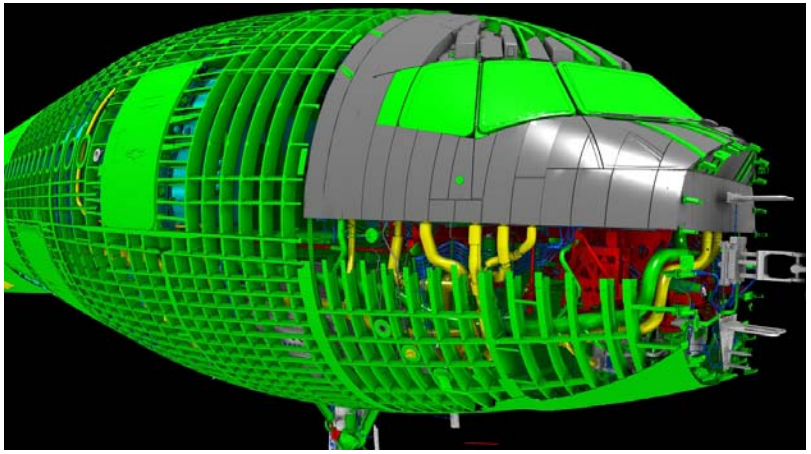
Cell Broadband Engine™ Architecture (CBEA) Technology Competitive Roadmap



All future dates and specifications are estimations only; Subject to change without notice.
Dashed outlines indicate concept designs.

Boeing 777 iRT Demo

- Hybrid Configuration
  - Ridgeback memory server (112GB memory)
  - QS21 rendering accelerators (6 Tflops, 14 blades)
- 350M Triangle model
  - 25GB working set
  - 23000x more complex than today's game models
- On demand transfers to blades
  - NFS RDMA over IB
- Real-time 1080p ray-traced output
- Compute Hierarchy
  - Head node load balancing blades
  - PPE load balancing SPEs

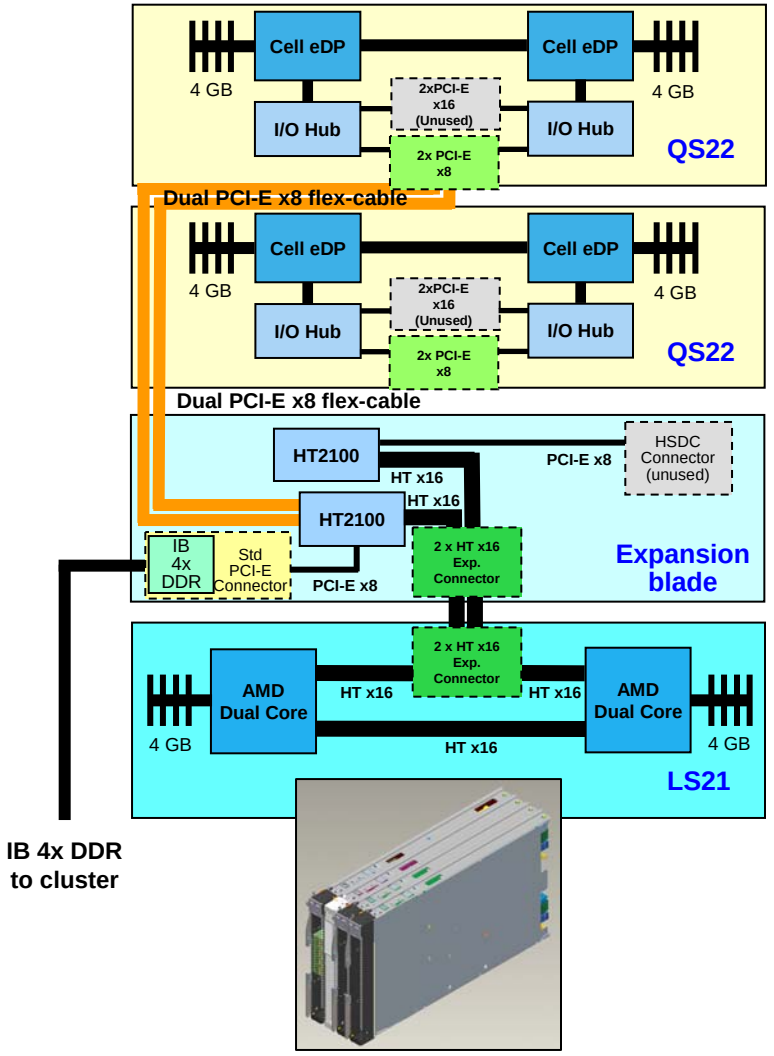


128GB	2GB	256KB	
(x86 disk) –>	(x86 memory) –>	(Cell memory) –>	(SPE local store) –> (SPE register file)
120MB/sec	2GB/sec	25GB/sec	50GB/sec

http://gametomorrow.com/blog/index.php/2007/11/09/cell-and-the-boeing-777-at-sc07/

A Roadrunner "Triblade" node integrates Cell and Opteron blades

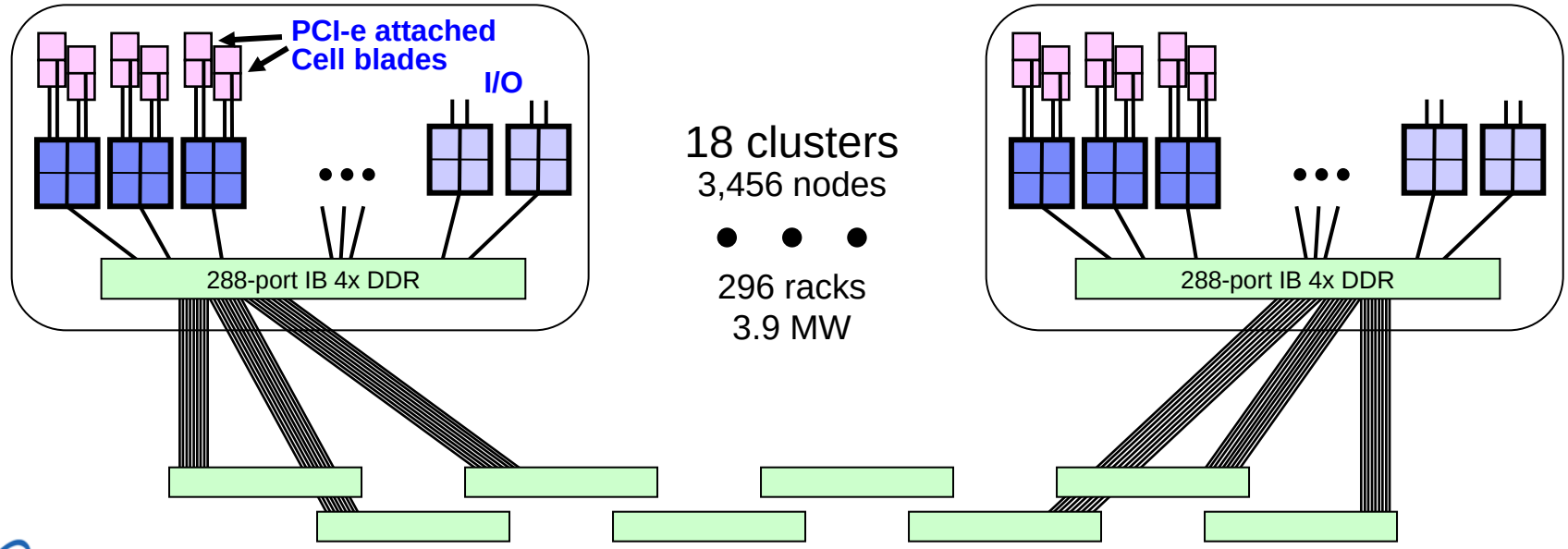
- QS22 is a future IBM Cell blade containing two new enhanced double-precision (eDP/PowerXCell™) Cell chips
- Expansion blade connects two QS22 via four internal PCI-E x8 links to LS21 and provides the node's ConnectX IB 4X DDR cluster attachment
- LS21 is an IBM dual-socket Opteron blade
- 4-wide IBM BladeCenter packaging
- Roadrunner Triblades are completely diskless and run from RAM disks with NFS & Panasas only to the LS21
- Node design points:
  - One Cell chip per Opteron core
  - ~400 GF/s double-precision & ~800 GF/s single-precision
  - 16 GB Cell memory & 8 GB Opteron memory



Roadrunner is a hybrid Cell-accelerated 1.4 PF system of modest size delivered in 2008

Connected Unit (CU) cluster
180 compute nodes w/ Cells
12 I/O nodes

12,960 Cell eDP chips ⇒ 1.3 PF, 52 TB
6,912 dual-core Opterons ⇒ 50 TF, 28 TB



Eight 2nd-stage 288-port IB 4X DDR switches
12 links per CU to each of 8 switches

Los Alamos NATIONAL LABORATORY EST.1943

ASC NNSA

Roadrunner Entry Level System
12 Hybrid Node Cluster

- Hybrid Compute Node
  - 24 - QS22s a future IBM Cell blade containing two new enhanced double-precision IBM® PowerXCell™8i processors
  - 12 - LS21 an IBM dual-socket Opteron blade
  - Conneced via four PCI-e x8 links
  - Includes a ConnectX IB 4X DDR cluster attachment
  - Compute node is diskless
- IBM x3655 I/O and Management Servers
- 4-wide IBM BladeCenter packaging
- 24 Port IB 4X DDR Switch & Fabric
- RHEL & Fedora Linux
- IBM SDK 3.0 for Multicore Acceleration
- IBM xCAT Cluster Management
  - System-wide GigEnet network
- Performance

	Host	Cell	Total
Peak (TF)	0.35	4.92	5.26
Memory (GB)	96	192	288
Ext IO (GB/s)	1.2		

Cell and hybrid speedup results are promising.

Application	***Type***	***Cell Only*** *(kernels)*		***Hybrid*** *(Opteron+Cell)*	
		CBE	*eDP*	*CBE+IB*	*eDP+PCIe*
SPaSM	full app	3x	4.5x	2.5x	>4x
VPIC	full app	9x	9x	6x	>7x
Milagro	full app	5x	6.5x	5x	>6x
Sweep3D	kernel	5x	9x	5x	>5x

- all comparisons are to a single Opteron core
- parallel behavior unaffected, as will be shown in the scaling results
- Cell / hybrid SPaSM implementation does twice the work of Opteron-only code
- Milagro Cell-only results are preliminary
- first 3 columns are measured, last column is projected

Courtesy John Turner, LANL

These results were achieved with a relatively modest level of effort.

Code	Class	Language	Lines of code		FY07 FTEs
			Orig.	Modified	
VPIC	full app	C/C++	8.5k	10%	2
SPaSM	full app	C	34k	20%	2
Milagro	full app	C++	110k	30%	2 x 1
Sweep3D	kernel	C	3.5k	50%	2 x 1

- all staff started with little or no knowledge of Cell / hybrid programming
- 2 x 1 denotes separate efforts of roughly 1 FTE each
- most efforts also added code

Courtesy John Turner, LANL

Where can we take Cell/B.E. next?

- Build bridges to facilitate code porting and code portability
  - E.g. compiler managed instruction and data caches
    - Target is competitive chip-level efficiency without Cell-specific software
    - Still allows full Cell benefit with optimized libraries and tuning
  - E.g. Multicore (and) Acceleration software development toolkit
    - Allow a wider audience to write parallel codes for a node
    - Porting across wide variety of systems
- Continue to enhance the Synergistic Processor Elements
  - Continue to increase application reach
  - Continue to measure ourselves on
    - application performance/W
    - application performance/mm2
- Integrate the equivalent of a RoadRunner node on a chip
  - Leverage Power 7 technology
  - Allows a 10PFlop system of reasonable size
  - Improved SPE – main core latency and bandwidth
  - Improved cross-system latencies