PACS-CS
A massively parallel cluster for computational sciences

Taisuke Boku

Center for Computational Sciences /
Graduate School of Systems and Information Engineering
University of Tsukuba

PACS-CS Project Overview

- Purpose
  - Creating a new CCS computational facility with tens of TFLOPS and Performing various target applications with large scale parallelism
  - Not just procuring a machine but developing unique one for CCS application requirements
- Period
  - Apr. 2005 – Mar. 2008 (FY 2005-2007)
- Team
  - Collaboration with computer scientists and computational scientists in CCS
  - Collaboration with university and vendor: U. Tsukuba, Hitachi and Fujitsu

HPC resources required at CCS

- Target applications and computation characteristics
  - full QCD （small size of complex matrix calculation + nearest neighboring communication)
  - Nano material science （CG method）
  - Astrophysics （hybrid computing for particles & field）
  - Environment, Biology （parameter search）
- Shift from MPP
  - Previous machine: CP-PACS with 2048 CPU & 614 Gflops
  - We need replacement of MPP with Commodity Technology
  - Main target applications require the *bandwidth both on memory and network*

Previous main resource: CP-PACS

#1 in TOP500 on Nov.'96
Peak perf. 614 GFLOPS
Linpack: 368 GFLOPS

Dropped off from TOP500 list on Nov. '03

Operation stopped on Sep. '05

Concept of PACS-CS

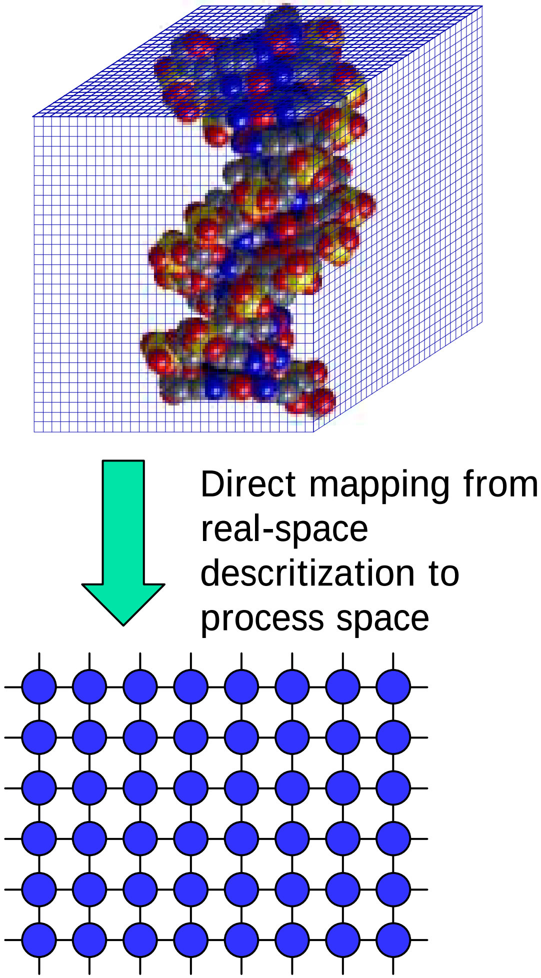
- Making an MPP-like system based on commodity technology
- Cluster is OK, but we keep the balance among
  CPU : memory : network performances
- To reduce the cost, considering our target applications

↓

- **PACS-CS**
  Parallel Array Computer System for Computational Sciences

Real Space Approach in Computational Science

- Real Space modeling
  - MPP code based on real-space descritization
    ⇒ Large scale parallelism with
    NNM comm. + collective comm.
  - Reducing Comp. Time (not reducing Comp. Amount)
    - Traditional methods：FFT, spectrum, …
      ⇒Indirect computation, not enlarging model space
    - Real Space method：
      ⇒Simulating real model on real space
  - We need re-program the traditional codes



General HPC clusters (as in yr. 2004)

- Intel-compatible CPU (Xeon, Opteron, Itanium2, ...)
- Dual CPU SMP
  - To reduce the space and the number of network interface keeping total system peak performance
  - Rack of memory bandwidth (memory wall problem) (but very fast for Linpack !)
  - Low sustained performance on network bound applications
- So much CPU frequency
  - For very high peak performance
- SAN (System Area Network)
  - MyrinetXP: dual connection for 500MB/s -> 10Gbps
  - Infiniband: x4 spec. for 1GB/s
  - Gb Ethernet is still OK for non-network bound applications (10GbE will come soon, but still expensive)

CPU / Memory performance balance

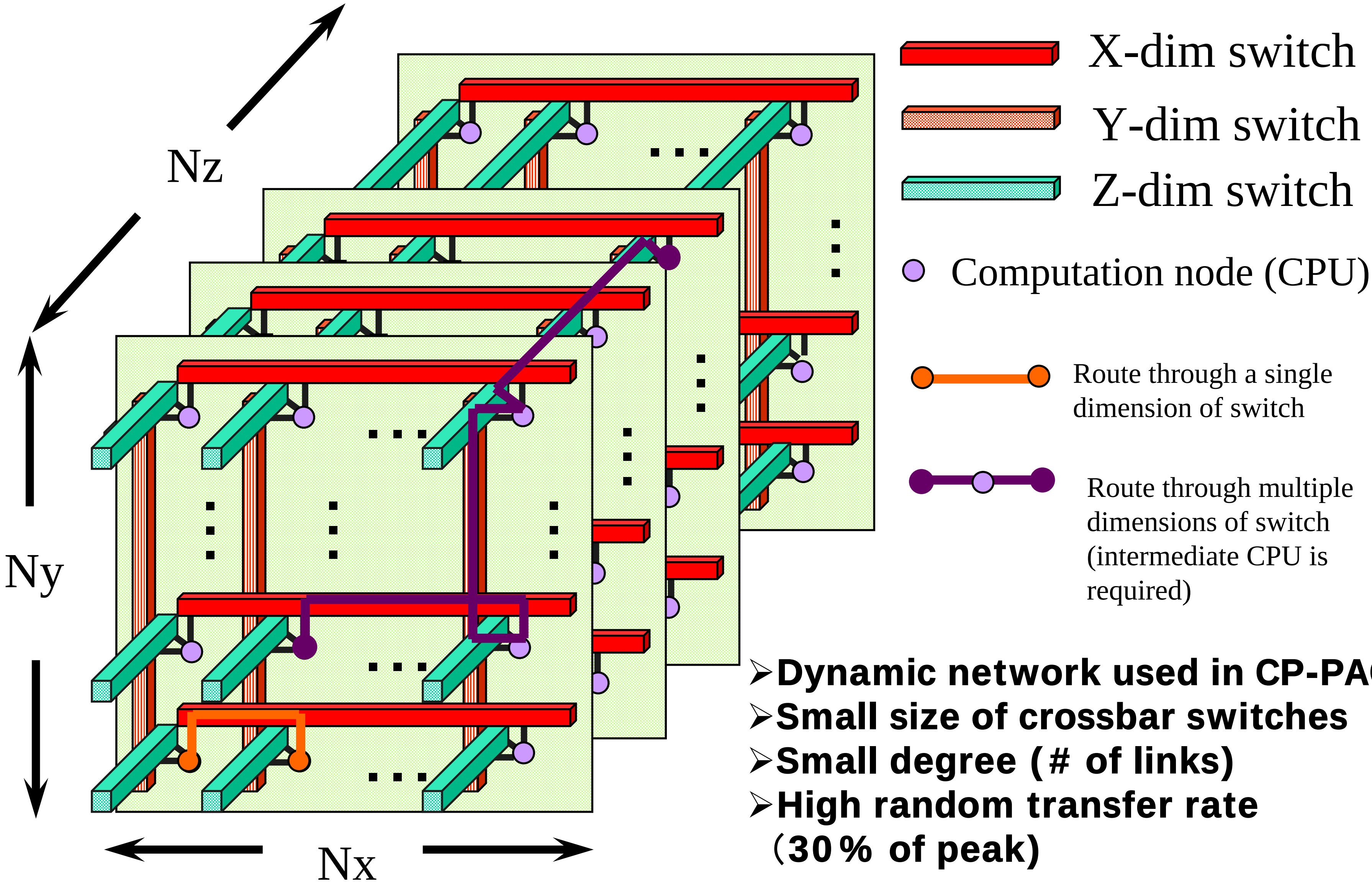
- Single CPU / node (not SMP)
- Memory wall problem ⇒ appropriate CPU speed (not too high frequency)
  - No chasing for the peak speed and frequency to CPU
  - We don't make "Linpack-aware machine"
- High density implementation
  - Same as traditional HPC clusters in SMP: 2CPU/1U
  - Keeping the same total memory size per chassis (1U)

Balance between node and network performance

- Our target application's parallelizing models
  - Nearest Neighboring Communication in n-dimension
  - Collective Communication (Broadcast/Reduction)
  - Not much random traffic
- Using commodity technology (= Ethernet)
  - Trunked links make wider bandwidth
  - Relatively large message size (non latency-sensitive)
  - Multi-dimensional implementation for much more bandwidth
  - No I/O-bus (PCI) bottleneck

➡ Hyper-Crossbar Network with trunked GbE

3-dimensional Hyper-Crossbar (3D-HXB) Network



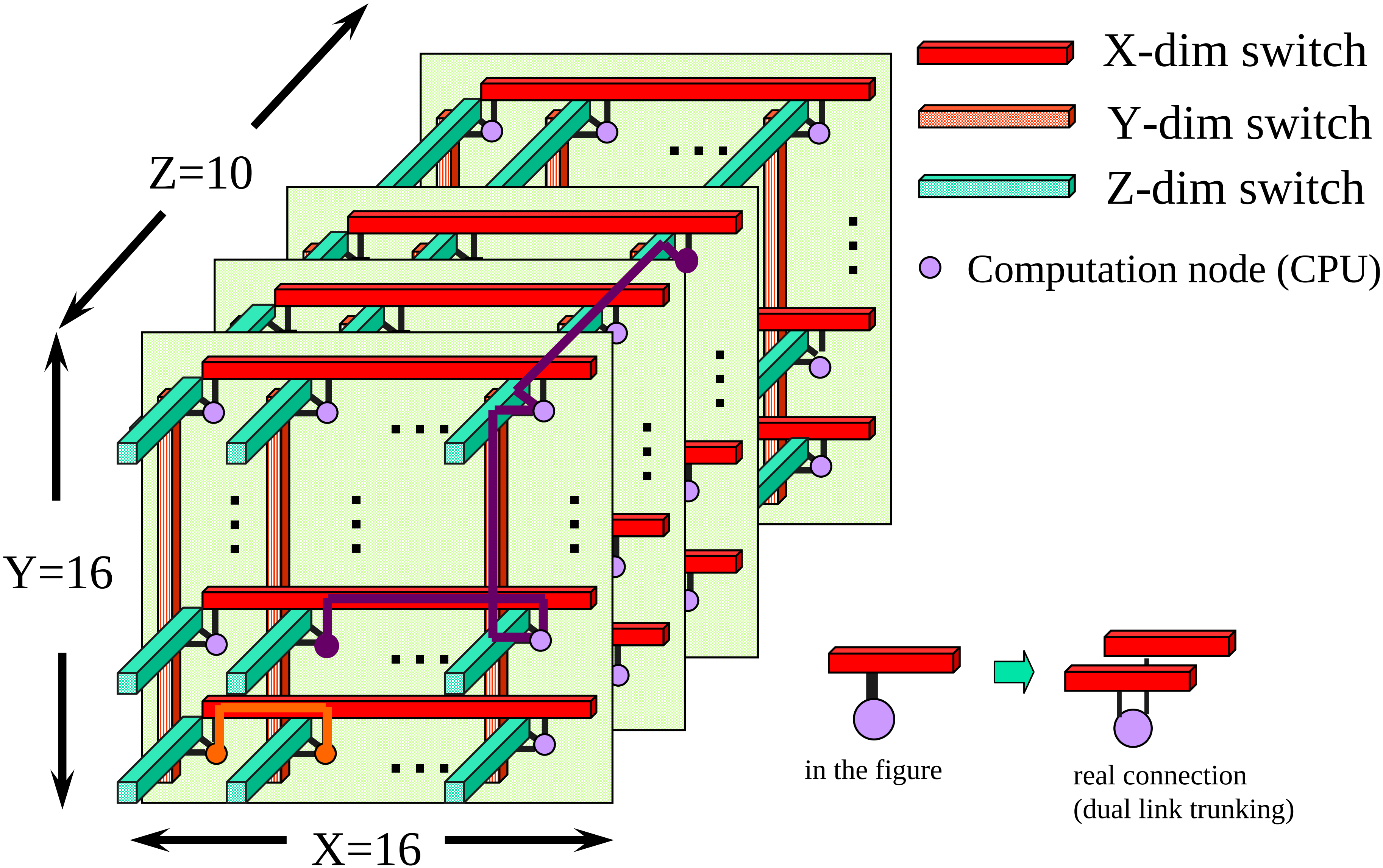
- **Dynamic network used in CP-PACS**
- **Small size of crossbar switches**
- **Small degree (# of links)**
- **High random transfer rate (30% of peak)**

HXB based on trunked GbE

- Effective use of low price commodity technology
  - NIC chip
  - Switch（L2, 10～20 ports）
- Network bandwidth
  - A few links are trunked ⇒ Software solution is enough
  - Simultaneous transfer on multiple dimensions
    bandwidth ＝（link bandwidth）×(# of trunks) × (# of dim.)
  - Distributing I/O load from single I/O point (ex. PCI) to multiple ones
- Routing is required for transfer on multiple dimensions
  - 3-D configuration is enough for up to 4096 nodes (16 CPUs / dimension)
  - Software trunking + routing

Logical connection among nodes (CPUs)

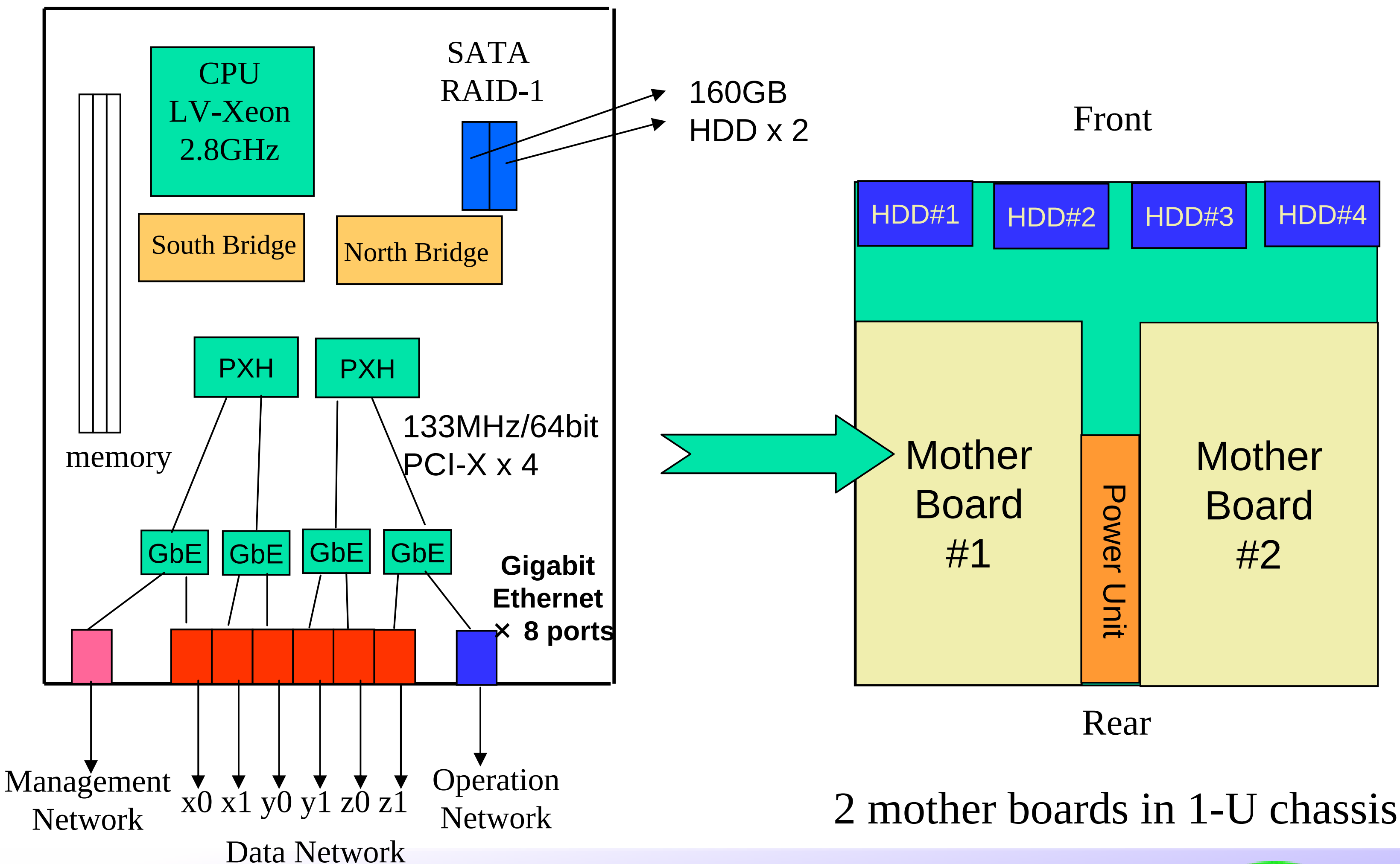
(3-D HXB network) 2560 nodes



Special network driver for GbE-trunk-HXB

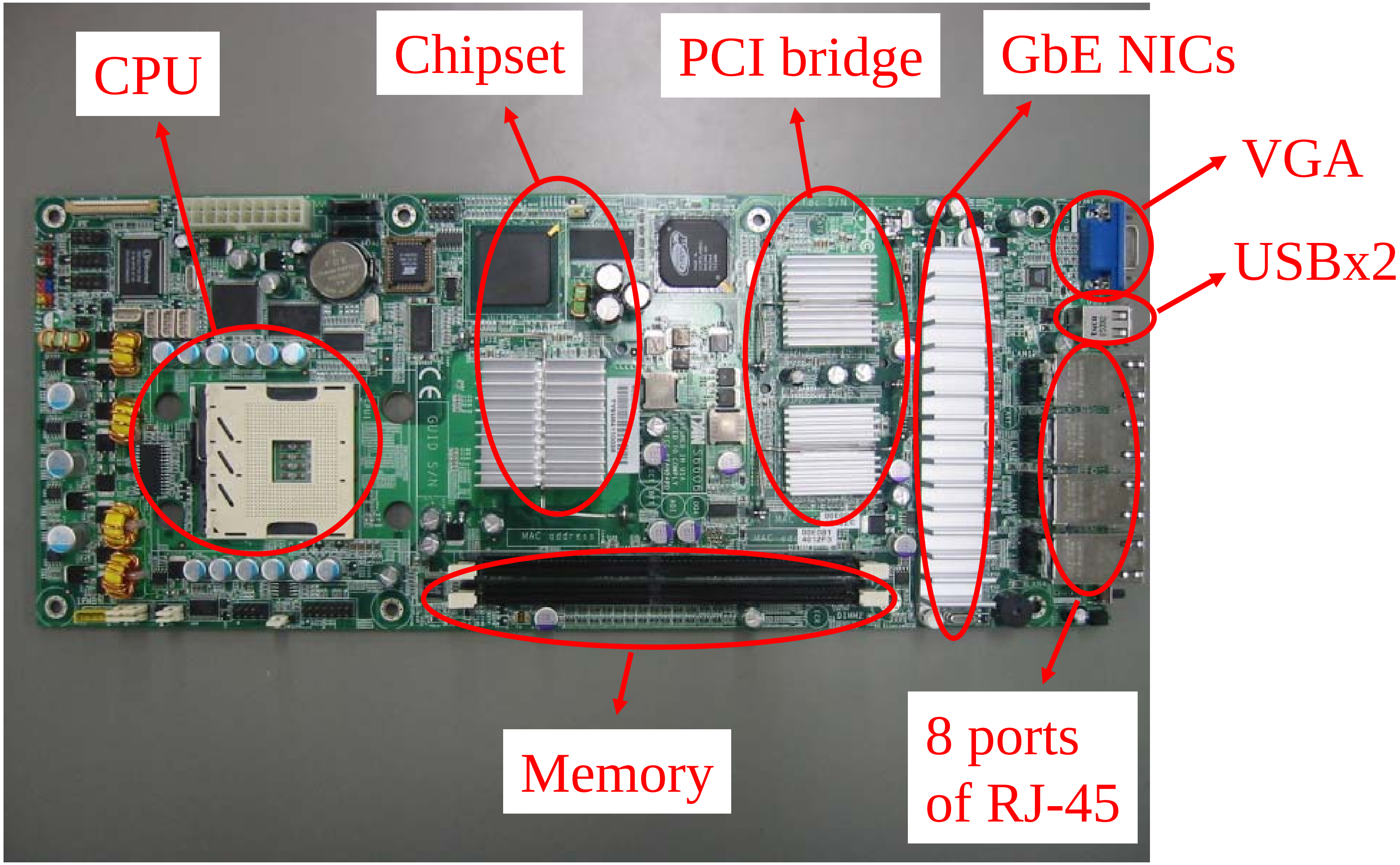
- PM-Ethernet/HXB enables
  - Direct inter-node communication on single dimension
  - Multiple GbE links are trunked to multiply bandwidth
  - Up to 3-D simultaneous sending/receiving
    - 250 MByte/sec (dual-link GbE) x 3 = 750 MByte/sec
  - Routing for a message requiring 1 or 2 hops of transfer on intermediate nodes
  - Fault tolerant operation for single link failure (future plan)
  - For more information ⇒ another paper on ICS'06

Mother board & Chasis

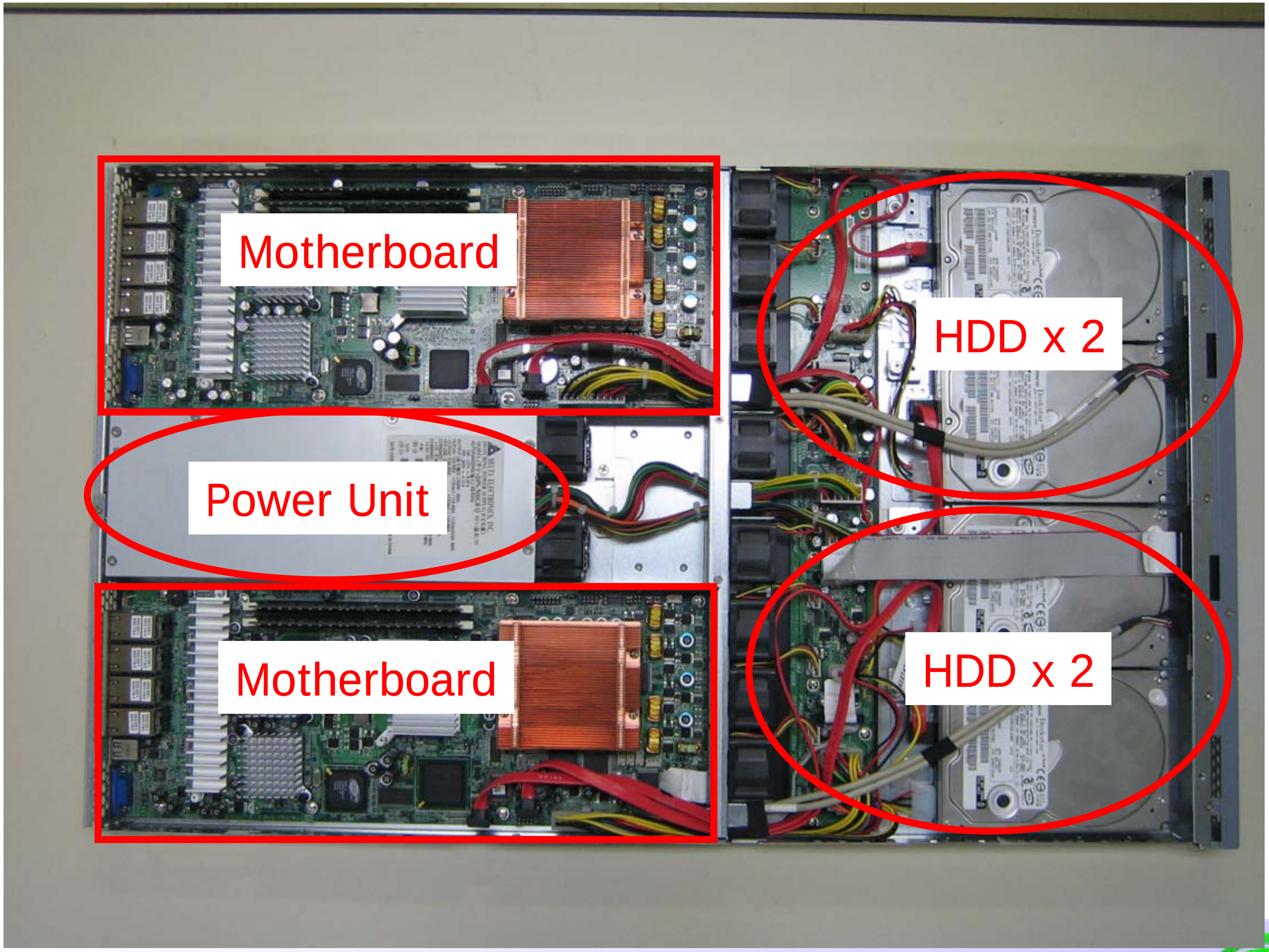


2 mother boards in 1-U chassis

Specially designed motherboard



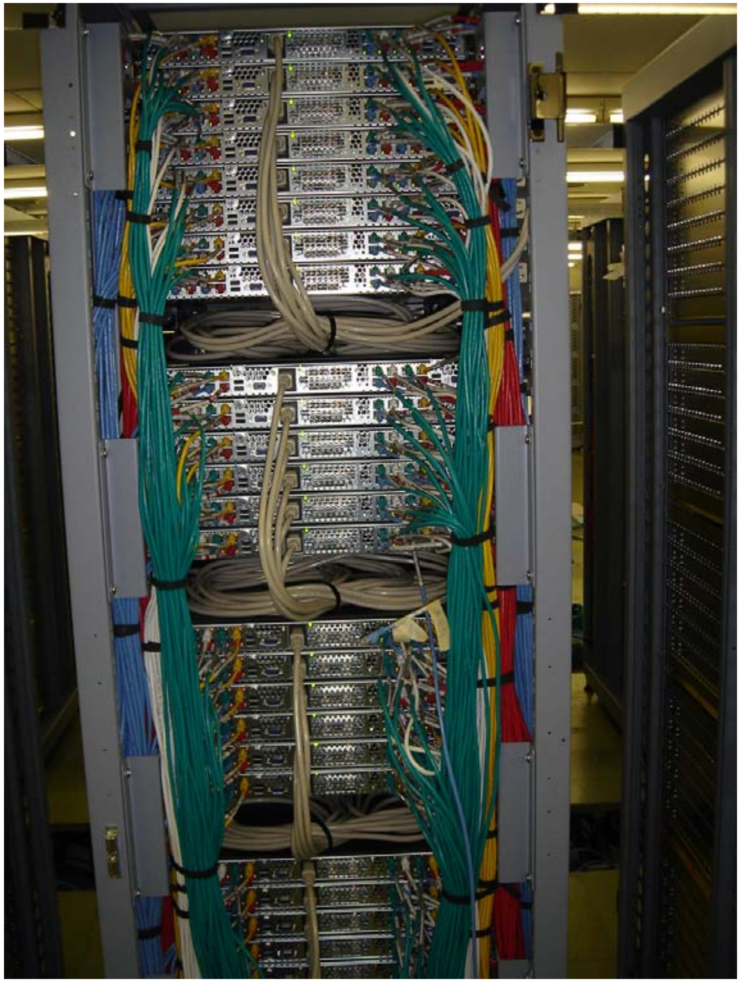
Unit chassis (19inch x 1U)



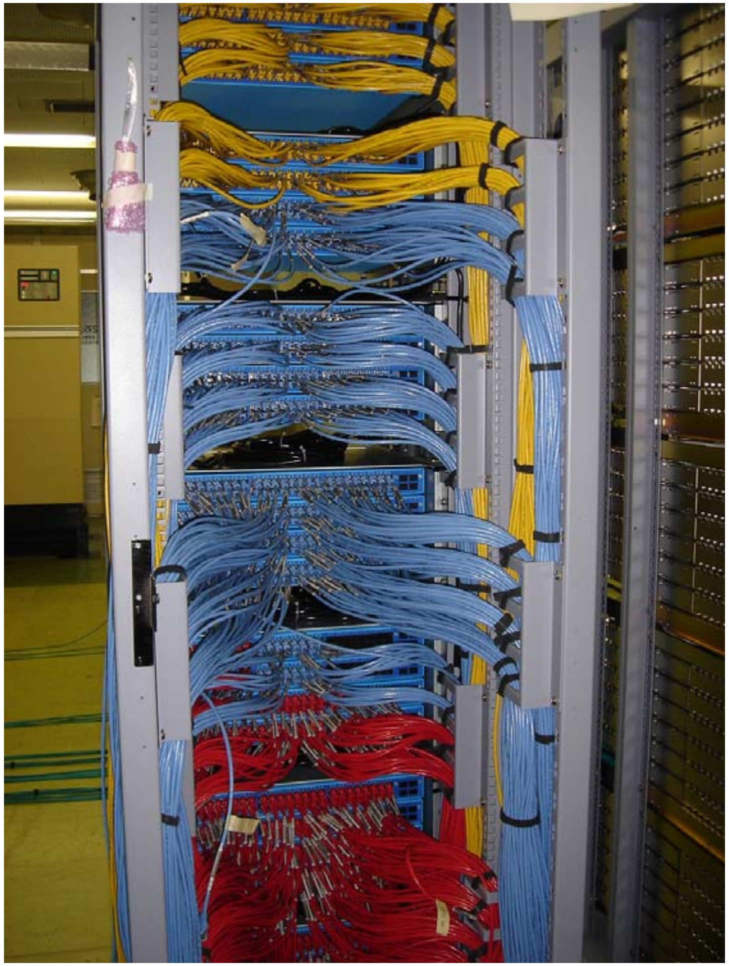
Summary of PACS-CS spec.

# of nodes	2560 (16 x 16 x 10)
peak performance	14.3 Tflops
node configuration	single CPU / node
CPU	Intel LV Xeon EM64T, 2.8GHz, 1MB L2 cache
memory	2GB/node (5.12 TB/system), DDR2 interleaved
network for parallel processing	3-dimensional Hyper-Crossbar Network
link bandwidth	one-sided: 250MB/s/dim. one-sided: 750MB/s (3-D simultaneous trans.)
local HDD	160 GB/node（RAID-1）
total system size	59 rack
power consumption	550 kW

Real machine (just finished to construct!)

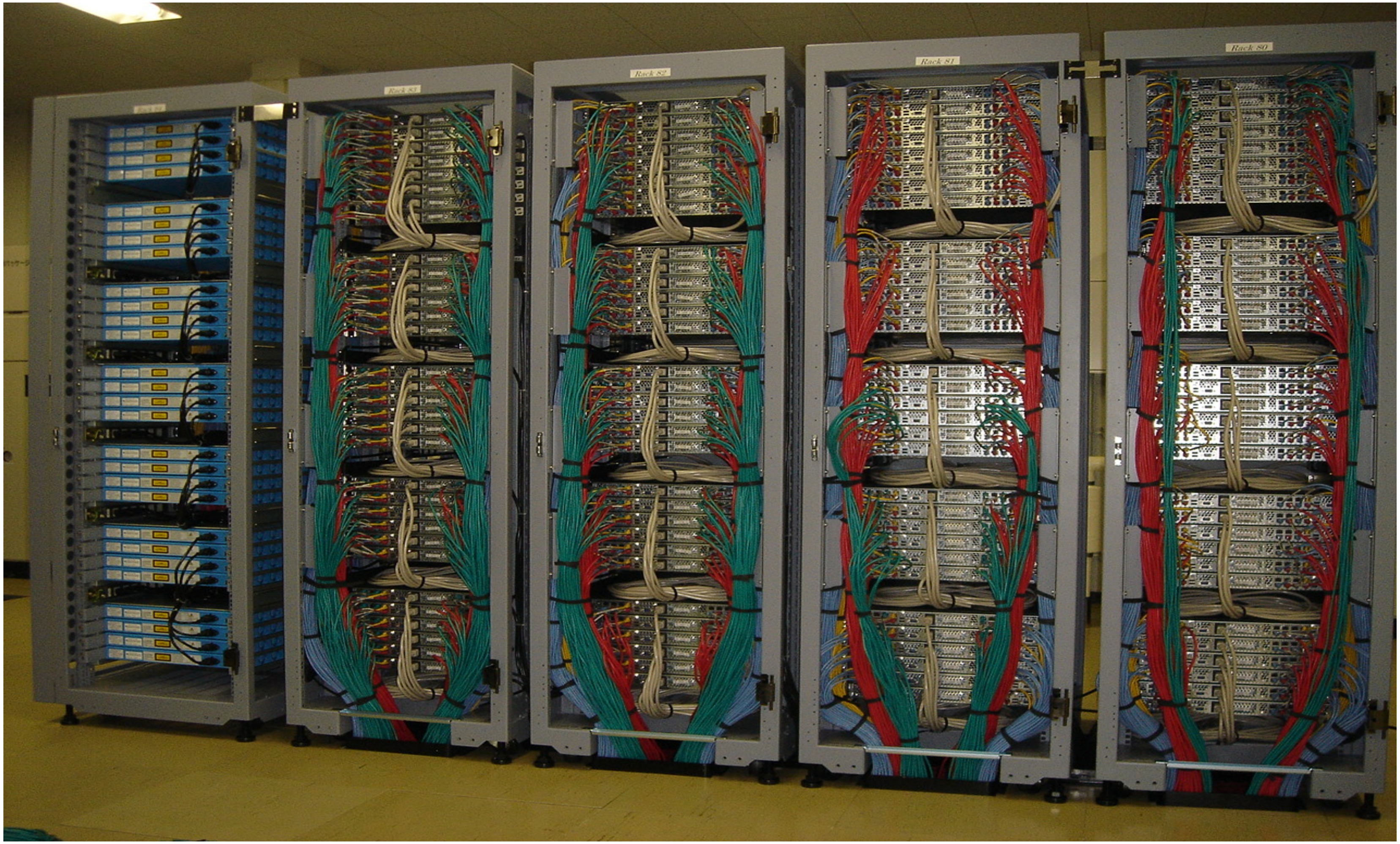


Node Rack



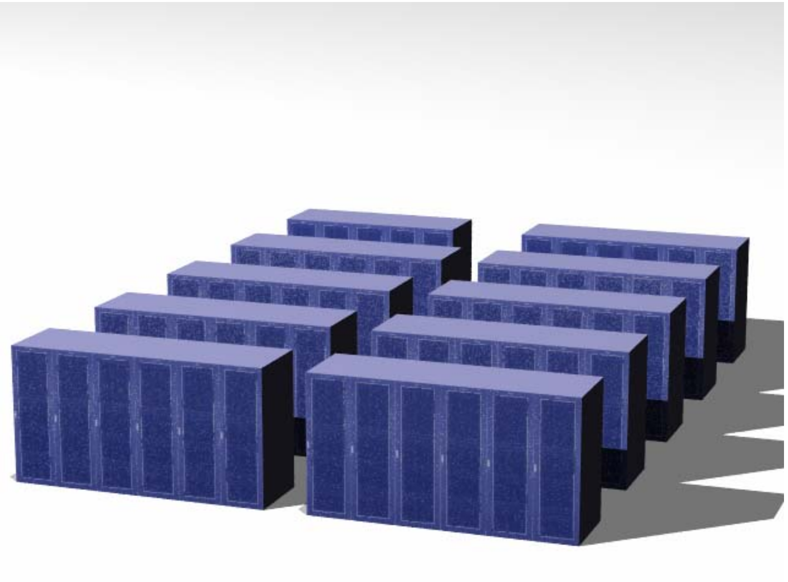
Switch Rack

Real machine (cont'd)



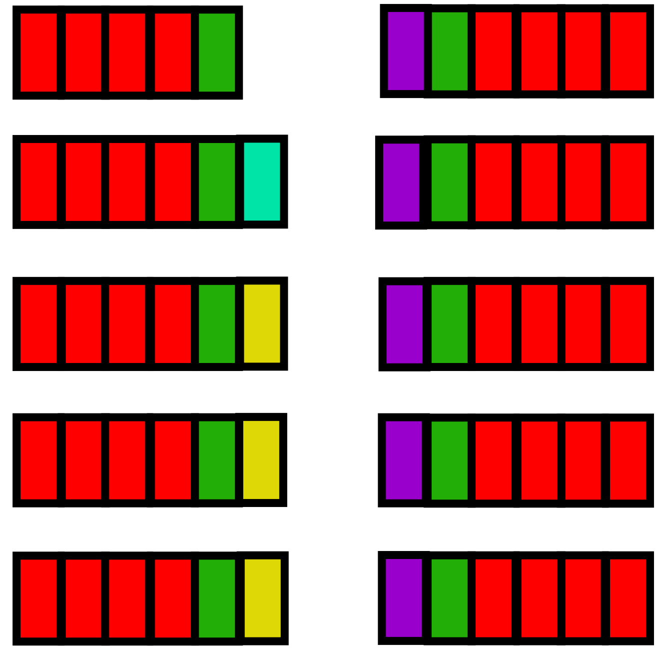
Full system of PACS-CS (2560 nodes)

Whole system image and rack floor plan



Whole system image

Node racks and network swtich racks are separated



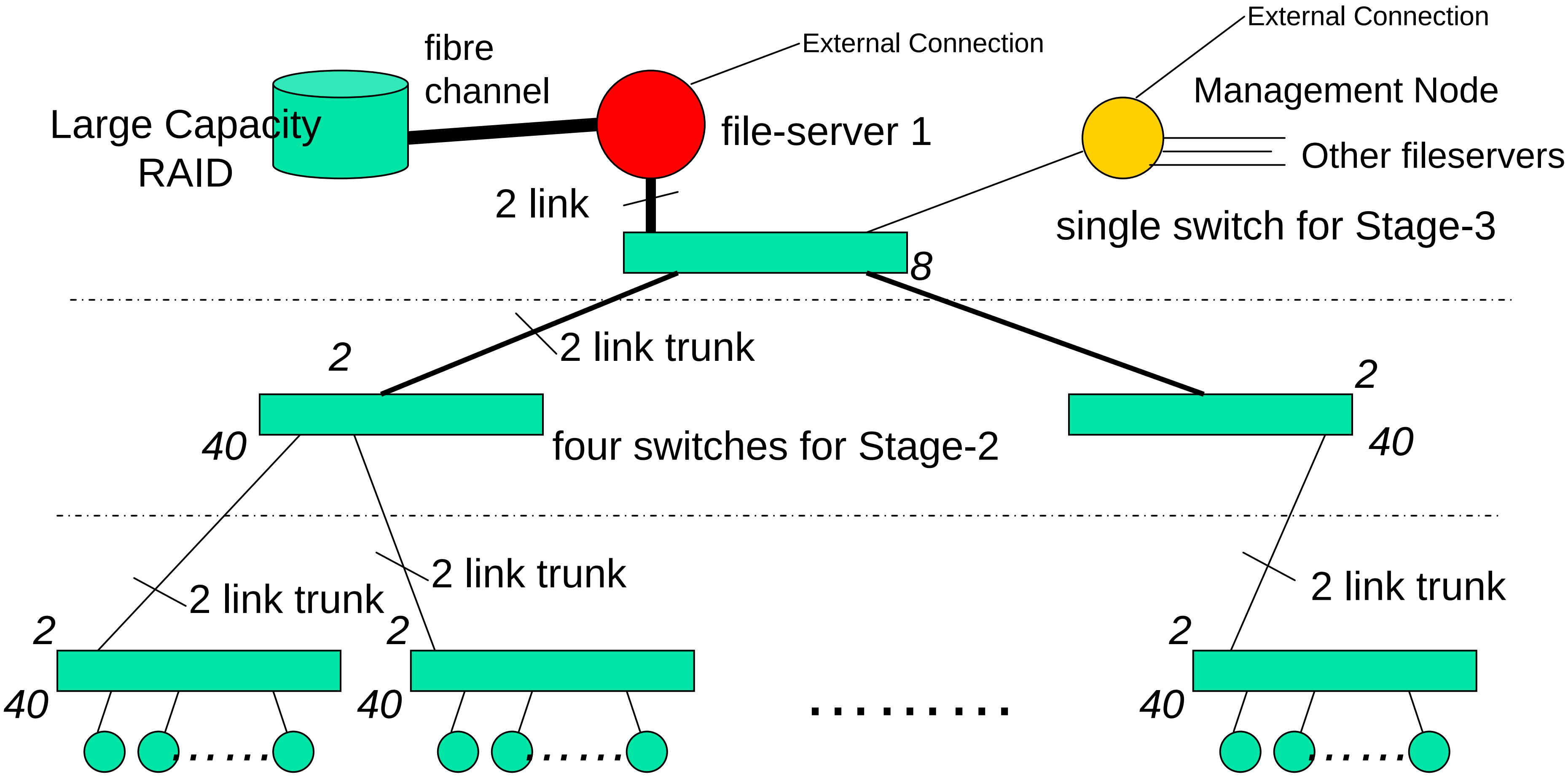
- Node rack
- X & Y dim. switch rack
- Z dim. switch rack
- Operation switch rack
- File server & RAID

4 categories of interconnection network

- Parallel Processing Network (for Data)
  - 3-D HXB network based on dual-GbE trunking
  - For high speed parallel processing on applications
- General Purpose Network
  - Generic tree network with link aggregation (LACP)
  - Generic UNIX network processing : NFS, NIS, DNS, rsh, ...
- Operation & Maintenance Network
  - Tree network
  - Remote operation (power on/off, reboot, individual/broadcast console access) to each/all node
- Surveillance Network
  - Watching a large number of (about 380) switches
  - All switches are managed and monitored by SNMP

General Purpose Network

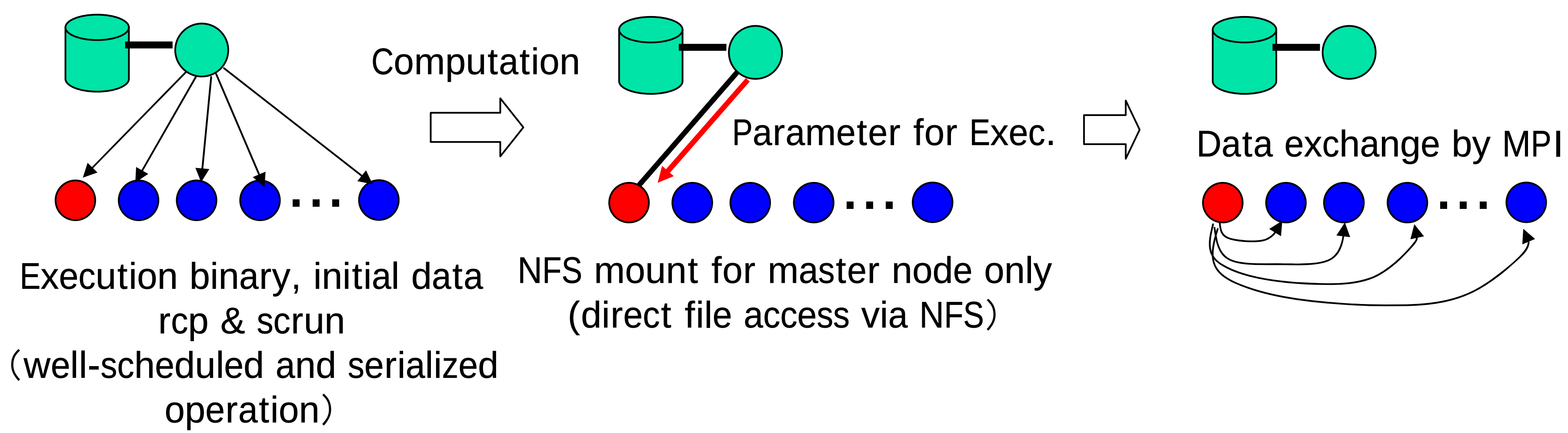
- Basic fabric: 48 port GbE switch (with LACP)
- For a single file-server （multiplied by the number of file servers）



64 switches for Stage-1（40 CPU × 64=2560 CPU）

Access to the file server

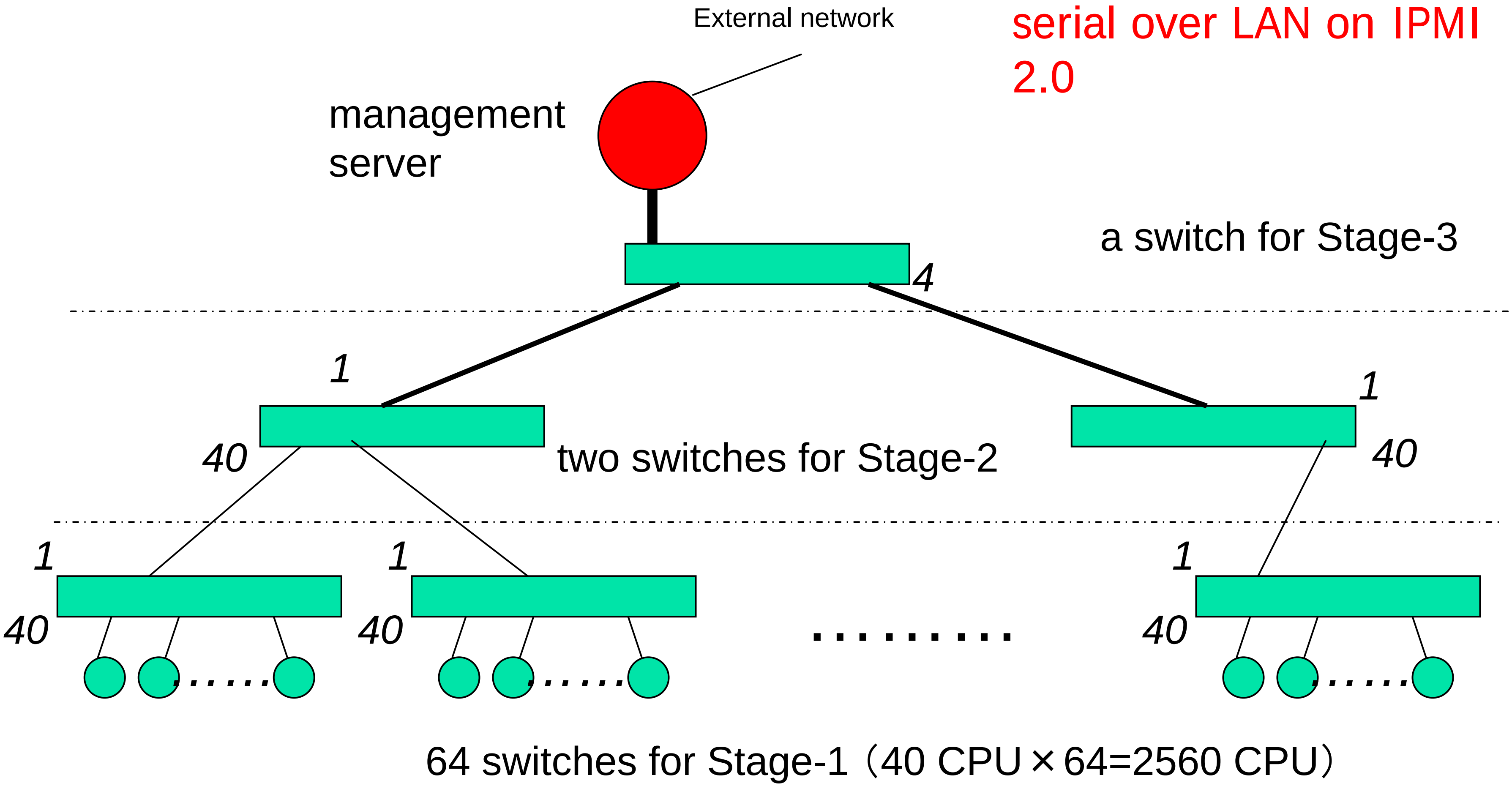
- "All nodes access the file server simultaneously" is impossible
- Each node is equipped with a local HDD (160GB) - data have to be copied
  from file server before calculation
- Result file are transferred to the file server after computation
- Well-scheduled file transfer to keep a reasonable bandwidth
- Several special nodes can NFS mount to the file server
  ⇒ Dynamic parameter changing



Operation & Maintenance Network

- Basic fabric: 48 ports GbE switch

Remote console feature by serial over LAN on IPMI 2.0

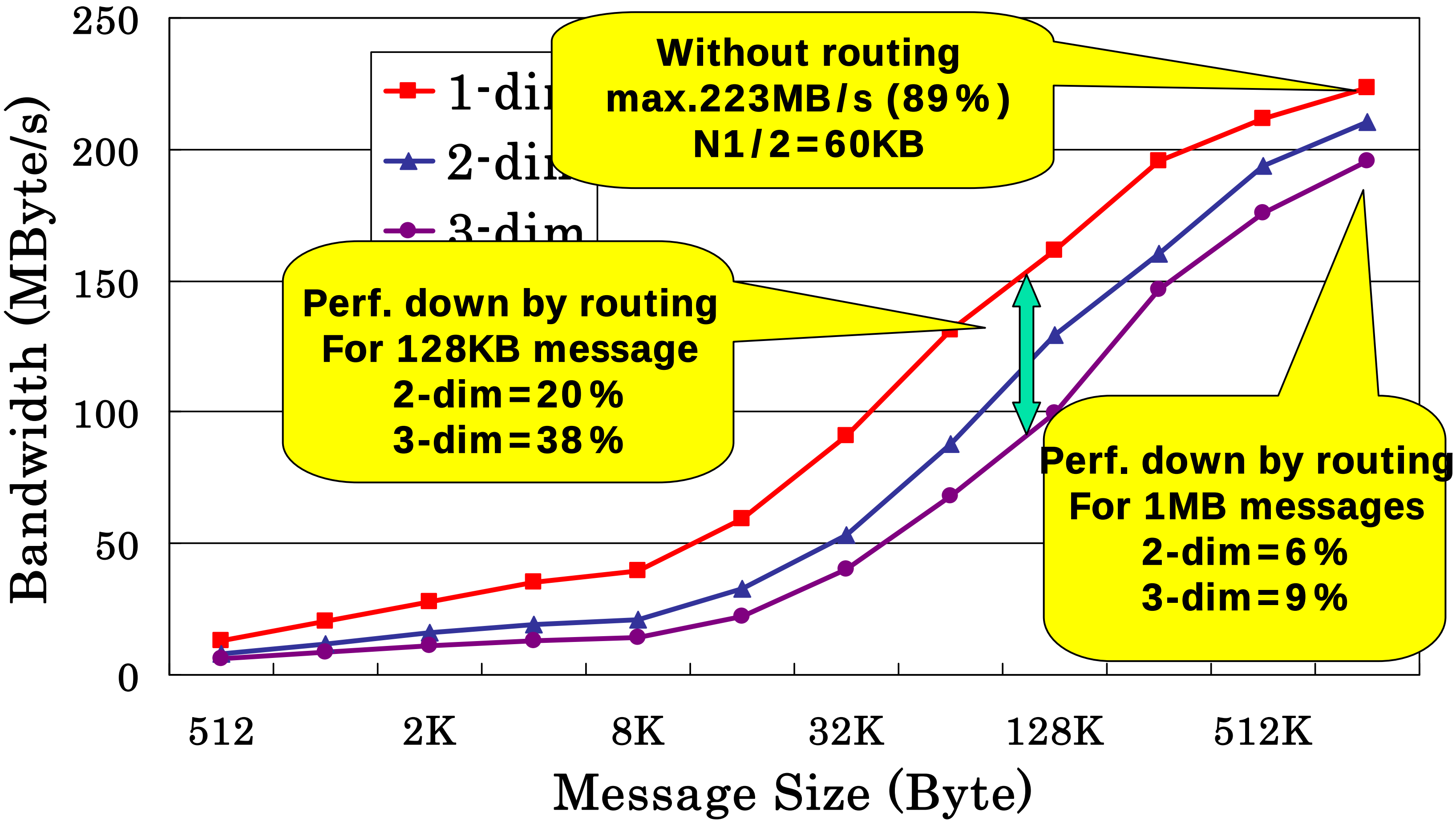


Software

- Linux + SCore
  - SCore: Cluster middleware to support partitioning and job management for thousands of nodes
  - PM/Ethernet-HXB driver
  - Checkpoint/Restart
- Scalable health-care monitor with GUI
- Batch/Queue（OpenPBS）
- Parallel programming in MPI
- Languages: Fortran90, C, C++
- Math Libraries

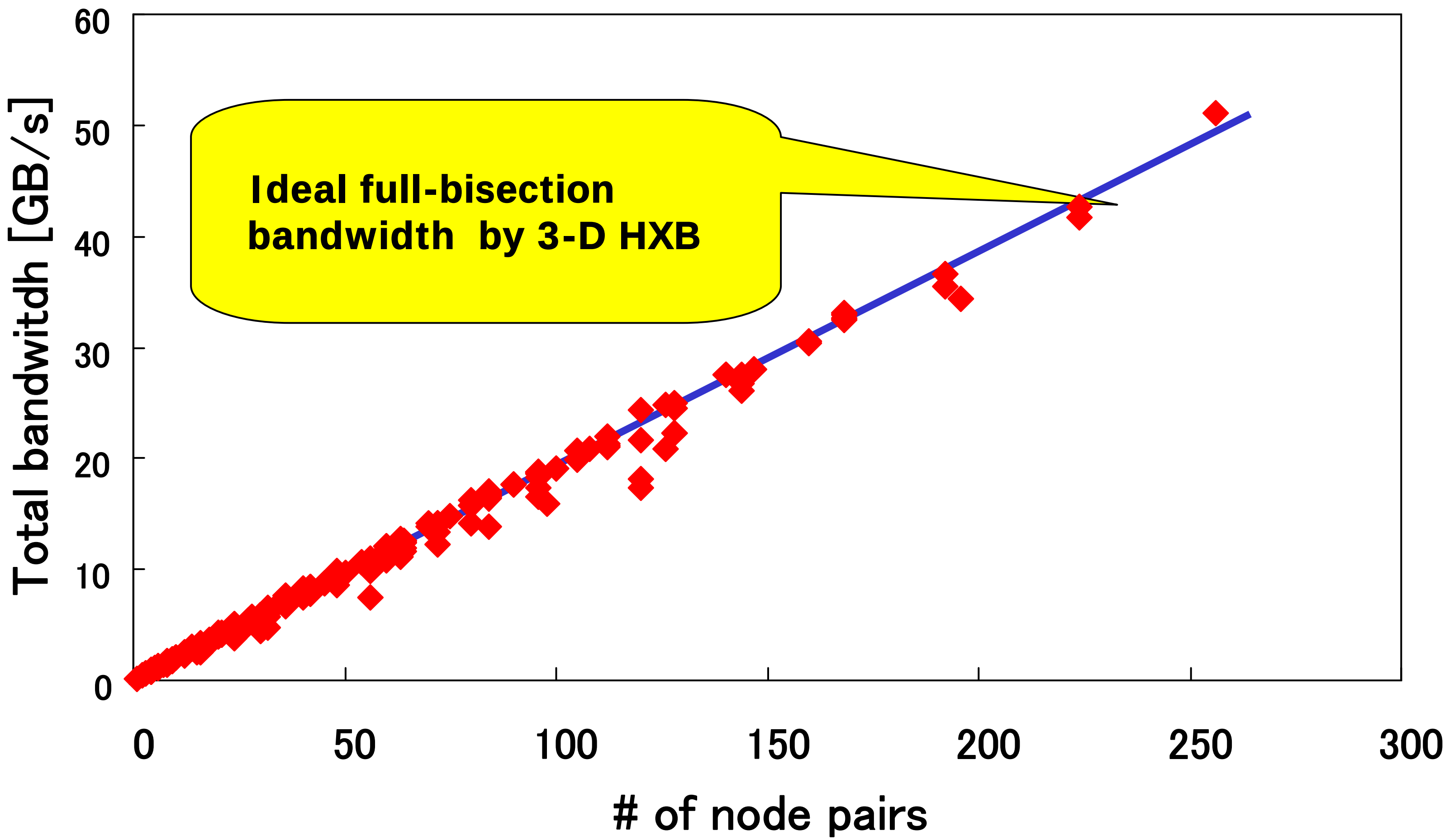
P2P performance on 1-dim communication

- Ping-pong performance in MPICH (single dimension)



Aggregated bisection bandwidth

- Bisection bandwidth in MPICH (Simultaneous ping-pong on a single dimension) 8x8x8=512 node, 1MB message on 8x8x8 system)



Simultaneous communication

- Simultaneous 3-D comm. (burst data transfer of long message)

Bandwidth/node [MB/s] (% to peak)	256 node	512 node
average	586.8 (78.2%)	582.0 (77.6%)
max.	619.3 (82.6%)	629.6 (84.0%)
min.	559.2 (74.6%)	434.0 (57.9%)

Almost 80% of performance is achieved on simultaneous 3-D comm.

- Simultaneous 1-D. comm. with routing (diagonal comm., burst transfer)

Bandwidth/node [MB/s] (% to peak)	256 node	512 node
average	186.2 (74.5%)	184.1 (73.6%)
max.	123.5 (49.4%)	232.5 (93.0%)
min.	122.7 (49.1%)	230.9 (92.4%)

Almost 70% of performance is achieved with routing function

Linpack performance

- 10.35 TFLOPS with 2560 nodes
(#34 at 2006/Jun. TOP500 list, #2 as Made-in-Japan machine)
- Performance differs by 2-D array configuration on HPL

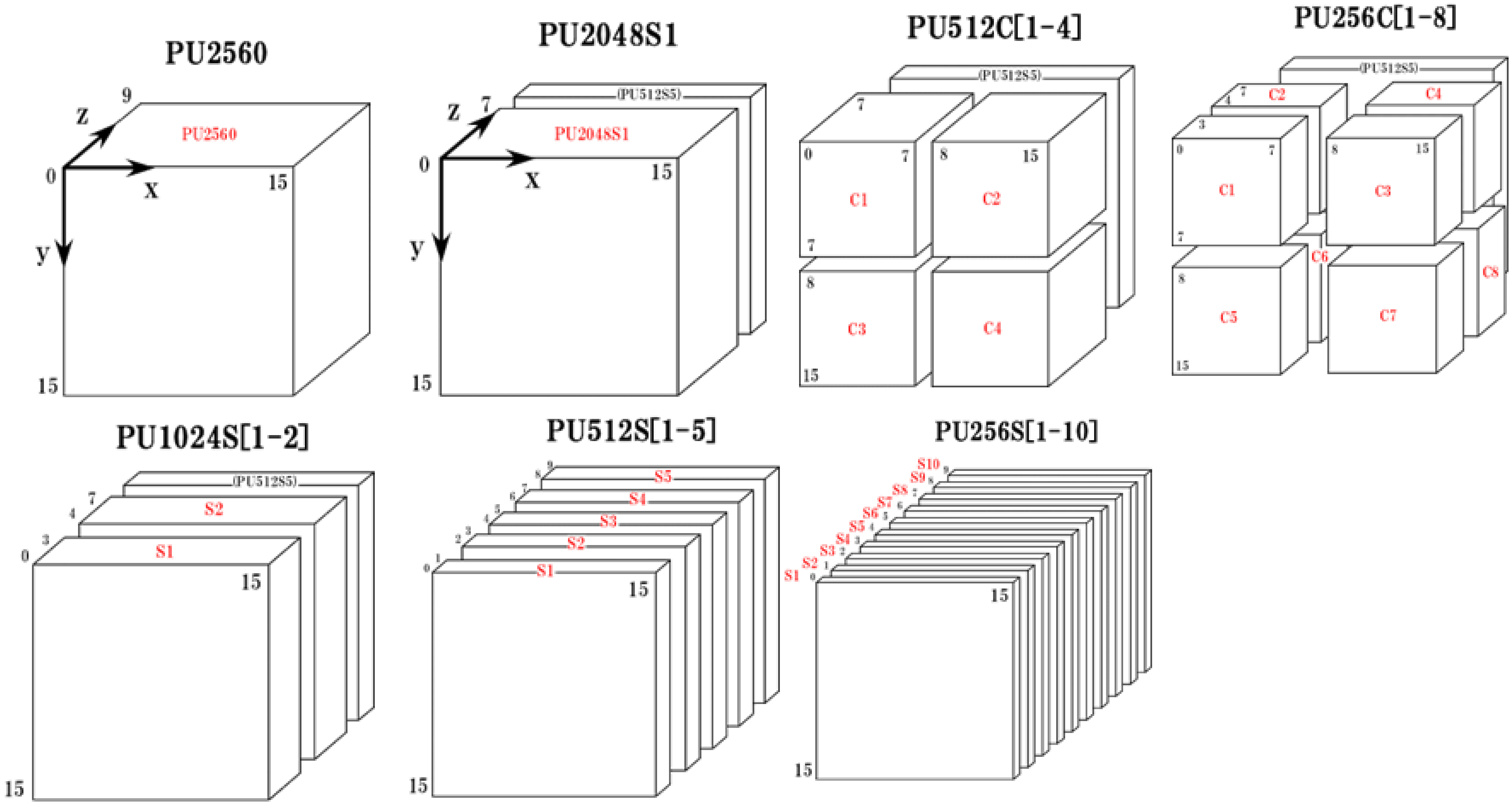
P×Q	N	Rmax (Tflops)	Efficiency (%)	3D-HXB routing
16×160	706560	10.33	72.05	No
32×80	722944	10.35	72.20	Yes

- 6.7 hours of running time
- No error on 10 times of running
- We performed node aging test, complete check of 5000 of HDD, and system level check before full installation

Installation

- System design and implementation
  - Hardware + basic software : Hitachi
  - Network software: Fujitsu
- "Multi-Vendor" solution is available for such a commodity-based work
- System operation started from July 2006
- Mostly used with 256 or 512 nodes of groups (1.4 – 2.8 TFLOPS/partition)
- Performance tuning for 3-D simultaneous communication is still under going ⇒new PMvX API & library
- PM/Ethernet-HXB on SCore middleware was released under GPL

Partitions



Mixing and scheduling them according to user's request and applications

Operation

- Currently under “Interdisciplinary Computational Science Promotion Program” with 13 of research groups
- Most of jobs are running with 256 or 512 node group, some with 128 node group
- Almost 100% of resource is always used (except regular & irregular maintenance)
- Very stable with a couple of troubles on CPU, motherboard under regular maintenance (every month)
- A couple of local HDD failure per month, but covered by RAID-1 system and no effect on application running

Applications running now

- Particle Physics: Lattice QCD
- Material Science:
  - RS-DFT (Real Space Density Function Theory)
  - QM/MM, MD (AMBER), CPMD
- Nuclear Physics & Material Science: RT-DFT (Real-Time DFT)
- Geoenvironment: NICAM & WRF (climate simulation)
- Biology: Tree-Puzzle
- ….

(details will be presented in each division/group report)

Conclusions

- PACS-CS is an MPP based on commodity technology
- Balance among CPU : Memory : Network performances is the essential
- High cost-effectiveness network suitable for a certain class of applications : GbE trunked 3D-HXB
- Same density with traditional dual CPU SMP node
- 2560 CPU, 14.3 TFLOPS system is running since July 2006