Convergence of Supercomputing and Extreme Big Data on the TSUBAME Supercomputer Exascale

松岡聡・東工大
Satoshi Matsuoka
Tokyo Institute of Technology

OpenSFS @ Tokyo
2013/10/17

TSUBAME2.5
Supercomputer

Themes of the Day...

- How do you respond to the followings?
- "We don't need to invest in all that supercomputer R&D stuff; we invest into clouds, mobiles, etc., for big data and we will just leverage off those..."
- "Sure, supercomputers are pretty big, but giants Google/Amazon/... will have enough resource in the cloud for big data, so we will just use those..."

The current "Big Data" are not really that Big...

- Typical "real" definition: "Mining people's privacy data to make money"
- Corporate data are usually in data warehoused silo -> limited volume, in Gigabytes~Terabytes, seldom Petabytes.
- Processing involve simple O(n) algorithms, or those that can be accelerated with DB-inherited indexing algorithms
- Executed on re-purposed commodity "web" servers linked with 1Gbps networks running Hadoop/HDFS
- Vicious cycle of stagnation in innovations...
- **Breaking Down of Corporate Silos⇒ Convergence with Supercomputing with <u>Extreme Big Data</u>**
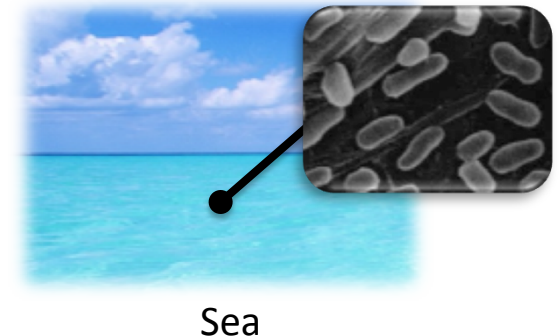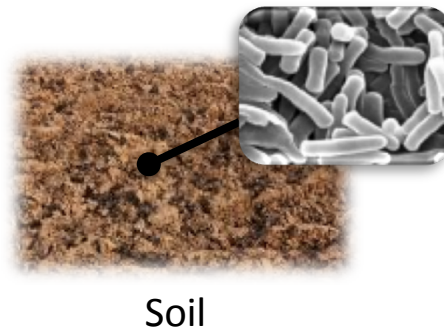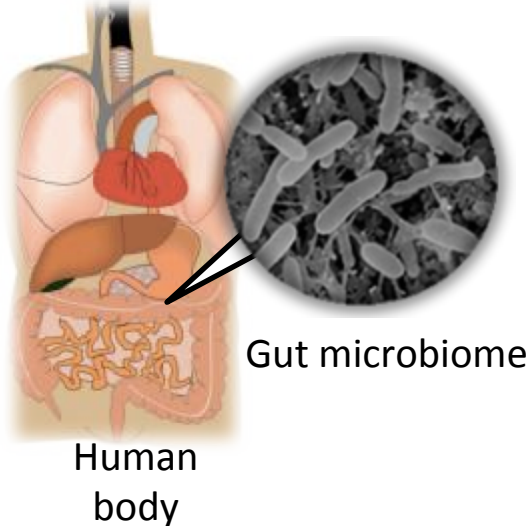
We will have tons of unknown genes

Metagenome analysis

- Directly sequencing uncultured microbiomes obtained from target environment and analyzing the sequence data
  - Finding novel genes from unculturable microorganism
  - Elucidating composition of species/genes of environments

Examples of microbiome

Gut microbiome

Human body

Soil

Sea
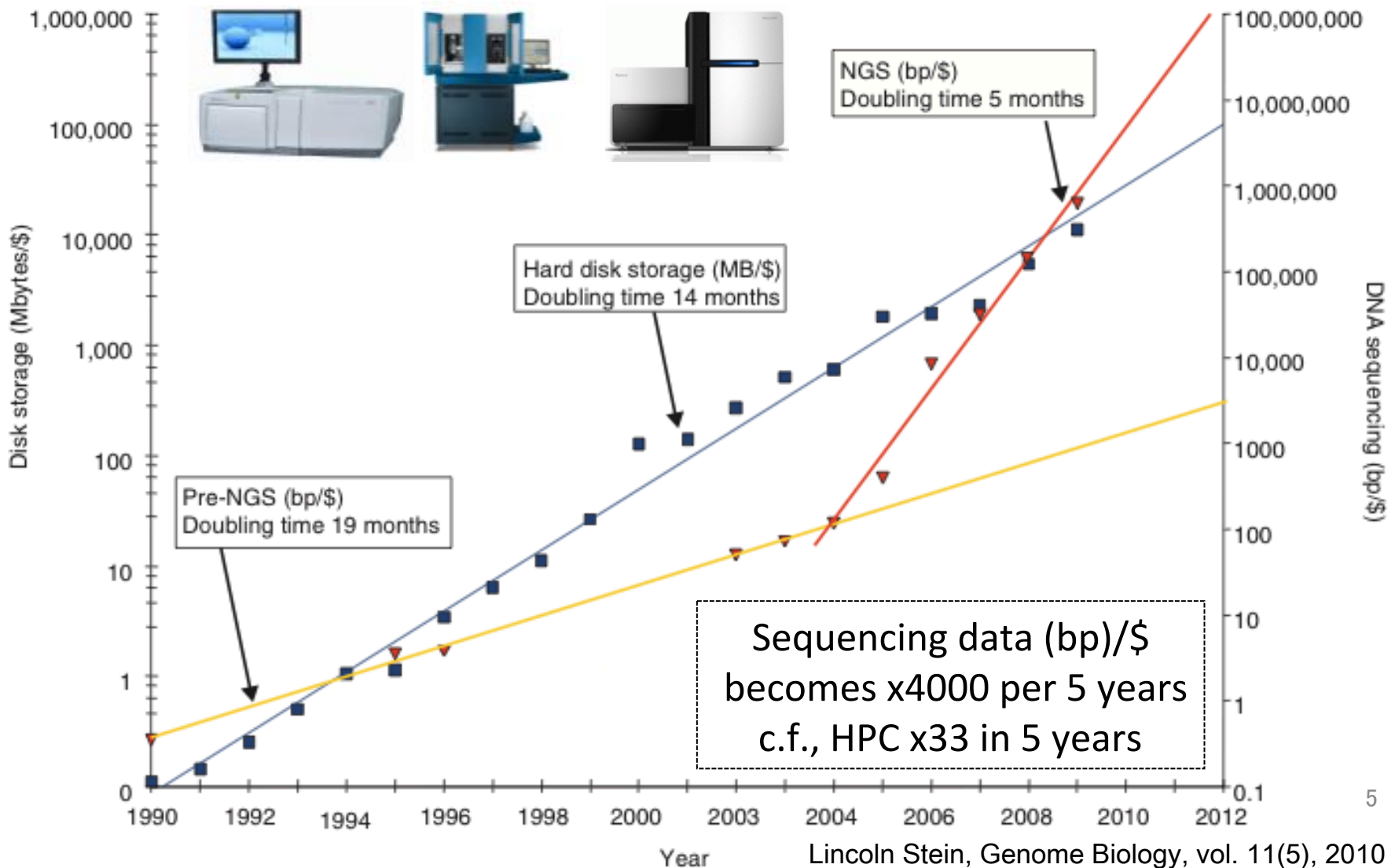
Extreme Big Data in Genomics

Impact of new generation sequencers

[Slide courtesy Yutaka Akiyama @ Tokyo Tech.]

Sequencing data (bp)/$
becomes x4000 per 5 years
c.f., HPC x33 in 5 years

Lincoln Stein, Genome Biology, vol. 11(5), 2010

Extreme Big Data Example in Social NW rates and volumes are immense

Slide courtecy David A. Bader @ Georgia Tech

- Facebook:
  - ~1 billion users
  - average 130 friends
  - 30 billion pieces of content shared / month
- Twitter:
  - 500 million active users
  - 340 million tweets / day
- Internet – 100s of exabytes / year
  - 300 million new websites per year
  - 48 hours of video to You Tube per minute
  - 30,000 YouTube videos played per second

Continuous Billion-Scale Social Simulation with Real-Time Streaming Data (Toyotaro Suzumura/IBM-Tokyo Tech)
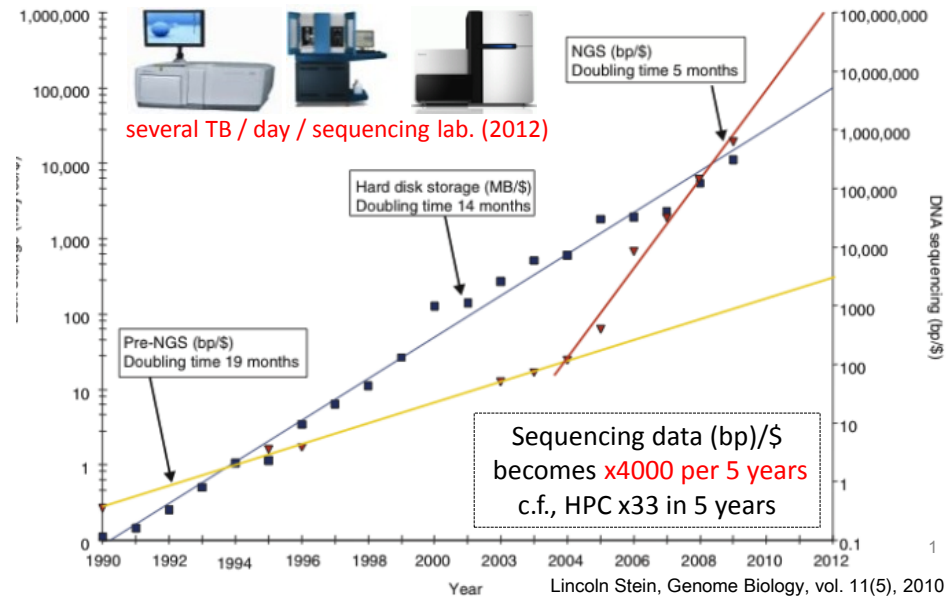
- **Applications**
  - Target Area: Planet (Open Street Map)
  - **7 billion people**
- **Input Data**
  - Road Network (Open Street Map) for Planet: **300 GB** (XML)
  - Trip data for 7 billion people
    - **10 KB (1 trip) x 7 billion = 70 TB**
  - Real-Time Streaming Data (e.g. Social sensor, physical data)
- **Simulated Output for 1 Iteration**
  - **700 TB**

Extreme Big Data in Genomics

Impact of new generation sequencers

[Slide Courtesy Yutaka Akiyama @ Tokyo Tech.]

several TB / day / sequencing lab. (2012)

- NGS (bp/$) Doubling time 5 months
- Hard disk storage (MB/$) Doubling time 14 months
- Pre-NGS (bp/$) Doubling time 19 months

Sequencing data (bp)/$ becomes x4000 per 5 years c.f., HPC x33 in 5 years

Lincoln Stein, Genome Biology, vol. 11(5), 2010

Future "Extreme Big Data"

- **NOT mining Tbytes Silo Data**

- **Peta~Zetabytes of Data**
- **Ultra High-BW Data Stream**
- **Highly Unstructured, Irregular**
- **Complex correlations between data from multiple sources**
- **Extreme Capacity, Bandwidth, Compute All Required**

6

"Extreme Big Data" will change everything

- "Breaking down of Silos" (Rajeeb Harza, Intel VP of Technical Computing)
- Already happening in Science & Engineering due to Open Data movement
- More complex analysis algorithms: O(n log n), O(m x n), …
- Will become the NORM for competitiveness reasons.

TSUBAME2.0 Nov. 1, 2010
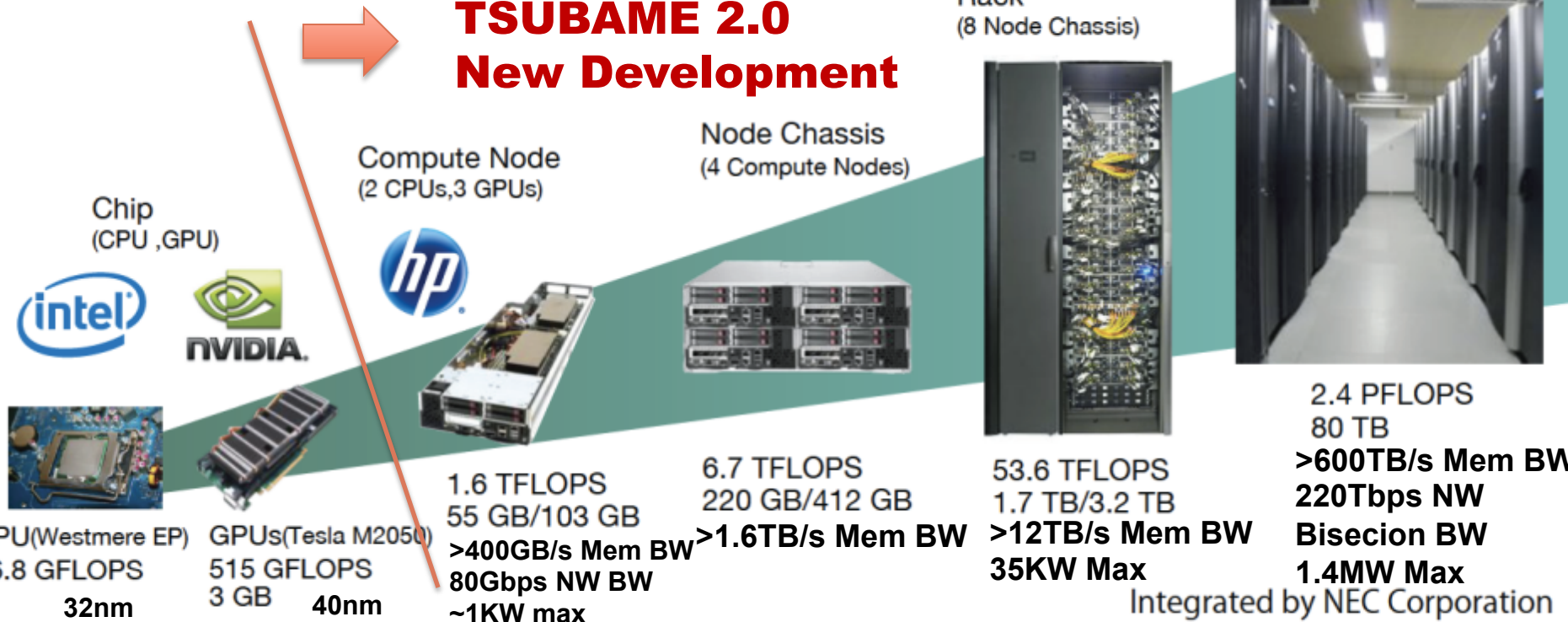## "The Greenest Production Supercomputer in the World"

TOKYO TECH

TSUBAME2.0: A GPU-centric Green 2.4 Petaflops Supercomputer

Tsubame 2.0: "Tiny" footprint, very power efficient
- Floorspace less than 200m² (2,100 ft²)
- Top-class power efficient machine on the Green 500

TSUBAME 2.0 New Development

System
(42 Racks)
1408 GPU Compute Nodes,
34 Nehalem "Fat Memory" Nodes

Rack
(8 Node Chassis)

Node Chassis
(4 Compute Nodes)

Compute Node
(2 CPUs, 3 GPUs)

Chip
(CPU, GPU)

intel

NVIDIA

hp

CPU(Westmere EP)
76.8 GFLOPS
32nm

GPUs(Tesla M2050)
515 GFLOPS
3 GB   40nm

1.6 TFLOPS
55 GB/103 GB
>400GB/s Mem BW
80Gbps NW BW
~1KW max

6.7 TFLOPS
220 GB/412 GB
>1.6TB/s Mem BW

53.6 TFLOPS
1.7 TB/3.2 TB
>12TB/s Mem BW
35KW Max

2.4 PFLOPS
80 TB
>600TB/s Mem BW
220Tbps NW
Bisecion BW
1.4MW Max

Integrated by NEC Corporation

TSUBAME2.0 Compute Node

Thin Node

Infiniband QDR x2 (80Gbps)

1.6 Tflops
400GB/s
Mem BW
80GBps NW
~1KW max

Productized
as HP ProLiant
SL390s

HP SL390G7 (Developed for TSUBAME 2.0)
GPU: NVIDIA Fermi M2050 x 3
  515GFlops, 3GByte memory /GPU
CPU: Intel Westmere-EP 2.93GHz x2
(12cores/node)
Multi I/O chips, 72 PCI-e (16 x 4 + 4 x 2) lanes --- 3GPUs + 2 IB QDR
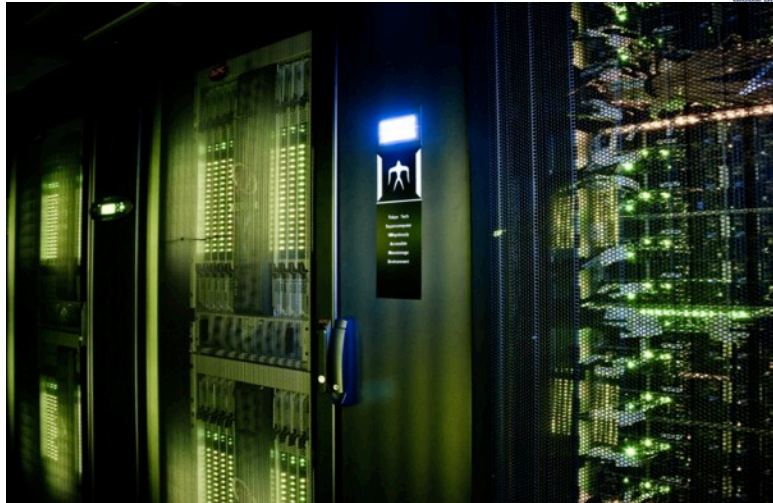Memory: 54, 96 GB DDR3-1333
SSD：60GBx2, 120GBx2

Total Perf
2.4PFlops
Mem：~100TB
SSD: ~200TB

2010: TSUBAME2.0 as No.1 in Japan

**Total 2.4 Petaflops
#4 Top500, Nov. 2010**

*All Other Japanese
Centers on the Top500
COMBINED 2.3 PetaFlops*

TSUBAME Wins Awards...

"Greenest Production Supercomputer in the World"

the Green 500

Nov. 2010, June 2011

(#4 Top500 Nov. 2010)

TSUBAME Wins Awards...

SC11

ACM Gordon Bell Prize
Special Achievements in Scalability and Time-to-Solution

**Takashi Shimokawabe, Takayuki Aoki,
Tomohiro Takaki, Akinori Yamanaka,
Akira Nukada, Toshio Endo,
Naoya Maruyama, Satoshi Matsuoka**

*Peta-Scale Phase-Field Simulation for Dendritic
Solidification on the TSUBAME 2.0 Supercomputer*

Scott Lathrop
SC11 Conference Chair

Thom H. Dunning, Jr.
Gordon Bell Chair

ACM Gordon Bell Prize 2011

Special Achievements in Scalability and Time-to-Solution

**"Peta-Scale Phase-Field Simulation for Dendritic
Solidification on the TSUBAME 2.0 Supercomputer"**

TSUBAME Wins Awards...

Commendation for Sci &Tech by Ministry of Education 2012 (文部科学大臣表彰)

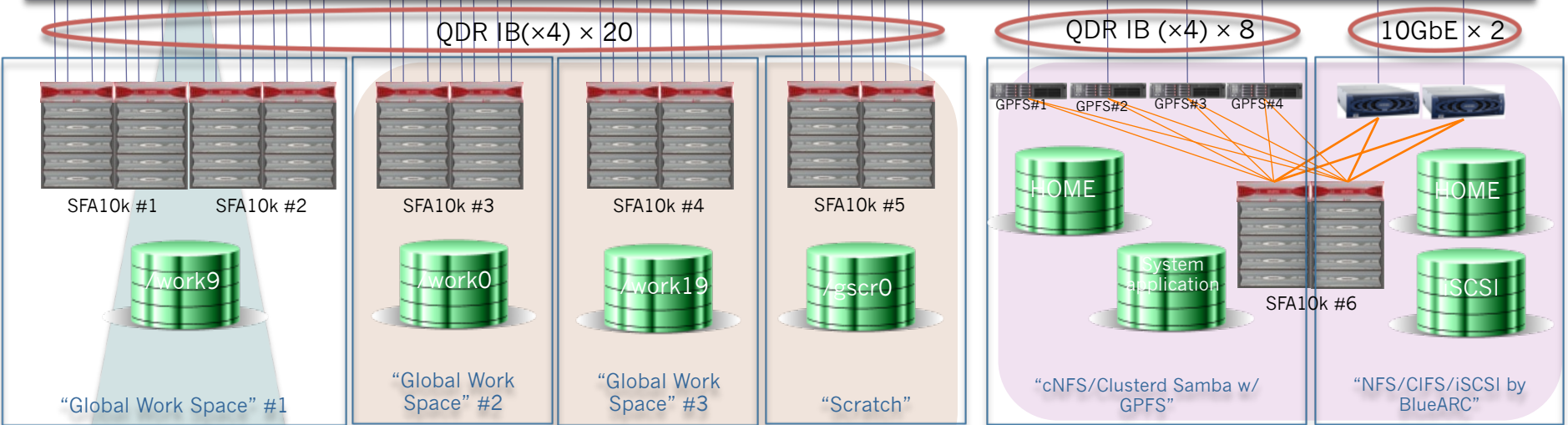**Prize for Sci & Tech, Development Category**
Development of Greenest Production Peta-scale Supercomputer

Satoshi Matsuoka, Toshio Endo, Takayuki Aoki

TSUBAME2.0 Storage Overview

TSUBAME2.0 Storage 11PB（7PB HDD, 4PB Tape）

Infiniband QDR Network for LNET and Other Services

QDR IB(×4) × 20

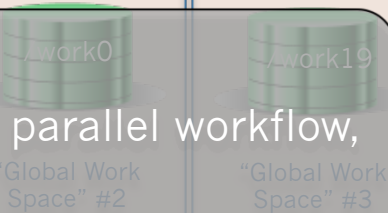QDR IB (×4) × 8

10GbE × 2

SFA10k #1　　SFA10k #2

SFA10k #3　　SFA10k #4　　SFA10k #5

GPFS#1　GPFS#2　GPFS#3　GPFS#4

/work9

/work0

/work19

/gscr0

HOME

System application

SFA10k #6

HOME

iSCSI

"Global Work Space" #1

"Global Work Space" #2

"Global Work Space" #3

"Scratch"

"cNFS/Clusterd Samba w/ GPFS"

"NFS/CIFS/iSCSI by BlueARC"

Lustre　　3.6 PB

Home Volumes　1.2PB

GPFS with HSM

Parallel File System Volumes

"Thin node SSD"

"Fat/Medium node SSD"

2.4 PB HDD + ～4PB Tape

250 TB, 300TB/s

130 TB=> 500TB～1PB

Scratch

Grid Storage

TSUBAME2.0 Storage Overview

TSUBAME2.0 Storage 11PB（7PB HDD, 4PB Tape）

Infiniband QDR Network for LNET and Other Services

QDR IB(×4) × 20   QDR IB (×4) × 8   10GbE × 2

SFA10k #1   SFA10k #2   SFA10k #3   SFA10k #4   SFA10k #5

GPFS#1   GPFS#2   GPFS#3   GPFS#4   HOME

- Home storage for computing nodes
- Cloud-based campus storage services

Concurrent Parallel I/O (e.g. MPI-IO)

/work9   /work0   /work19   /gscr0   System Application   SFA10k #6   iSCSI

Read mostly I/O (data-intensive apps, parallel workflow, parameter survey)

"Global Work Space" #1   "Global Work Space" #2   "Global Work Space" #3   "Scratch"   "cNFS/Clusterd Samba w/ GPFS"   "NFS/CIFS/iSCSI by BlueARC"

GPFS with HSM   Lustre?? 6 PB   Home Volumes **1.2PB**

Parallel File System Volumes

Fine-grained R/W I/O (checkpoints, temporary files, Big Data processing)

Long-Term Backup

2.4PB HDD + ~4PB Tape

"Thin node SSD"   "Fat/Medium node SSD"

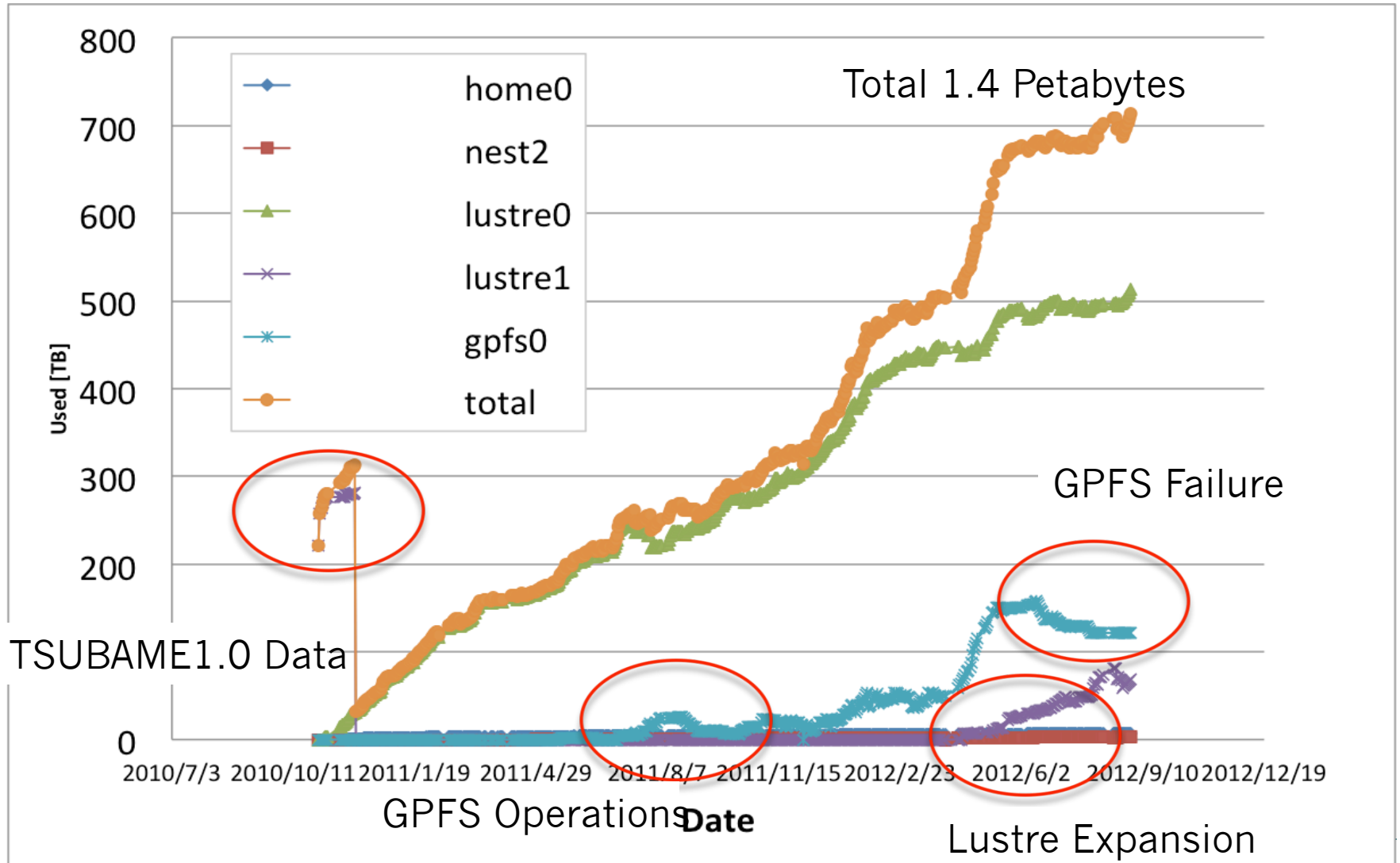Data transfer service between SCs/CCs

130 TB=> 500TB~1PB

250 TB, 300GB/s

Scratch   HPCI Storage

TSUBAME2.0 Storage Usage

Hadoop on TSUBAME (Tsudoop)

- Script-based invocation
  - acquire computing nodes via PBS Pro
  - deploy a Hadoop environment on the fly (incl. HDFS)
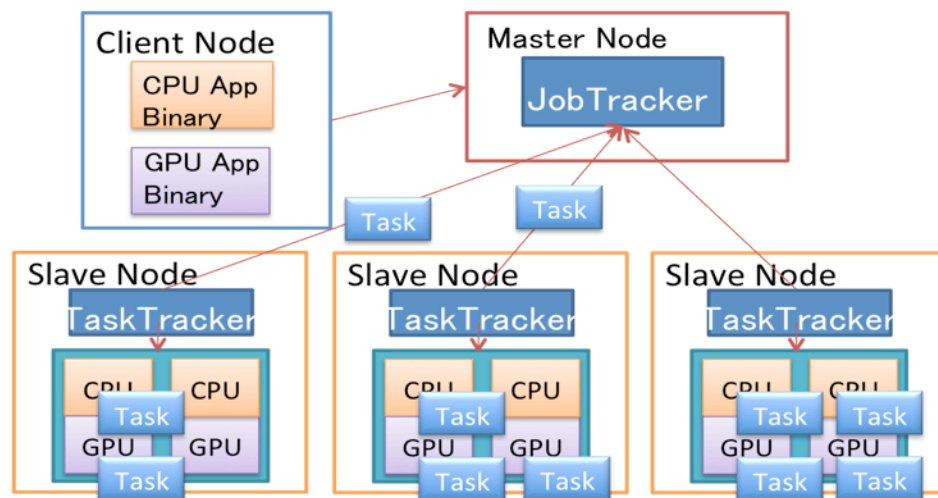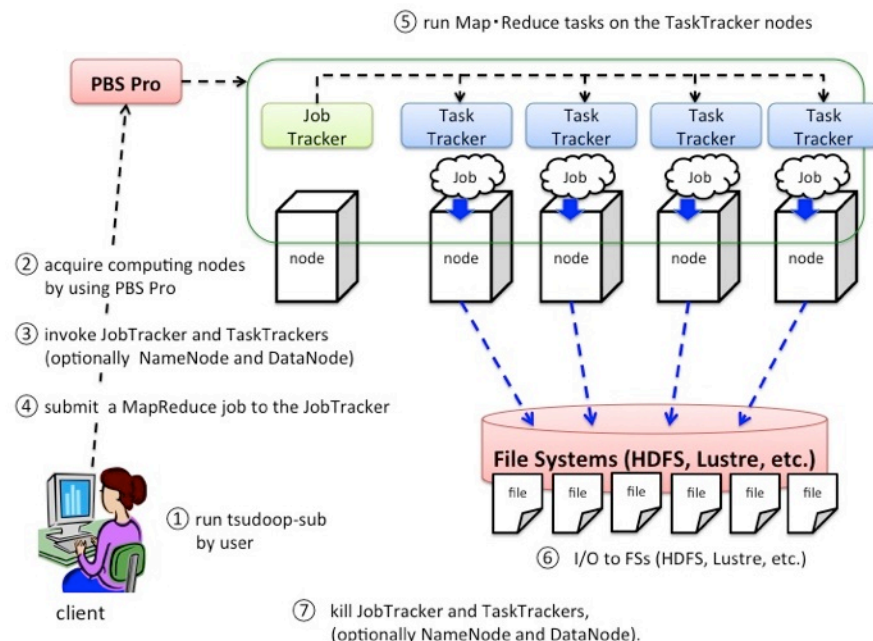 - execute a user MapReduce jobs

- Various FS support
  - HDFS by aggregating local SSDs
  - Lustre, GPFS (to appear)

- Customized Hadoop for executing CUDA programs (experimental)
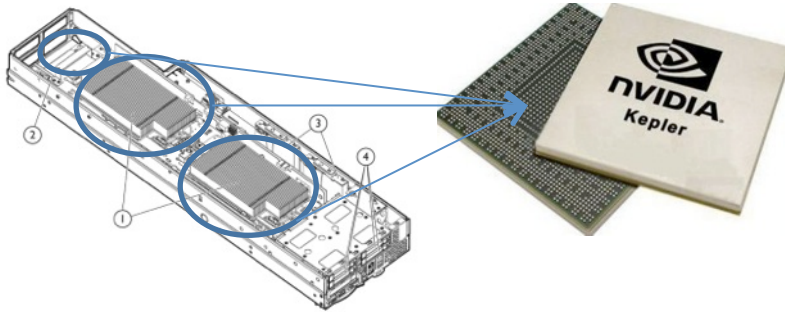 - Hybrid Map Task Scheduling
    - Automatically detects map task characteristics by monitoring
    - Scheduling map tasks to minimize overall MapReduce job execution time
    - Extension of Hadoop Pipes features

⑤ run Map・Reduce tasks on the TaskTracker nodes

② acquire computing nodes by using PBS Pro

③ invoke JobTracker and TaskTrackers (optionally NameNode and DataNode)

④ submit a MapReduce job to the JobTracker

① run tsudoop-sub by user

File Systems (HDFS, Lustre, etc.)

⑥ I/O to FSs (HDFS, Lustre, etc.)

⑦ kill JobTracker and TaskTrackers, (optionally NameNode and DataNode).

Towards TSUBAME 3.0
Interim Upgrade TSUBAME2.0 to 2.5 (Early Fall 2013)

- Upgrade the TSUBAME2.0s GPUs
  NVIDIA Fermi M2050 to Kepler K20X

SFP/DFP peak from 4.8PF/ 2.4PF => 17PF/5.7PF

c.f. The K Computer 11.2/11.2
Acceleration of Important Apps
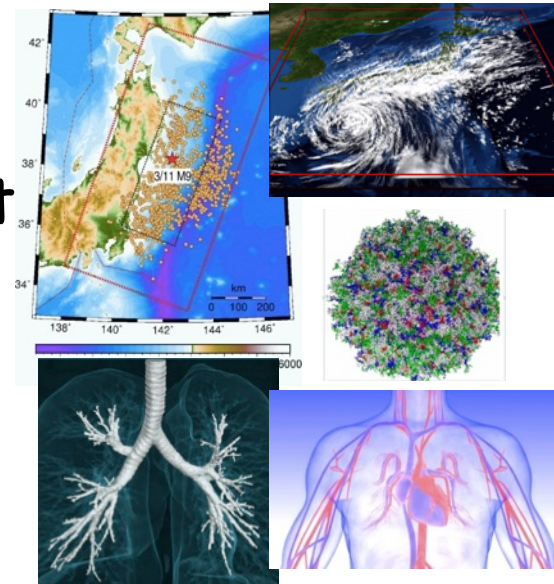Considerable Improvement
Summer 2013

TSUBAME2.0 Compute Node
Fermi GPU 3 x 1408 = 4224 GPUs

Service List

service	assigned nodes			running jobs		users
S	100%	352 / 352 nodes				
S96	100%	41 / 41 nodes		5%	41 / 76 jobs	4
G	99%	475 / 477 nodes		100%	62 / 62 jobs	12
V	83%	364 / 437 nodes		80%	1531 / 1904 jobs	3
L128	100%	10 / 10 nodes		66%	10 / 15 jobs	1
L128F	100%	10 / 10 nodes		71%		
L256	37%	3 / 8 nodes		100%	3 / 3 jobs	1
L512	100%	2 / 2 nodes		100%	2 / 2 jobs	1
H/X	93%	301 (+ 95) / 420 nodes		100%	87 / 87 jobs	8
ALL	93%	1558 (+ 95) / 1757 nodes		73%	1921 / 2615 jobs	93

Significant Capacity
Improvement at low cost
& w/o
Power Increase

TSUBAME3.0 2H2015

TSUBAME2.0⇒2.5 Thin Node Upgrade

Thin Node

Peak Perf.

4.08 Tflops
~800GB/s Mem BW
80GBps NW
~1KW max

Infiniband QDR x2 (80Gbps)

Productized as HP ProLiant **SL390s** Modified for TSUABME2.5

HP SL390G7 (Developed for TSUBAME 2.0, Modified for 2.5)

GPU: NVIDIA Kepler K20X x 3
1310GFlops, 6GByte Mem(per GPU)

CPU: Intel Westmere-EP 2.93GHz x2
Multi I/O chips, 72 PCI-e (16 x 4 + 4 x 2)
lanes --- 3GPUs + 2 IB QDR
Memory: 54, 96 GB DDR3-1333
SSD：60GBx2, 120GBx2

NVIDIA Fermi
M2050
1039/515
GFlops

NVIDIA Kepler
K20X
3950/1310
GFlops

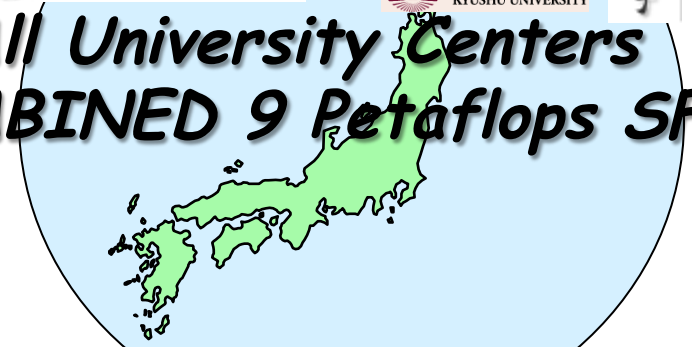|	TSUBAME2.0	TSUBAME2.5
Thin Node x 1408 Units		
Node Machine	HP Proliant SL390s	← No Change
CPU	Intel Xeon X5670 (6core 2.93GHz, Westmere) x 2	← No Change
GPU	NVIDIA Tesla M2050 x 3 ● 448 CUDA cores (Fermi) ➤ SFP 1.03TFlops ➤ DFP 0.515TFlops ● 3GiB GDDR5 memory ● 150GB Peak, ~90GB/s STREAM Memory BW	NVIDIA Tesla K20X x 3 ● 2688 CUDA cores (Kepler) ➤ SFP 3.95TFlops ➤ DFP 1.31TFlops ● 6GiB GDDR5 memory ● 250GB Peak, ~180GB/s STREAM Memory BW
Node Performance (incl. CPU Turbo boost)	● SFP 3.40TFlops ● DFP 1.70TFlops ● ~500GB Peak, ~300GB/s STREAM Memory BW	● SFP 12.2TFlops ● DFP 4.08TFlops ● ~800GB Peak, ~570GB/s STREAM Memory BW
TOTAL System		
Total System Performance	● SFP 4.80PFlops ● DFP 2.40PFlops ● Peak ~0.70PB/s, STREAM ~0.440PB/s Memory BW	● SFP 17.1PFlops (x3.6) ● DFP 5.76PFlops (x2.4) ● Peak ~1.16PB/s, STREAM ~0.804PB/s Memory BW (x1.8)

2013: TSUBAME2.5 No.1 in Japan in Single Precision FP, 17 Petaflops

All University Centers
COMBINED 9 Petaflops SFP

**Total
17.1 Petaflops SFP
5.76 Petaflops DFP**

K Computer
11.4 Petaflops SFP/DFP

Linpack Benchmark

- Linpack: Dense matrix solver by LU decomposition with pivotting
  - Used in Top500/Green500 supercomputer ranking!
- On TSUBAME2.5, we adopted "In-core" algorithm, where the whole matrix data are placed on GPU device memory
  - K20X on T2.5 has 2x larger memory than M2050 on T2.0
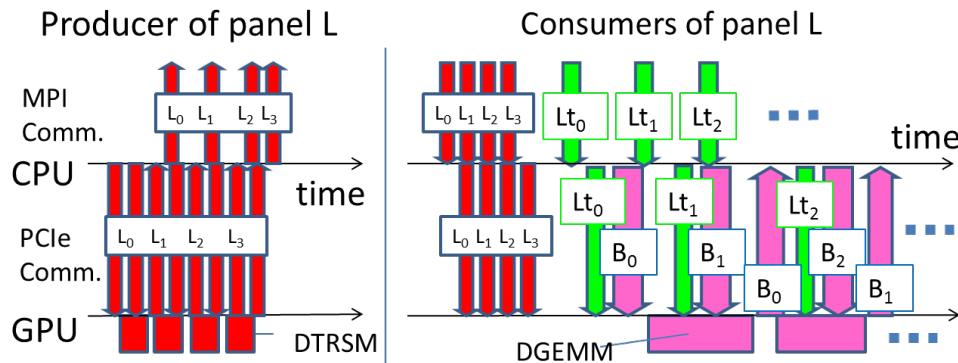 - PCIe communication has relatively larger effects

	TSUBAME2.0	TSUBAME2.5
Algorithm	Out-of-core	In-core
N (matrix size)	2,490,368	1,760,000
NB (block size)	1024	192
Speed (PFlops)	1.192	2.843
Rank in Top500	No. 4 in 11/2010	TBA in 11/2013
Power (MWatt)	1.244	0.958
Speed/Power (GFlops/Watt)	0.958	>2.40
Rank in Green500	No. 2 in 11/2010	TBA in 11/2013

2.39x (Speed)

>2.50x (Speed/Power)

High-Performance General Solver for Extremely Large-scale Semidefinite Programming Problems [Fujisawa]

1. Mathematical Programming : one of the most important mathematical programming
2. Many Applications : combinatorial optimization, control theory, structural optimization, quantum chemistry, sensor network location, data mining, etc.

Parallel Algorithm of Cholesky Factorization

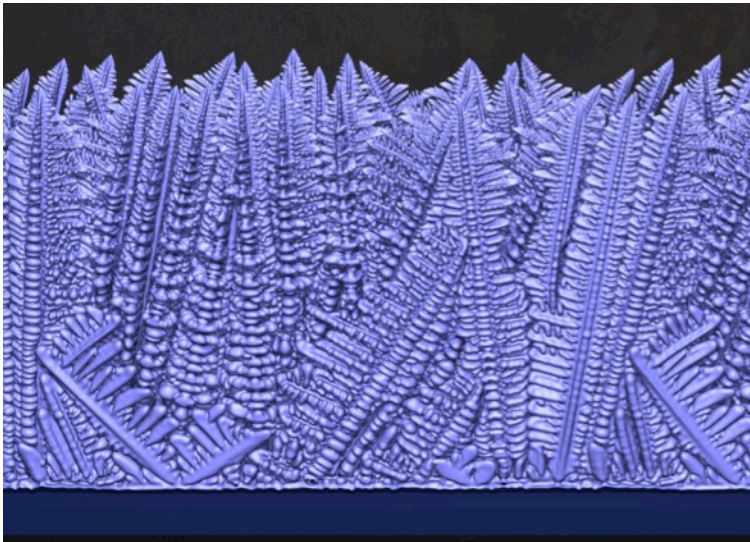GPU computation, PCI-e communication, and MPI communication are overlapped

Producer of panel L

MPI Comm.

L_0 L_1 L_2 L_3

CPU — time

PCIe Comm.

L_0 L_1 L_2 L_3

GPU — DTRSM

Consumers of panel L

L_0 L_1 L_2 L_3 Lt_0 Lt_1 Lt_2 · · ·

time

Lt_0 Lt_1 Lt_2

L_0 L_1 L_2 L_3 B_0 B_1 B_2 · · ·

B_0 B_1

DGEMM

1.713PFLOPS(DP) with 4080GPUs!!

Speed of CHOLESKY (TFlops)

1713
1526
1186
952 1019
964
826
719
825
707
513
470
506
388
314
509
438
329
306
233

number of nodes: 400 512 700 1024 1360

QAP6/org(2.0)   QAP6/new(2.0)   QAP6/new(2.5)
QAP7/org(2.0)   QAP7/new(2.0)   QAP7/new(2.5)
QAP8/org(2.0)   QAP8/new(2.0)   QAP8/new(2.5)
QAP9/org(2.0)   QAP9/new(2.0)   QAP9/new(2.5)
QAP10/org(2.0) QAP10/new(2.0) QAP10/new(2.5)

- **SDPARA** is a parallel implementation of the interior–point method for Semidefinite Programming Parallel computation for **two major bottleneck parts**
  - **ELEMENTS** ⇒ Computation of Schur complement matrix (SCM)
  - **CHOLESKY** ⇒ Cholesky factorization of Schur complement matrix (SCM)
- **SDPARA** could attain high scalability using **16,320 CPU cores** on the TSUBAME 2.5 supercomputer and some techniques of processor affinity and memory interleaving when the computation of SCM **(ELEMENTS)** constituted a bottleneck.
- With 4,080 NVIDIA GPUs on the TSUBAME 2.0 & 2.5 supercomputer, our implementation achieved 1.019 PFlops(TSUBAME 2.0) & 1.713PFlops(TSUBAME 2.5) in double precision for a large–scale problem **(CHOLESKY)** with over two million constraints.

Phase-field simulation for Dendritic Solidification [Shimokawabe, Aoki et. al.] Gordon Bell 2011 Winner
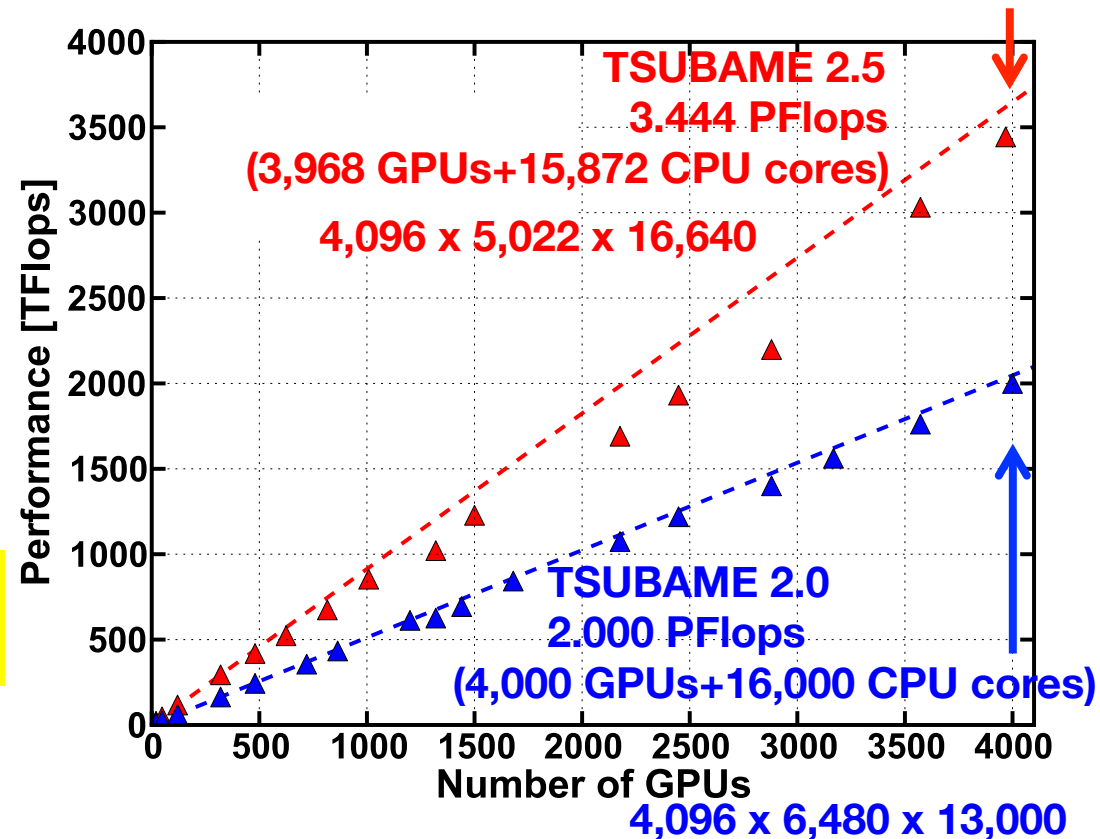
Weak scaling on TSUBAME (Single precision)
Mesh size（1GPU+4 CPU cores）:4096 x 162 x 130

Developing lightweight strengthening material by controlling microstructure

Low-carbon society

TSUBAME 2.5
3.444 PFlops
(3,968 GPUs+15,872 CPU cores)
4,096 x 5,022 x 16,640

TSUBAME 2.0
2.000 PFlops
(4,000 GPUs+16,000 CPU cores)
4,096 x 6,480 x 13,000

- Peta-Scale phase-field simulations can simulate the multiple dendritic growth during solidification required for the evaluation of new materials.

- 2011 ACM Gordon Bell Prize Special Achievements in Scalability and Time-to-Solution

Peta-scale stencil application :
A Large-scale LES Wind Simulation using Lattice Boltzmann Method [Onodera, Aoki]

Large-scale Wind Simulation for a 10km x 10km Area in Metropolitan Tokyo

10,080 x 10,240 x 512 (4,032 GPUs)

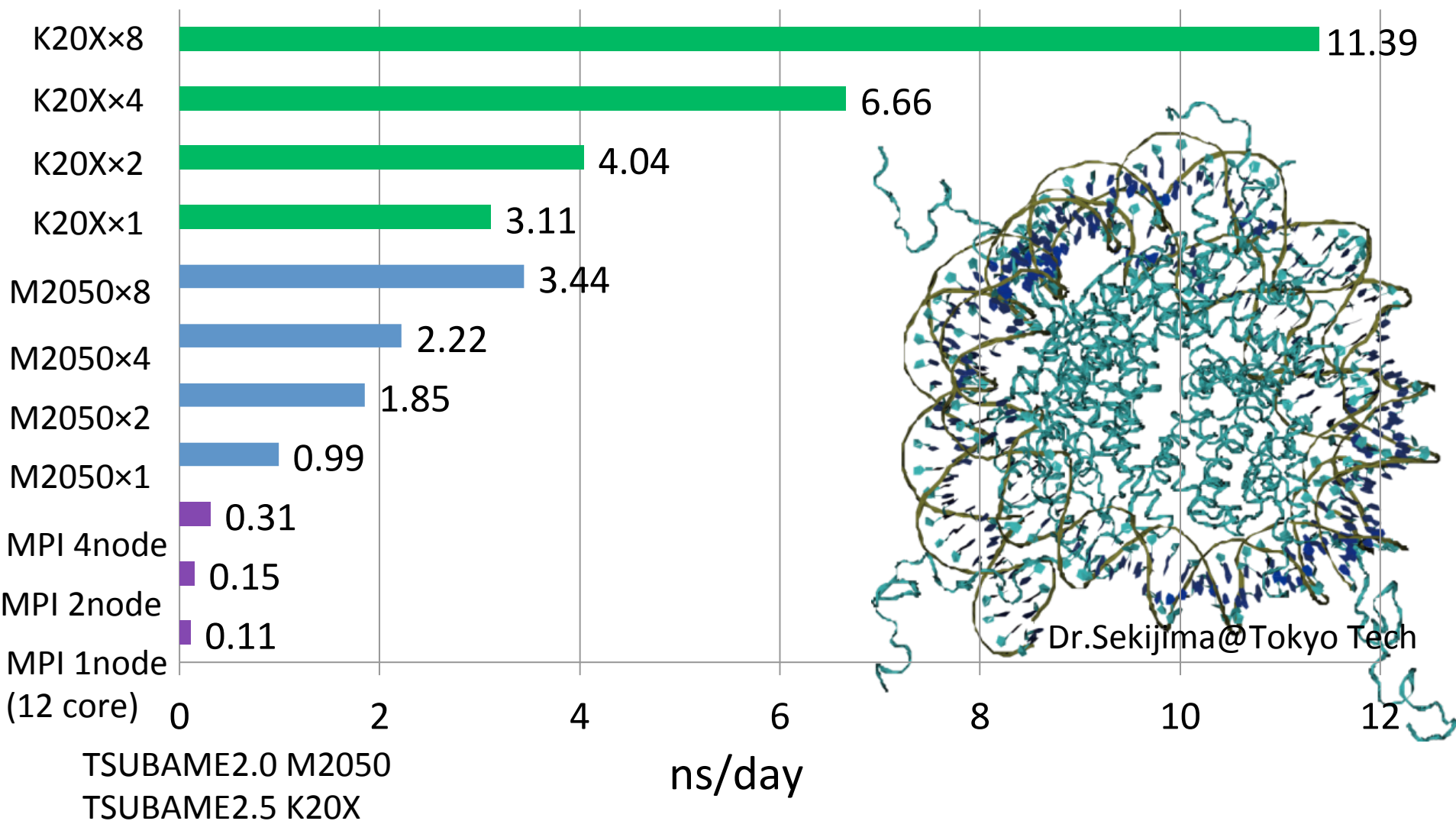The above peta-scale simulations were executed as the TSUBAME Grand Challenge Program, Category A in 2012 fall.

Weak scalability in single precision
(N = 192 x 256 x 256)

- ▲ **TSUBAME 2.5 (overlap)**
- ● **TSUBAME 2.0 (overlap)**

TSUBAME 2.5
1142 TFlops (3968 GPUs)
288 GFlops / GPU

x 1.93

TSUBAME 2.0
149 TFlops (1000 GPUs)
149 GFlops / GPU

Performance [TFlops] vs Number of GPUs

- The LES wind simulation for the area 10km × 10km with 1-m resolution has never been done before in the world.
- We achieved 1.14 PFLOPS using 3968 GPUs on the TSUBAME 2.5 supercomputer.

AMBER pmemd benchmark
Nucleosome = 25,095 atoms

	ns/day
K20X×8	11.39
K20X×4	6.66
K20X×2	4.04
K20X×1	3.11
M2050×8	3.44
M2050×4	2.22
M2050×2	1.85
M2050×1	0.99
MPI 4node	0.31
MPI 2node	0.15
MPI 1node (12 core)	0.11

ns/day

Dr.Sekijima@Tokyo Tech

TSUBAME2.0 M2050
TSUBAME2.5 K20X

GHOSTM:
A GPU-Accelerated HOmology Search
Tool for Metagenomics

Homology search is one of important
methods to annotate DNA sequences

AGTTA... → sequence
database

Search similar sequences

Data

- soil metagenomic data (SRR407548)

  150 bp, 100,000 entries

- KEGG GENES amino acid sequences

  4 GB, (May, 2013)

1 GPU + 1 CPU core

Computing time [sec]

19360.9

x1.8

10784.7

25000.0

20000.0

15000.0

10000.0

5000.0

0.0

TSUBAME 2.0   TSUBAME 2.5

MEGADOCK-GPU

Predicting protein-protein interaction network
via protein-protein docking calculations

Protein-protein interaction network is very
important to understand cell behavior and diseases.

Docking calculations for 352 pairs

3 GPUs + 12 CPU cores

Computation speed ratio [vs. 1 CPU core]

- 1 CPU core: 1.00
- 12 CPU cores: 8.93
- 12 CPU cores 3 GPUs (Tesla M2050, TSUBAME 2.0): 37.11
- 12 CPU cores 3 GPUs (Tesla K20Xm, TSUBAME 2.5): 83.94

x2.25

Application	TSUBAME2.0 Performance	TSUBAME2.5 Performance	Boost Ratio
Top500/Linpack (PFlops)	1.192	2.843	2.39
Green500/Linpack (GFlops/W)	0.958	> 2.400	> 2.50
Semi-Definite Programming Nonlinear Optimization (PFlops)	1.019	1.713	1.68
Gordon Bell Dandrite Stencil (PFlops)	2.000	3.444	1.72
LBM LES Whole City Airflow (PFlops)	0.600	1.142	1.90
Amber 12 pmemd 4 nodes 8 GPUs (nsec/day)	3.44	11.39	3.31
GHOSTM Genome Homology Search (Sec)	19361	10785	1.80
MEGADOC Protein Docking (vs. 1CPU core)	37.11	83.49	2.25

Graph500 "Big Data" Benchmark

HPCwire

Kronecker graph BSP Problem

$$\arg\max_{\Theta} P\left(\Big| \xleftarrow{\text{Kronecker}} \Theta\right)$$

A: 0.57, B: 0.19
C: 0.19, D: 0.05

$$\begin{array}{|c|c|c|} \hline 1 & 1 & 0 \\ \hline 1 & 1 & 1 \\ \hline 0 & 1 & 1 \\ \hline \end{array}$$

G_1

G_4 adjacency matrix

twitter

facebook

amazon.com

November 15, 2010
Graph 500 Takes Aim at a New Kind of HPC
Richard Murphy (Sandia NL => Micron)
" **I expect that this ranking may at times look very different from the TOP500 list. Cloud architectures will almost certainly dominate a major chunk of part of the list**."

The 4th Graph500 List (Jun2012)   TSUBAME #4 w/GPUs

		Toyotaro Suzumura, Koji Ueno, Tokyo Institute of Technology				
Rank	Installation Site	Machine	Number of nodes	Number of cores	Problem scale	GTEPS
1	DOE/SC/Argonne National Laboratory	Mira/BlueGene/Q	32768	524288	38	3541.00
1	LLNL	Sequoia/Blue Gene/Q	32768	524288	38	3541.00
2	DARPA Trial Subset, IBM Development Engineering	Power 775, POWER7 8C 3.836 GHz	1024	32768	35	508.05
3	Information Technology Center, The University of Tokyo	Oakleaf-FX (Fujitsu PRIMEHPC FX 10)	4800	76800	38	358.10
4	GSIC Center, Tokyo Institute of Technology	TSUBAME	1366	16392	35	317.09
5	Brookhaven National Laboratory	BLUE GENE/Q	1024	16384	34	294.29
6	DOE/SC/Argonne National Laboratory	Vesta/BlueGene/Q	1024	16384	34	292.36
		Pleiades, SGI ICE X, dual "sandybridge"				
8	NERSC/LBNL	XE6				
9	NNSA and IBM	NNSA/SC Blue Gene/Q	65536			

GSIC Center, Tokyo Institute of Technology
HP Cluster Platform SL390s G7
is ranked
No.4
on Graph500 Ranking of Supercomputers with
317.09 GEs on Scale 35
at the fourth Graph500 list published at the

GRAPH 500

(Tsuname2.0)

Reality: Top500 Supercomputers Dominate No Cloud IDCs at all TSUBAME2.0 #3(Nov.2011) #4(Jun.2012)
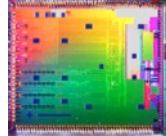
3500 Fiber Cables > 100Km
w/DFB Silicon Photonics
End-to-End 7.5GB/s, > 2us
Non-Blocking 220Tbps Bisection
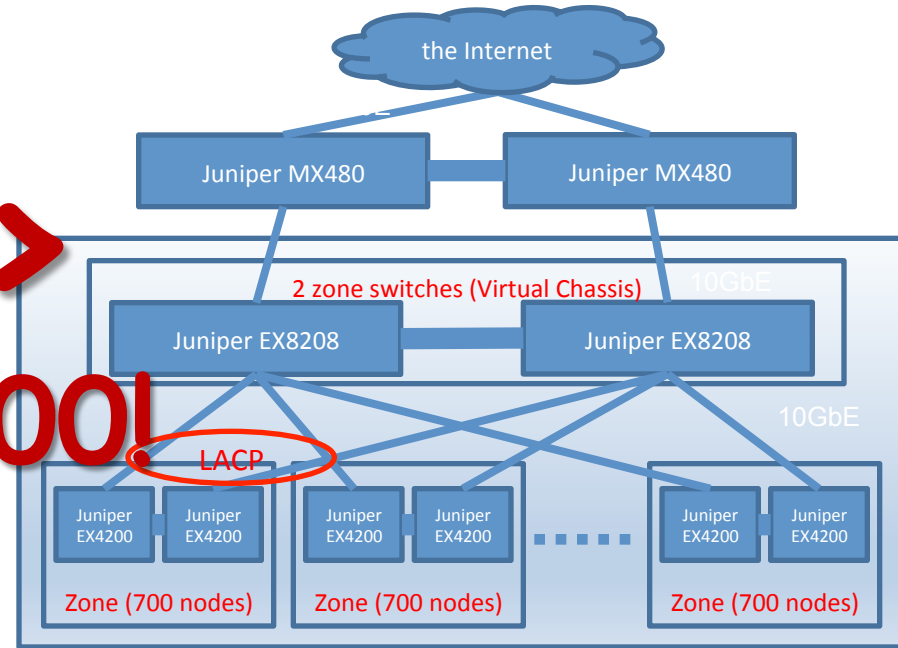
Supercomputer Tokyo Tech. Tsubame 2.0 #4 Top500 (2010)

Advanced Silicon Photonics 40G single CMOS Die 1490nm DFB 100km Fiber

~1500 nodes compute & storage
Full Bisection Multi-Rail
Optical Network
**Injection 80GBps/Node
Bisection 220Terabps**

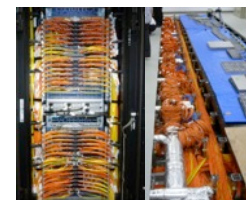**A Major Northern Japanese Cloud Datacenter (2013)**

×1000!

8 zones, Total 5600 nodes,
**Injection 1GBps/Node
Bisection 160Gigabps**

But what does "220Tbps" mean?

Global IP Traffic, 2011-2016 (Source Cicso)							
	2011	**2012**	**2013**	**2014**	**2015**	**2016**	**CAGR** 2011-2016
By Type (PB per Month / Average Bitrate in Tbps)							
Fixed Internet	23,288	32,990	40,587	50,888	64,349	81,347	28%
	71.9	101.8	125.3	157.1	198.6	251.1	
Managed IP	6,849	9,199	11,846	13,925	16,085	18,131	21%
	21.1	28.4	36.6	43.0	49.6	56.0	
Mobile data	597	1,252	2,379	4,215	6,896	10,804	78%
	1.8	3.9	7.3	13.0	21.3	33.3	
Total IP traffic	30,734	43,441	54,812	69,028	87,331	110,282	29%
	94.9	134.1	169.2	213.0	269.5	340.4	

TSUBAME2.0 Network has TWICE the capacity of the <u>Global Internet</u>, being used by 2.1 Billion users

Global Server Shipments are Flat – ~40% Capacity Growth Rate (~30% for non-HPC)

Service rate proportional to North-South bandwidth
IDC Capacity Growth CAGR thus ~30%

"Convergence" of Supercomputing and Big Data with supercomputing leadership

Source: Assessing trends over time in performance, costs, and energy use for servers, Intel, 2009.

HPC: x1000 in 10 years

CAGR ~= 100%

**IDC: x30 in 10 years
Server unit sales flat
(replacement demand)**

CAGR ~= 30-40%

Attendees:
US: 25
Europe: 11
Japan 9

Next meeting
Fukuoka, Japan
Feb. 27-28
Adjacent Big Data Workshop
Feb. 26

Exec Committee
Pete Beckman
Jean-Yves Berthou
Jack Dongarra
Yutaka Ishikawa
Satoshi Matsuoka
Philippe Ricoux

Charleston, South Carolina, USA, April 30- May 1

BIG DATA *AND* **EXTREME-SCALE COMPUTING**

http://www.exascale.org/bdec/

TSUBAME Evolution

Graph 500
No. 3 (2011)

Focused Research Towards
Tsubame 3.0 and Beyond towards Exa

- **Green Computing**: Ultra Power Efficient HPC
- **High Radix Bisection Networks** – HW, Topology, Routing Algorithms, Placement…
- **Fault Tolerance** – Group-based Hierarchical Checkpointing, Fault Prediction, Hybrid Algorithms
- **Scientific "Extreme" Big Data** – Ultra Fast I/O, Hadoop Acceleration, Large Graphs => **Convergence**
- **New memory systems** – Pushing the envelops of low Power vs. Capacity vs. BW, exploit the deep hierarchy with new algorithms to decrease Bytes/Flops
- **Post Petascale Programming** – OpenACC and other many-core programming substrates, Task Parallel
- **Scalable Algorithms for Many Core** – Apps/System/ HW Co-Design

"Software Technology that Deals with Deeper Memory Hierarchy in Post-petascale Era"

2012-2017, PI: Toshio Endo, Tokyo Tech

Growing "Memory wall" will be an obstacle to larger and fast simulations in post-petascale era

- Deeper memory hierarchy and locality improvement are keys
- Goals: ~100PB/s and ~100PB simulations on Exaflops system

Towards deeper hierarchy

Locality Improvement of Stencil Computations

In dev mem   Larger than dev mem

HMC

DDR

Next-Gen NVRAM

(NGB/s)

Flash

Trade off

Bandwidth (GB/s): 10, 100, 1000, 10000, 100000

Capacity(GB): 1, 10, 100, 1000, 10000

[GFlops]   MP-M   MMTB   MMT   MM   Usual + Naïve

size	MP-M	MMTB	MMT	MM	Usual + Naïve
240	105.63	97.12	64.34	49.57	
480	110.86	104.95	67.45	56.2	
	116.25	104.13	76.81	63.3	
720	104.58	92.06	72.67	63.97	4.49
	109.39	87.18	73.02	59.26	3.85
960				57.57	
1200	114.46	80	53.4	47.42	3.96
1440	105.93	67.64	44.18	41.69	2.81
1680	98.66	57.96	40.75	36.34	4.63
1920	89.81	50.88	31.97	29.88	4.11
2160	75.16	41.62	29.03	29.26	4.13

size of each dimension in single precision (float) format.

JST CREST "System Software for Post Petascale Data Intensive Science" (FY2011-15)

Co-PI	Institute	
Osamu Tatebe	University of Tsukuba	Project Leader
Yoshihiro Oyama	University of Electro-Communications	

- Objective
  - R&D based on scale-out file system architecture
  - Target snapshot, 100 TB/s

- Research topics
  - Scale-out distributed file system
    - Scale out to O(10K) I/O servers by utilizing access locality
    - Metadata server clustering to scale the performance out
  - Compute node OS
    - Kernel driver, process scheduling, client caching, operation offload
  - Runtime for Data-Intensive Computing
    - Efficient runtime of workflow execution, MapReduce, and MPI-IO for the scale-out distributed file system

JST CREST: Advanced Computing and Optimization Infrastructure for Extremely Large-Scale Graphs on Post Peta-Scale Supercomputers

- Innovative Algorithms and implementations
  - Optimization, Searching, Clustering, Network flow, etc.
- Extreme Big Graph Data for emerging applications
  - **2^{30} ~ 2^{42} nodes** and **2^{40} ~ 2^{46} edges**
  - **Over 1M threads** are required for real-time analysis
- Many applications on post peta-scale supercomputers
  - Analyzing massive cyber security and social networks
  - Optimizing smart grid networks
  - Health care and medical science
  - Understanding complex life system

5,000~100,000nodes (1~20MW)

HMC 32GB | DDR 100GB | NVRAM?
320GB/s
1TB/s
Vector+Scalar 10TFlops
40GB/s | ~20GB/s
Flash SSA 1TB
Interconnect | Parallel FS

---

of edges

1 trillion edges

Human Brain Project

Symbolic Network

Graph500 (Huge)

Graph500 (Large)

Graph500 (Medium)

Twitter (tweets / day)

Graph500 (Small)

1 billion edges

Graph500 (Mini)

Graph500 (Toy)

USA-road-d.USA.gr

USA Road Network

USA-road-d.LKS.gr

USA-road-d.NY.gr

1 billion nodes

1 trillion nodes

$\log_2(m)$

$\log_2(n)$

of nodes

- ■ Example: Symbolic Network
  - ■ **Human Brain Project http://www.humanbrainproject.eu/**
  - ■ Understanding the human brain is one of the greatest challenges facing 21st century science
  - ■ **89 billion neurons**(nodes)
  - ■ **1 trillion connections**(edges)
  - ■ Over 10^{17} bytes memory(storage) and 10^{18} Flops for brain simulator

K computer: 65536nodes
Graph500: 5524GTEPS

of edges

$\log_2(m)$

1 trillion edges

1 billion edges

Human Brain Project

Symbolic Network

Graph500 (Huge)

Graph500 (Large)

Graph500 (Medium)

Twitter (tweets / day)

Graph500 (Small)

Graph500 (Mini)

Graph500 (Toy)

USA-road-d.USA.gr

USA Road Network

USA-road-d.LKS.gr

USA-road-d.NY.gr

1 billion nodes

1 trillion nodes

15　　20　　25　　30　　35　　40　　45

$\log_2(n)$

of nodes

Android tablet
Tegra3 1.7GHz : 1GB RAM
0.15GTEPS: 64.12MTEPS/W

Our achievements (Super computer) : **Graph500**

Rank	1st 2010/11		2nd 2011/06
1	SCALE36 / 7.0G / 8192 node		SCALE38 / 18.47 G / 32768 node
2	SCALE32 / 5.6G / 9544 node		SCALE38 / 18.36 G / 32768 node
3	SCALE29 / 1.3G / 128 node		SCALE37 / 43.38 G / 4096 node

Rank	3rd 2011/11		4th 2012/06
1	SCALE32 / 253G / 4096 node		SCALE38 / 3541G / 32768 node
2	SCALE37 / 113G / 1800 node		SCALE35 / 508G / 1024 node
3	SCALE37 / 103G / 4096 node		SCALE38 / 358G / 4800 node
4	SCALE36 / 100G / 1366 node		SCALE35 / 317G / 1366 node

Rank	5th 2012/11
1	SCALE40 / 15363G / 65536 node
2	SCALE39 / 10461G / 32768 node
3	SCALE38 / 5848G / 15384 node
4	SCALE40 / 5524G / 65536 node

4th List

University of Tokyo FX10

TITECH TSUBAME 2.0

TITECH TSUBAME 2.0
The first implementation using many GPUs

K Computer in AICS, Japan

Twitter network (Application of Graph500 Benchmark)

Follow-ship network 2009

User j

(i, j)-edge

User i

41 million vertices and 2.47 billion edges

Our NUMA-optimized BFS on 4-way Xeon system

69 ms / BFS

⇒ **21.28 GTEPS**

Six-degrees of separation

Frontier size in BFS

with source as User 21,804,357

Lv	Frontier size	Freq. (%)	Cum. Freq. (%)
0	1	0.00	0.00
1	7	0.00	0.00
2	6,188	0.01	0.01
3	510,515	1.23	1.24
4	29,526,508	70.89	72.13
5	11,314,238	27.16	99.29
6	282,456	0.68	99.97
7	11536	0.03	100.00
8	673	0.00	100.00
9	68	0.00	100.00
10	19	0.00	100.00
11	10	0.00	100.00
12	5	0.00	100.00
13	2	0.00	100.00
14	2	0.00	100.00
15	2	0.00	100.00
Total	41,652,230	100.00	-

Extreme Big Data (EBD)

Next Generation Big Data Infrastructure Technologies Towards Yottabyte/Year

Principal Invesigator
Satoshi Matsuoka

Global Scientific Information and Computing Center
Tokyo Institute of Technolgoy

Extreme Big Data not just traditional HPC!!!
--- Analysis of required system properties

[Slide courtesy Alok Choudhary, Northeastern U]

EBE Research Scheme

Future Non-Silo Extreme Big Data Apps

Large Scale Metagenomics

Ultra Large Scale Graphs and Social Infrastructures

Massive Sensors and Data Assimilation in Weather Prediction

Co-Design Co-Design Co-Design

EBD Bag

Graph Store

EBD System Software incl. EBD Object System

Cartesian Plane

EBD KVS

NVM/ 2Tbps HBM
4~6HBM Channels
1.5TB/s DRAM &
NVM BW

NVM/

PCB

Exascale Big Data HPC

Convergent Architecture (Phases 1~4)
Large Capacity NVM, High-Bisection NW

Cloud IDC
Very low BW & Efficiency

Supercomputers
Compute&Batch-Oriented

Extreme Big Data (EBD) Team
Co-Design EHPC and EDB Apps

- **Satoshi Matsuoka (PI), Toshio Endo, Hitoshi Sato (Tokyo Tech.) (Tasks 1, 3, 4, 6)**

- **Yutaka Akiyama, Ken Kurokawa (Tokyo Tech, 5-1)**

- **Osamu Tatebe (Univ. Tsukuba) (Tasks 2, 3)**

- **Toyotaro Suzumura (Tokyo Tech. and IBM Lab, 5-2)**

- **Michihiro Koibuchi (NII) (Tasks 1, 2)**

- **Takemasa Miyoshi (Riken AICS, 5-3)**

100,000 Times Fold EBD "Convergent" System Overview

EBD System Software (Matsuoka Group)

Performance Modeling for EBD Apps

Large Scale Genomic Correlation

Large Scale Graphs and Social Infrastructure Apps

Data Assimilation in Large Scale Sensors and Exascale ...herics

Interactive Scheduler for EBD-based Analysis

Algorithm Kernels on EBD

| Indexing | Sort | Matching | Graph Search | Clustering |

EBD Programming Model

System-level Programing for EBD Object

Application-level Programing on EBD

EBD- I/O (Many-core I/O)

GPUfs

NetCDF

HDF5

TSUBAME Evolution
Towards EBD (Matsuoka Group)

Graph 500
No. 3 (2011)

Awards

25-30PF

5.7PF

3.0

Phase2
Fast I/O
5~10PB
10TB/s
1ExaB/Day

2.5

Phase1
Fast I/O
250TB
300GB/s
30PB/Day

K-Computer
10.5PF
1TB/s

No.1 line in the world

No.4   No.5   No.5   14 1

2287.6TF

2.0

25   30   41   64

No.500 line in the world

163.2TF

1.2

109.7TF

1.1

No.7   No.9   No.14

49.5TF

TSUBAME
1.0

100PF

10PF

1PF

100TF

10TF

2007   2009   2011   2013   2015H2

Preliminary I/O Evaluation on GPU and NVRAM

How to design local storage for next-gen supercomputers ?

- Designed a local I/O prototype using 16 mSATA SSDs

~320K IOPS
(3 μ sec)

- Capacity: **4TB**
- Read bandwidth: **8 GB/s**

mSATA mSATA mSATA mSATA

RAID card

Mother board

I/O performance of multiple mSATA SSD

- Raw mSATA 4KB
- RAID0 1MB
- RAID0 64KB

Bandwidth [MB/s]

mSATAs

~ 7.39 GB/s from
16 mSATA SSDs (Enabled RAID0)

I/O performance from GPU to multiple mSATA SSDs

- Raw 8 mSATA
- 8 mSATA RAID0 (1MB)
- 8 mSATA RAID0 (64KB)

Throughuput [GB/s]

Matrix Size [GB]

~ 3.06 GB/s from
8 mSATA SSDs to GPU

Target C/R strategies & Storage designs

Single-level

PFS

Multi-level

Local Storage

PFS

Synchronous

Computation

Sync ckpt

Sync ckpt

Asynchronous

Computation

Background process

Async ckpt

Async ckpt

Coordinated

P0
P1
P2
P3

ckpt

ckpt

Uncoordinated

P0
P1
P2
P3

ckpt

ckpt

ckpt

msg logging

ckpt

Flat buffer

Compute node 1
Compute node 2
Compute node 3
Compute node 4

SSD 1
SSD 2
SSD 3
SSD 4

PFS (Parallel file system)

Burst buffer

Compute node 1
Compute node 2
Compute node 3
Compute node 4

SSD 1
SSD 2
SSD 3
SSD 4

PFS (Parallel file system)

Multi-level Asynchronous C/R Model

- Compute checkpoint/restart *"Efficiency"* for *C/R strategy comparison*
  - *Efficiency*： Fraction of time an application spends only in computation in optimal checkpoint interval

$$Efficiency = \frac{ideal \;\; runtime}{expected \;\; runtime}$$

ideal runtime : No failure and No checkpoint

expected runtime : Computed by the models

$$f : (L_{i=1\dots N}, \; O_{i=1\dots N}, \; R_{i=1\dots N})$$

		Duration		
	$t + c_k$		r_k	
No failure	(k) →	$p_0(t+c_k)$ $t_0(t+c_k)$	(k) →	$p_0(r_k)$ $t_0(r_k)$
Failure	(k) → i	$p_i(t+c_k)$ $t_i(t+c_k)$	(k) → i	$p_i(r_k)$ $t_i(r_k)$

t : Interval
c_c : c-level checkpoint time
r_c : c-level recovery time
λ_i : i-level checkpoint time

$p_0(T) = e^{-\lambda T}$
$t_0(T) = T$
$p_i(T) = \frac{\lambda_i}{\lambda}(1 - e^{-\lambda T})$
$t_i(T) = \frac{1 - (\lambda T + 1) \cdot e^{-\lambda T}}{\lambda \cdot (1 - e^{-\lambda T})}$

$\begin{cases} p_0(T) & : \text{No failure for } T \text{ seconds} \\ t_0(T) & : \text{Expected time whe } p_0(T) \end{cases}$

$\begin{cases} p_i(T) & : i \text{-level failure for } T \text{ seconds} \\ t_i(T) & : \text{Expected time whe } p_i(T) \end{cases}$

- Input: Each level of
  - L_i : Checkpoint Latency
  - O_i : Checkpoint overhead
  - R_i : Restart time
- Output: *"Efficiency"*

Efficiency

Source: Sato, K., Maruyama, N., Mohror, K., Moody, A., Gamblin, T., de Supinski, B. R. and Matsuoka, S.: Design and Modeling of a Non-Blocking Checkpointing System (SC12)

Recursive Structured Storage Mode (Collaboration with DoE LLNL)

- Generalization of storage architectures with "*context-free grammar*"

  - A tier i hierarchical entity (H_i), has a storage (S_i) shared by (m_i) upper hierarchical entities (H_{i-1})
  - $H_{i=0}$ is a compute node
  - $H_N \{m_1, m_2, \ldots, m_N\}$

$i = 0$　　　　$i > 0$

Storage Model: $H_N \{m_1, m_2, \ldots, m_N\}$

r_i	Sequential read throughput from compute nodes ($H_{i=0}$)
w_i	Sequential write throughput from compute nodes ($H_{i=0}$)
m_i	The number of a upper hierarchical entities (H_{i-1}) sharing S_i

<# of C/R nodes per S_i >

||

$$\frac{K^*}{<\text{\# of } S_i > (= \Pi^N_{k=i+1} m_k)}$$

*K: C/R cluster size

Example

Flat buffer system: $H_2 \{1, 4\}$　　　Burst buffer system: $H_2 \{2, 2\}$

Experimental Setup

1 Compute node

Node 1   Node 2   ....   Node 1088

Read: 500 MB/s
Write: 260 MB/s

S_1 S_1 .... S_1

S_2

Flat buffer system: $H_2 \{1, 1088\}$

Aggregate Read: 544 GB/s
Aggregate Write: 283 GB/s

Read: 10 GB/s
Write: 10 GB/s

Checkpoint size: 5 GB/node
Logging cluster size: 16 nodes *

32 Compute node

Node 1   ...   Node 32   ...   Node 1088

Read: 16 GB/s
Write: 8.32 GB/s

S_1 ... S_1

S_2

The system sizes are based on the Coastal cluster at LLNL (88.5TFLOPS)

Burst buffer system: $H_2 \{32, 34\}$

*
Guermouche, A., Ropars, T., Snir, M. and Cappello, F.: HydEE: Failure Containment without Event Logging for Large Scale Send-Deterministic MPI Applications

56

Efficiency with Increasing Failure Rates and Checkpoint Costs

- Assuming message logging overhead is 0

- The burst buffer system always achieves a higher efficiency

  ⇒ Stores checkpoints on fewer nodes

Legend:
- Flat Buffer-Coordinated
- Flat Buffer-Uncoordinated
- Burst Buffer-Coordinated
- Burst Buffer-Uncoordinated

Y-axis: Efficiency (0 to 1)
X-axis: Scale factor (xF, xL2) — values: 1, 2, 10, 50, 100

- All systems works equally well up to x10 => TSUBAME4.0 (2020) can go exascale

- Uncoordinated checkpointing: 70% efficiency on systems two orders of magnitude larger (if logging overhead is 0)

  ⇒ Partial restart exploit the bandwidth of both burst buffers and the PFS

Phase3 Scaling up to Petabyte/s I/O EBD 2017-18
DRAM+Flash(+Processor) 100 ExaB/Day, 30 ZetaB/Year

25.6GB/s
DDR4
channels

Intel Post-Skymont, NVIDIA Post Volta, Fujitsu fx-XX, etc.

Embedded
100Gbps
(~10GB/s)

Memory switch

25.6GB/s x 3~4 =80~100GB/s (DDR4-3200)
4~6 channels=>320~600GB/s
(12~24 DIMMS per socket)
4.8 Teraflops 10W, $500?

Rack
4 cabinets/64 nodes
25TB DRAM
786TB Flash
50 TB/s DRAM BW
1.54TB/s Flash BW
1.28TB/s NW BW
384TFLops
30.7KW, $1 mil

IDC/SC
650 Racks (~ES)
41,600nodes
16PB DRAM
511PB Flash
25.6PB/s DRAM BW
1PB/s Flash BW
(x1000 K-comp HDD)
250PFlops DFP
500PFlops SFP
830TB/s NW BW
20MW, $700 million

Phase4: 2019-20 DRAM+NVM+CPU with 3D/2.5D Die Stacking
-The Ultimate Convergence of BD and EC-

2Tbps HBM
4~6HBM Channels
1.5TB/s DRAM &
NVM BW

30PB/s I/O BW Possible
1 Yottabyte / Year

NVM/Flash
NVM/Flash
NVM/Flash
DRAM
DRAM
DRAM
Low Power CPU

High Powered Main CPU

NVM/Flash
NVM/Flash
NVM/Flash
DRAM
DRAM
DRAM
Low Power CPU

TSV Interposer

PCB

Large Scale BFS Using NVRAM

1. Introduction

- Large scale graph processing in various domains
  DRAM resources has increased

- Spread of Flash Devices
  Prof : Price per bit, Energy consumption
  Cons: Latency, Throughput

Using NVRAMs for large scale graph processing has possibilities of minimum performance degradation

2. Hybrid-BFS

Switch two approaches

Top-down

$$n_{frontier} < \frac{n_{all}}{\beta}$$

Bottom-up

$$n_{frontier} > \frac{n_{all}}{\alpha}$$

of frontiers:$n_{frontier,}$ # of all vertices:$n_{all,}$ parameter : α, β

3. Proporsal

① **offload small accesses data**

② **BFS with reading data from NVRAM**

4. Evaluation (Offload Top-down Graph : we could reduce half the size of DRAM [128GB -> 64 GB] at Scale 27)

4.1GTEPS(79.4%)

5.2GTEPS

2.8GTEPS (52.9%)

Legend:
- DRAM Only
- DRAM+ioDrive2
- DRAM+Intel SSD

Y-axis: GTEPS (0.00 – 6.00)

X-axis: Swiching Parameter

β=10α β=0.1α | β=10α β=0.1α | β=10α β=0.1α | β=10α β=0.1α

α=1.E+04 | α=1.E+05 | α=1.E+06 | α=1.E+07

5. Current Work

In Bottom-up approach,
all un-visited vertex have to do is find a edge which is connected to frontier's vertex.

A lot of edges are not accessed

Each vertex allocate only a few edges to DRAM

DRAM   NVRAM

Vertices ID

v_0

v_1

v_2

━━ : outgoing edges form V_n

Simulation : Reduce Bottom-up Graph(BG), Scale 27

In-DRAM BG Size to Full BG Size

Better

Only 20~30% accesses are NVRAM

40.0%
30.0%
20.0%
10.0%
0.0%

2   4   8   16   32

Max Number of Edges Which are Allocated in DRAM

6. Related Work and Summary

● Pearce, et al. : 1 TB DRAM and 12 TB NVRAM(Fusion-io ioDrive)
              52 MTEPS [Scale 36 : 69G vertices, 1100G edges]

● We could reduce half the size of DRAM with 20.6% performance degradation
   (**4.1** GTEPS)   **[Scale 27 : 130M** vertices**,   2.1G edges]**

Roger Pearce, Maya Gokhale, Nancy M. Amato, "Scaling Techniques for Massive Scale-Free Graphs in Distributed (External) Memory"
Parallel and Distributed Processing Symposium, International, 2013 IEEE 27th International Symposium on Parallel and Distributed Processing

High Performance Sorting

Fast algorithms:

Scalability

le
icot
anana
iwi

**Efficient
implementation**

R&D of EDB Distributed Object Store (co-PI: Osamu Tatebe, U-Tsukuba)

- Key design issues for Scaled-out IOPS and I/O bandwidth
  - Scalable distributed MDS (1M IOPS)
  - High Performance local object store
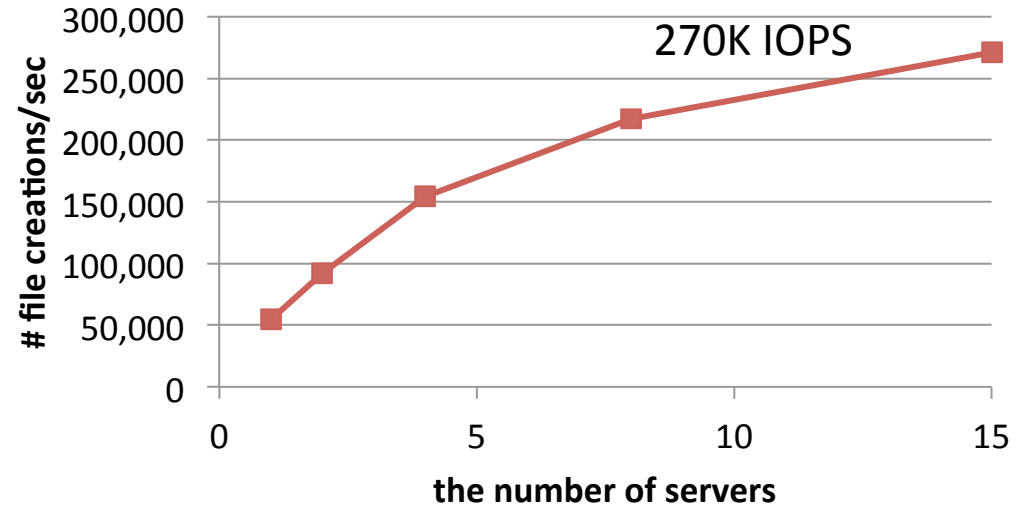 - Efficient parallel access (100 TB/s) and parallel query
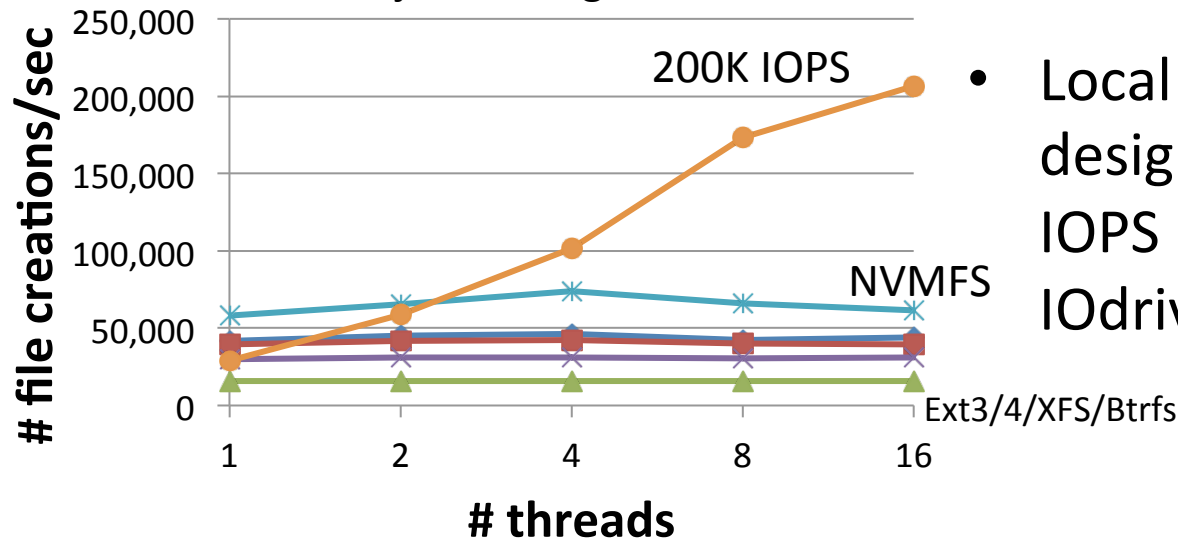
R&D of EDB Distributed Object Store

- Early distributed MDS design achieves 270K IOPS using 15 MDSs. [not published yet]

	Ours	GIGA+	skyFS	Lustre
IOPS	**270K**	98K	100K	80K
#servers (#cores)	15 (240)	32 (256)	32 (512)	1 (16)

Distributed MDS Performance

- Local Object Storage design achieves 200K IOPS using FusionIO IOdrive. [SWoPP 2013]

Local Object Storage Performance

This year's goal

- Conceptual Design of object store
  - Distributed metadata server for O(100K) clients
  - Local object store for NVRAM/Flash
  - Parallel query to maximize data locality
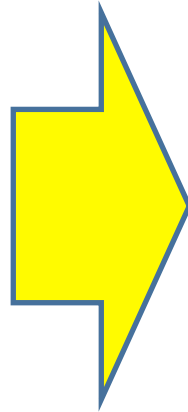
EBD Interconnect (Koibuchi Group)

Typical Data Centers
-Poor scalability
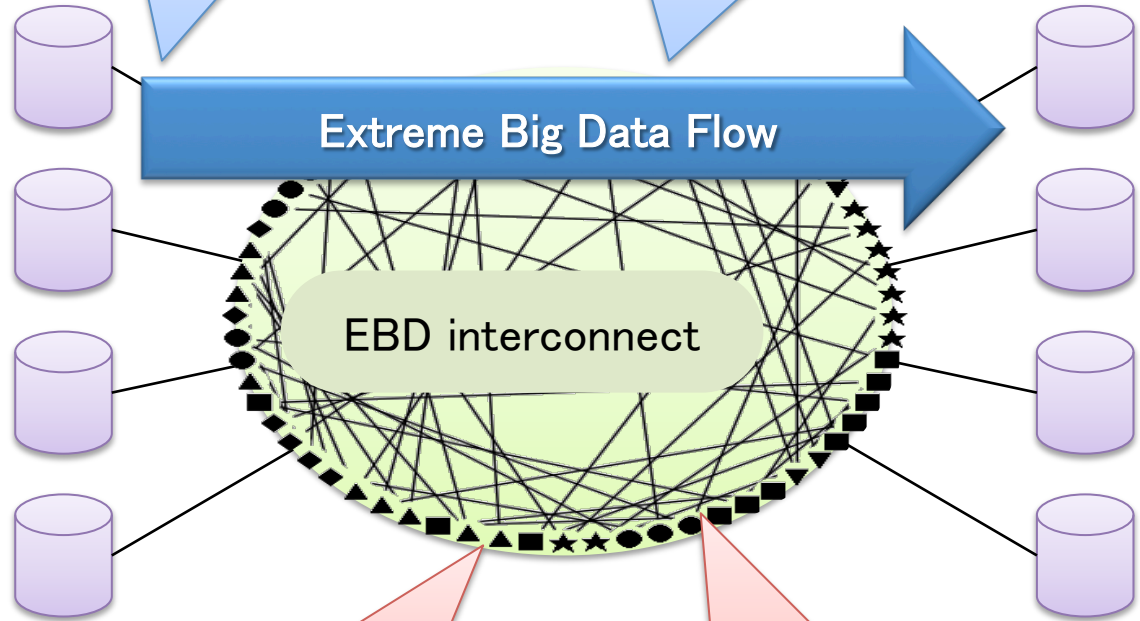- 1GbE + 10GbE
- TCP/IP basis

K Computer

Supercomputers
- Dedicated to neighboring and uniform access

EBD non-uniform access

Low latency write/read ~10 μ s for 4KB

Extreme Big Data Flow

EBD interconnect

Low-jitter topology w/ **random shortcuts**
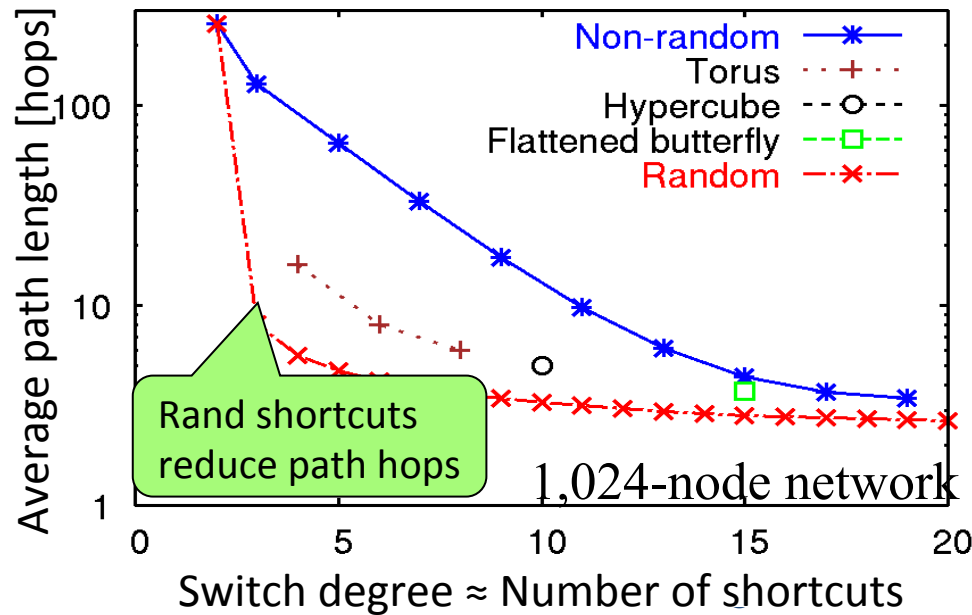
TCP/IP bypassing direct comm. to flash

Our current technology:
Rand Topology[ISCA12] Deadlock-free routing[IEEE Trans.12]
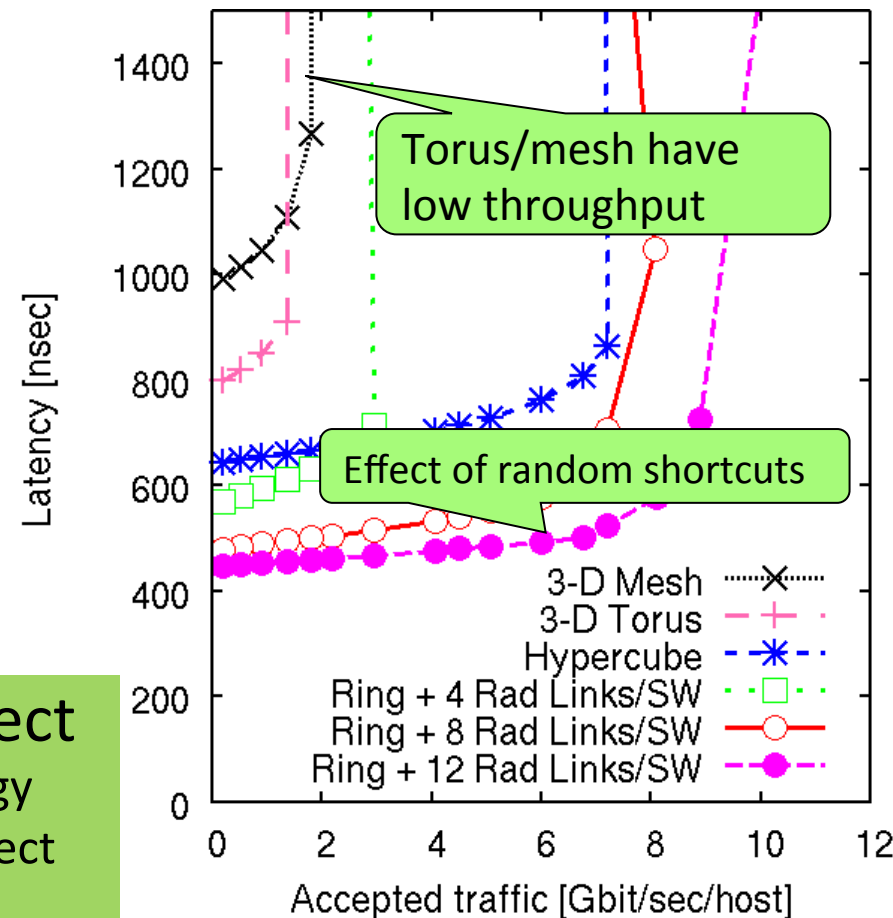Cabling Layout[HPCA13] Virtual routing method[IPDPS09]

EBD Interconnects (Cont'd)

Layout-conscious Random Topology and routing

Topology Graph Analysis

Simulation results under non-uniform shuffle access

Average path length [hops] vs Switch degree ≈ Number of shortcuts

Legend:
- Non-random
- Torus
- Hypercube
- Flattened butterfly
- Random

Rand shortcuts reduce path hops

1,024-node network

Latency [nsec] vs Accepted traffic [Gbit/sec/host]

Torus/mesh have low throughput

Effect of random shortcuts

Legend:
- 3-D Mesh
- 3-D Torus
- Hypercube
- Ring + 4 Rad Links/SW
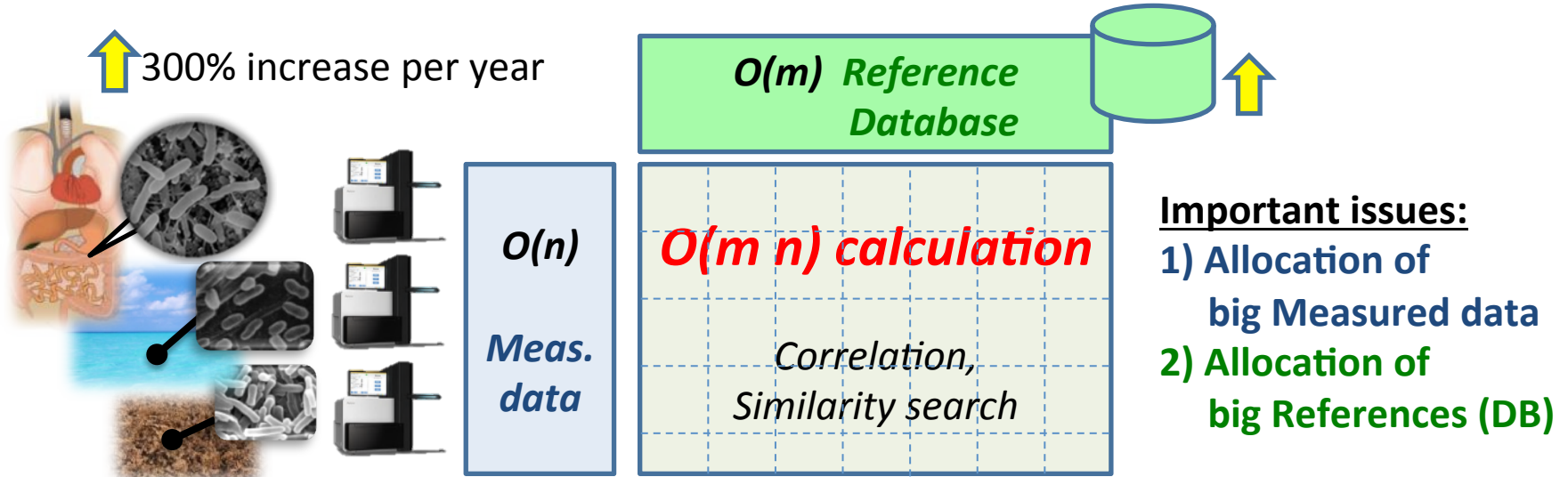- Ring + 8 Rad Links/SW
- Ring + 12 Rad Links/SW

(before) ➡ (after) EBD interconnect

- Torus/Fat tree → Random shortcut topology
- TCP/IP comm. to HDD → TCP/IP bypass direct comm. to flash
- 1G-10/20Gbps tech. → Modern InfiniBand tech.

API co-design for complicated I/O requirements
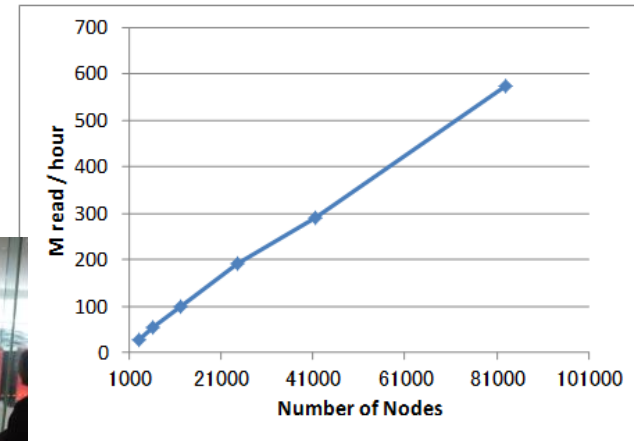（co-PI：Yutaka Akiyama, Tokyo Tech)

⬆ 300% increase per year

O(m) *Reference Database*

O(n) *Meas. data*

O(m n) calculation

Correlation, Similarity search

Metagenome sciences

Important issues:
1) **Allocation of big Measured data**
2) **Allocation of big References (DB)**

Simple batch of **BLASTX** software

3000-fold speed-up

GHOST-MP
OpenMP / MPI
load-balancing
data dispatcher

0.18 M Reads / hour
144core Xeon Cluster (2010)

572.8 M Reads / hour
82944node K-computer (2012)

M read / hour vs Number of Nodes

API co-design for complicated I/O requirements

1) Novel APIs for supporting abstraction of I/O

completely-centralized

completely-distributed

current complicated implementation on K-computer

A)

B)

C)

$10^4 \times$ Submaster i

$10^6 \times$ Worker j

simple. but need to "stage-out" millions of files

most efficient solution by now. however too much complicated.

New Idea: **"EBD bag"** (a kind of large-scale **Key-Value Store**)

Because most of results are **write-only**, and **independent** in time order

It **virtually enables completely distributed I/O programming** (B) efficiently..

2) System Evaluation through real big applications

Ultra-scale metagenome analysis, cancer genome, compound screening, etc.
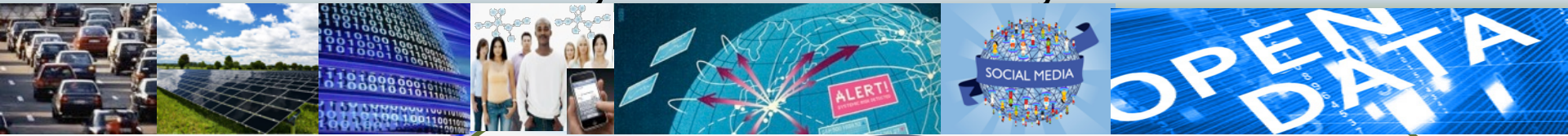
API co-design for complicated I/O requirements
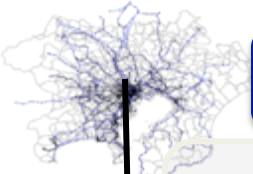
Plan for H25 (FY2013)

1. Requirement Analysis and Schematic Design for New APIs
   - ✓ EBD vs. EBD collective analysis procedures
   - ✓ Proposal of the "EBD bag" function

2. Preparation for evaluation through real big applications
   - ✓ Ultra-scale Metagenome analysis: data collection and system prototyping (ex. human oral microbiome)
   - ✓ Cancer genome analysis
   - ✓ Estimation of near-future I/O requirements in related fields (genomics, proteomics, drug design, etc.)

Suzumura Group
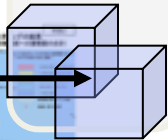# EBD Driven Planetary-Scale Social Analytics Infrastructure

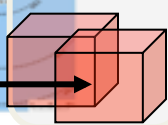**10 Tbps**
(Streaming Data including Satellite Image)

EBD-Driven Social Simulation

EBD-Driven Social Analytics

"Billion-Scale" and 10-Fold Real Time Discrete Event Simulation

Large-Scale Graph Analysis

Data Assimilation

Graph Partitioning

Centrality/BC/ BFS/RWR/ Clustering, etc

7 billion human beings on the planet with 3 billion-level road network

Grand Challenge Problem Size: 2^{42} vertices
(4.4 Trillion Vertices, 1.1 PB Memory)

Log data generation speed = 700 Tbps
Total log size per 1 simulation = 2.2 PB

Data Source

Data Source

Co-design

Supercomputer with 25 PFLOPS, 10PB (DRAM) and 511 PB (Flash), 1 Petabit/s (Comm.)
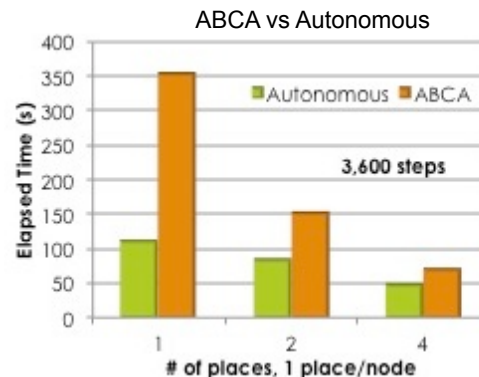
EBD Object Store

EBD Driven Planetary-Scale Social Analytics Infrastructure

A Study on Scalable Architecture and Optimization Methods for Billion-scale Social Simulation

- Motivation & Goal for 2013 and 2014: Our previous design (ABCA) cannot cope with billion-scale simulation in real-time due to tremendous amount of data and I/O, so this study is to propose the best architecture that can deal with real-time billion-scale social simulation on the future hardware designed for extremely big data processing

- Study the performance characteristics of the agent-based social simulation implementations of each candidate architecture and optimization methods
    - We started investigating from billion-scale traffic simulation

- Current status: we have completed the implementations for the first two architectures and evaluated them in million-scale simulation.

- Plan by the end of this year: Complete implementations of three candidate architectures and evaluate them in billion-scale simulation

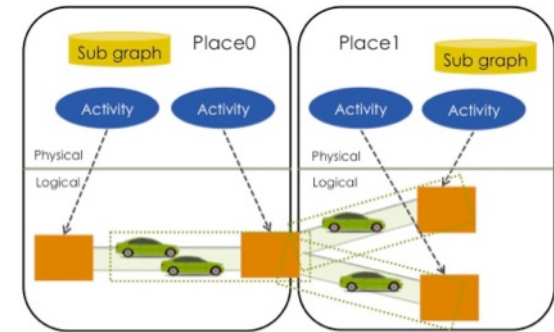- Future: we plan to make the framework more flexible to support more complicated social simulation

Agent-based Cellular Automata

Autonomous Architecture

Data Storage Architecture

ABCA architecture Optimization

- Tokyo Map (~160K cross points, ~230K roads, 46K agents)
- Tsubame S queue machine

ABCA vs Autonomous

Map: India
Cross points: 1.7M
Roads: 3.4M
Trips: 1M
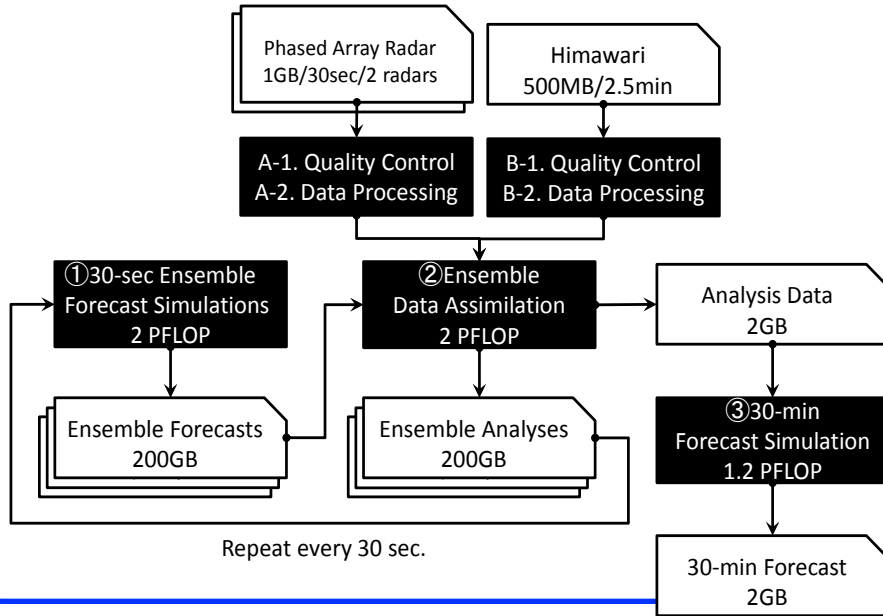
Fail-Safe EBD Workflow and Geometrical Search in Big Data Assimilation（co-PI：Takemasa Miyoshi, Riken AICS)

EBD Data Assimilation System in Weather Forecast
(Proposed simultaneously to Prof. Tanaka's CREST)

Phased Array Radar
1GB/30sec/2 radars

Himawari
500MB/2.5min

A-1. Quality Control
A-2. Data Processing

B-1. Quality Control
B-2. Data Processing

①30-sec Ensemble
Forecast Simulations
2 PFLOP

②Ensemble
Data Assimilation
2 PFLOP

Analysis Data
2GB

Ensemble Forecasts
200GB

Ensemble Analyses
200GB

③30-min
Forecast Simulation
1.2 PFLOP

Repeat every 30 sec.

30-min Forecast
2GB

4-dimensional Ensemble Kalman Filter
4D-LETKF

t_{n-1} time t_n

$\tilde{\bar{\mathbf{x}}}_a(t_{n-1}) = \bar{\mathbf{x}}_a(t_{n-1}) + \mathbf{X}_a(t_{n-1})\bar{\mathbf{w}}_a(t_n)$

$\tilde{\mathbf{X}}_a(t_{n-1}) = \mathbf{X}_a(t_{n-1})\mathbf{W}_a(t_n)$

$\bar{\mathbf{w}}_a = \tilde{\mathbf{P}}_a \mathbf{Y}_b^T \mathbf{R}^{-1}(\mathbf{y} - H(\bar{\mathbf{x}}));$

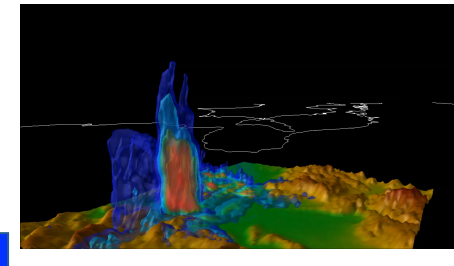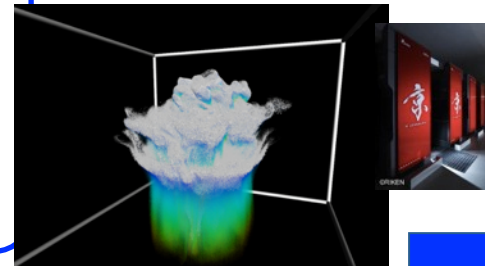$\mathbf{W}_a = [(K-1)\tilde{\mathbf{P}}_a]^{\frac{1}{2}}$

Weather Observation data keep flowing-in every 30s.

In case of hardware failure

Difficult to catch up once delayed
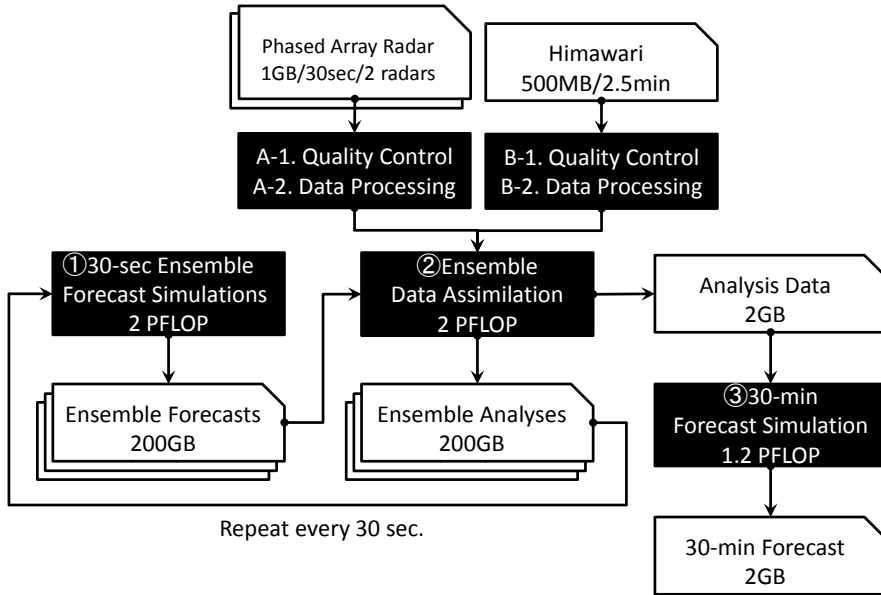
Failed Simulation

Phased Array Radar

4D-LETKF enables processing multiple steps at one time

Observations at multiple times are treated simultaneously.

Highly reliable system enabling to catch up in case of delay

Fail-safe workflow

Next-generation Data Assimilation System
(Proposed simultaneously to Prof. Tanaka's CREST)

Phased Array Radar
1GB/30sec/2 radars

Himawari
500MB/2.5min

A-1. Quality Control
A-2. Data Processing

B-1. Quality Control
B-2. Data Processing

①30-sec Ensemble
Forecast Simulations
2 PFLOP

②Ensemble
Data Assimilation
2 PFLOP

Analysis Data
2GB

Ensemble Forecasts
200GB

Ensemble Analyses
200GB

③30-min
Forecast Simulation
1.2 PFLOP

Repeat every 30 sec.

30-min Forecast
2GB

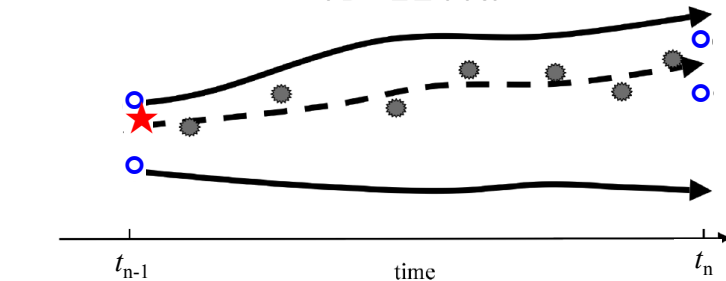**Observation data keep flowing-in every 30 sec.**

In case of hardware failure

It is hard to catch up once it gets delayed

4D-LETKF enables processing multiple steps at one time

4-dimensional Ensemble Kalman Filter
4D-LETKF

t_{n-1} time t_n

$\tilde{\mathbf{x}}_a(t_{n-1}) = \overline{\mathbf{x}}_a(t_{n-1}) + \mathbf{X}_a(t_{n-1})\overline{\mathbf{w}}_a(t_n)$
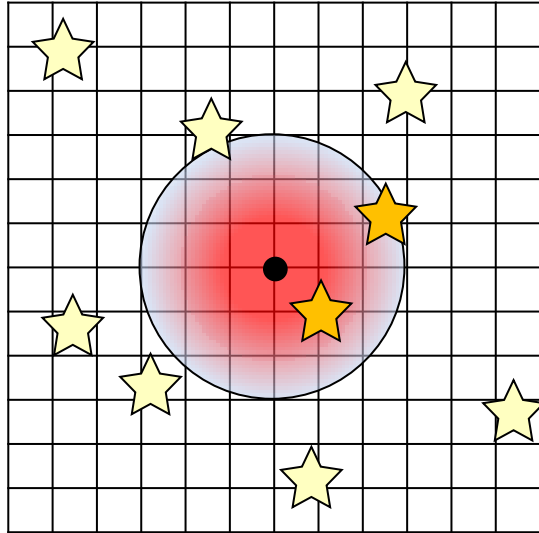
$\tilde{\mathbf{X}}_a(t_{n-1}) = \mathbf{X}_a(t_{n-1})\mathbf{W}_a(t_n)$

$\overline{\mathbf{w}}_a = \tilde{\mathbf{P}}_a \mathbf{Y}_b^T \mathbf{R}^{-1}(\mathbf{y} - H(\overline{\mathbf{x}}));$

$\mathbf{W}_a = [(K-1)\tilde{\mathbf{P}}_a]^{\frac{1}{2}}$

Observations at multiple times are treated simultaneously.

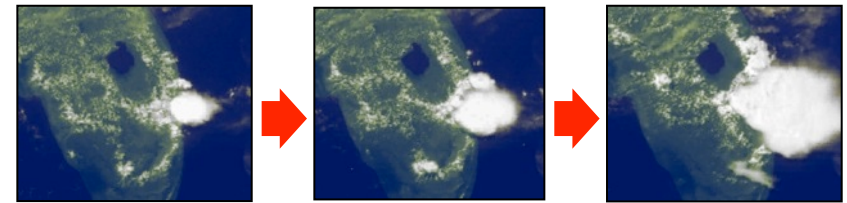**Highly reliable system enabling to catch up in case of delay**

Optimizing LETKF local search

LETKF (Local Ensemble Transform Kalman Filter) includes geographical search of nearby observations around each grid point on Earth

Search $O(10^3)$ nearby observations out of total $O(10^6)$ at each of $O(10^7)$ grid points.

About half of the total LETKF computer time

Co-design of Hardware and Software
Optimizing the spatial search algorithm suitable for the converged EBD architecture

●親水公園で水遊び

増水直前　　　　　　増水時

International Collaborators and Potential Industries

Alok Chaudhary
Professor, Northwestern U
Big data performance
and benchmarking

Rick Stevens
Associate Laboratory Director,
Argonne National Laboratory
Convergence Architecture

Robert Ross
Math. and Computing Sciences,
Argonne National Laboratory
Distributed Big Data Objects

Graph and Big Data
Benchmarking

Gabriel Antoniu
Scientific leader of the KerData
research team at INRIA Rennes
Distributed Filesystems