Booth Display & Demonstration @ #759

Katsuyuki Hasebe

The University of Tokyo
&
National Institute of Information and Communications Technology

Booth Display & Demonstration

- MaSTER-1
  - 5-port 10GbE Testbed
- High-Speed Filecopy System & Dynamic pacing with TCP Congestion Control
- GRAPE-DR processor chip/system
- BWC Paticipate

MaSTER-1

5-port 10GbE Testbed

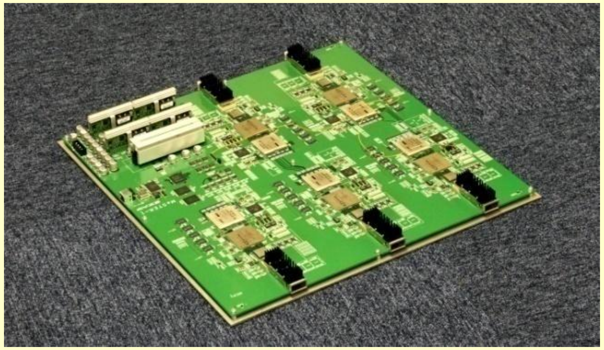
MaSTER-1 Overview

•10GbE (LAN PHY) experimental testbed
•5 XFP ports connected to FPGAs through MACs
•All FPGAs are connected all-to-all at the speed of 10 Gbps
•FPGAs can be communicated with a control PC thorough a USB port

MaSTER-1 Advantages

•Packets input from a port can be processed at the speed of 10 Gbps
•Packets can be output from any ports
•Each port has powerful configurable FPGA
•Each port has large memories to store packets

MaSTER-1 (12 Layers PCB)



MaSTER-1 Applications

・Programmable 10GbE switch
  •Currently running
•Packet Filters
•Packet Logger
•Pseudo Long Fat-pipe Networks
  •Maximum 400 ms delay

MaSTER-1 improves the performance of Parallel TCP Streams

•Multiple ports allow MaSTER-1 to handle multiple connections without switches
•MaSTER-1 can observe directly the packets transmitted by end hosts
•MaSTER-1 will clarify the problems with the method for dropping and merging packets in 10GbE switches
• MaSTER-1 is a good tool for verifying the performance of parallel TCP streams on Long Fat-pipe Networks (LFNs)

MaSTER-1

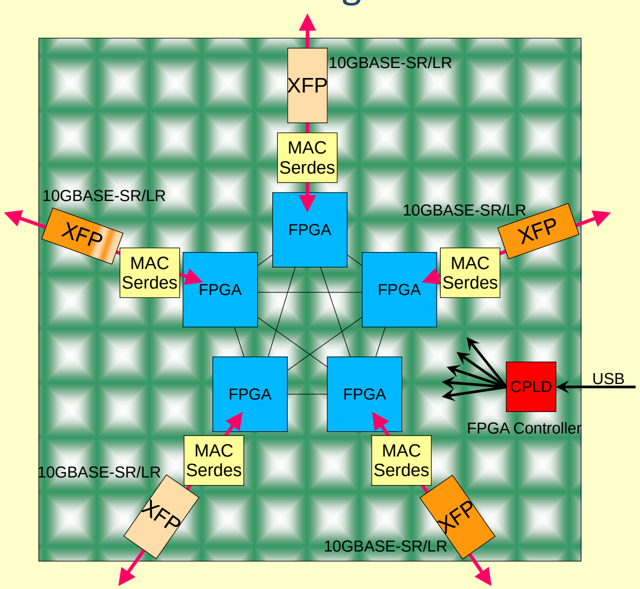
5-port 10GbE Testbed

MaSTER-1 Specification

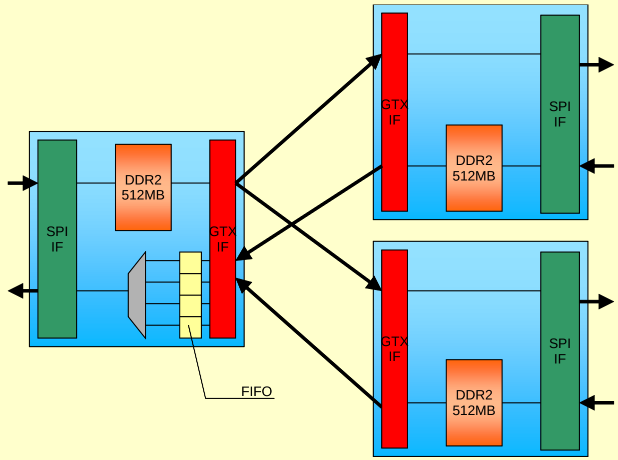
Ports	5 XFPs (10GBASE-SR/LR)
Processors	5 FPGAs (Xilinx XC5VFX70T-1FF1136) – 1 for each port
Interconnection	All-to-all 10 Gbps – Xilinx Rocket IO
Memory	DDR2 SDRAM 2.56 GiB – 512 MiB for each processor
I/O	USB 2.0
Dimension	430 mm x 430 mm x 50mm (WDH)

MaSTER-1 in a Rack

MaSTER-1 Block Diagram



MaSTER-1 FPGA Interconnections



MaSTER-1

Merging Stream Harmonizer

- Bursts in parallel TCP streams on Long Fat-pipe Networks
- Needless packet losses occur by burst overlaps on network switch
- Imbalance between streams

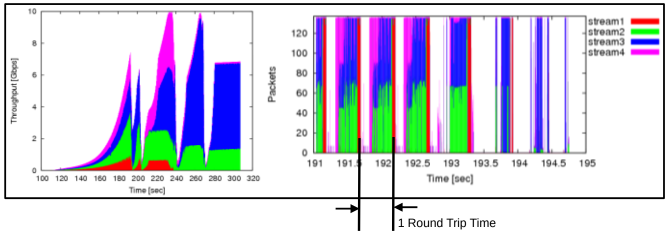
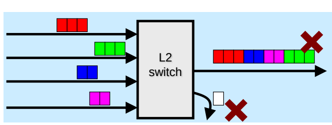
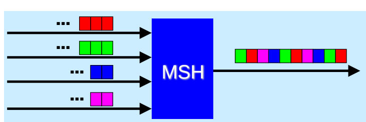


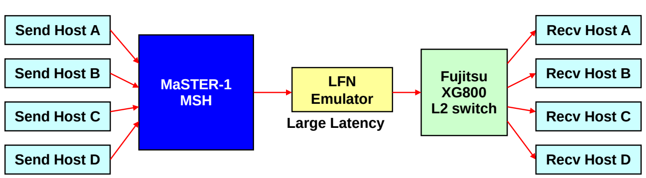
Fig. Four parallel TCP streams merged at conventional switch.
Left: Throughput / Right: Packet per sec (pps)



- A network switch receives packets of multiple streams
- The switch outputs the streams without eliminating bursts
- Packet losses occur at the switch



- Scatter and disperse streams at packet level
- Select the packets with round-robin algorithm
- Output each stream with packet scheduling



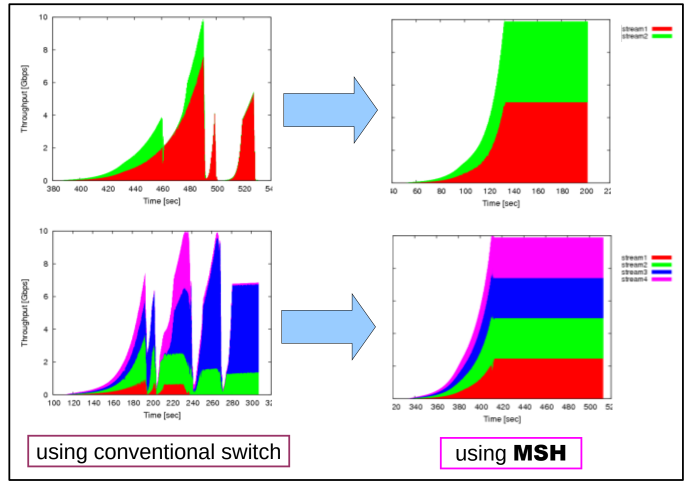
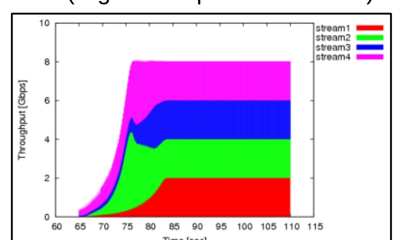


Fig. Experimental network
LFN Emulator (10 Gbps / 250 × 2 = 500 ms round trip time)
Use MSH as a substitute for merging L2 network switch

- MSH improved throughput decrease by eliminating packet losses
- MSH balanced 4 streams
- All streams can get throughput, almost 2.5 Gbps each

◆ Summary : MSH achieved

- Merging Stream Harmonizer (MSH) has large buffers and fine-grained scheduling
- multiple streams are stabilized and balanced on pseudo and real LFNs (Right: 8 Gbps on real LFNs)



NICT

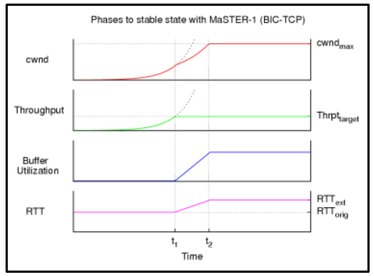
THE UNIVERSITY OF TOKYO

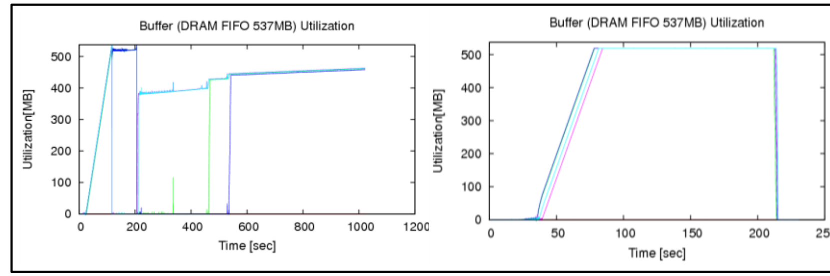
MaSTER-1

Merging Stream Harmonizer

Delay on bottleneck switch

- At a *bottleneck switch*, packets are stored in queues
  - *bottleneck switch*: bandwidth of the input is larger than that of the output
- When MSH paces streams to low bandwidth, MSH functions as *bottleneck switch*
- When there are packets in the queue, RTT increases





$$cwnd_{max} = util + thrpt \times RTT_{orig}$$

$$RTT_{ext} = cwnd_{max} / thrpt$$

- $cwnd_{max}$ = maximum size of congestion window
- util = buffer utilization on Merging Stream Harmonizer (MSH)
- thrpt = target throughput (MSH paces throughput at **thrpt**)
- RTT_{orig} = Round Trip Time of the network (original RTT)

Fig. In the situations of constant $cwnd_{max}$ (600MB), lower throughput demands larger buffers as queue
At 2 Gbps pacing, the buffer utilization is 507 MB (Right)
At 1 Gbps pacing, more than 537 MB is required, and buffer overflows occur (Left)

High-Speed Filecopy System & Dynamic pacing with TCP Congestion Control

1. Feature

Fast
- Max 7 Gbps file copy using TCP/IP over LFN.

Low-cost
- Commodity hardware.
  Total cost is about $5,000.

Small
- about 10 kg, two systems can be contained in one suitcase.

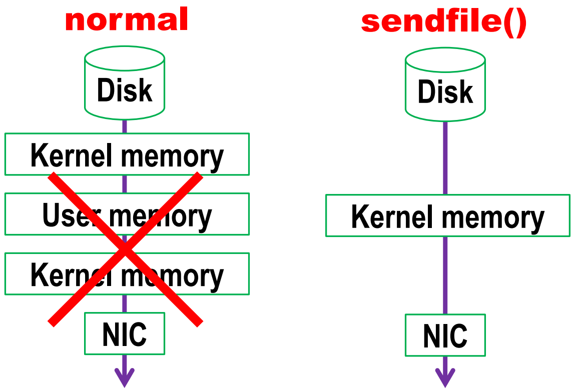
Easy-to-Use
- CUI / HTTP, optimized Apache & Firefox

2. Specification
- Intel Core i7 920 (2.93 GHz)
- 6 GB DDR3 SDRAM
- ASUS Rampage II GENE (X58, MicroATX)
- Chelsio S310E-CR
- Adaptec ASR-51245
- Intel X25-E 32 GB x 6 (RAID0)

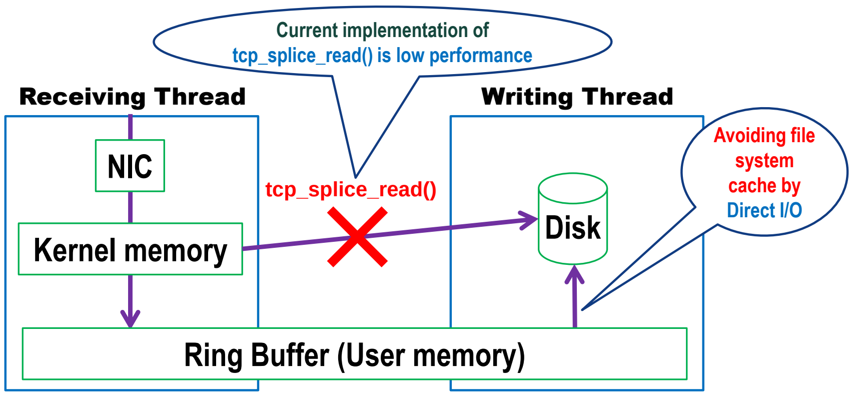
High-Speed Filecopy System & Dynamic pacing with TCP Congestion Control

3. Implementation

- Sender



- Receiver



Dynamic Pacing with TCP Congestion Control

1. Objective

- ✓ Optimize TCP throughput on shared network whose available bandwidth is not constant
- ✓ Keep TCP friendliness
- ✓ End-to-end software implementation (NO special hardware)

2. Method

- ✓ Pacing on MAC-layer controlled by device driver
- ✓ Pacing throughput is determined by TCP congestion control & available bandwidth estimation

Ultra-Speed file-Acquisition-system over Distance with Apache and fireFOX

Background

Bandwidth-Utilization of DR (1 to 1 transfer)

- Memory-memory transfer: 99%
- Disk-Disk 1server, 2 streams, RAID: 90%
- Disk-Disk 1 stream, 1 stream, RAID: 80%

However

- **Download via Web system: less than 10%**

Main Causes of performance decrease

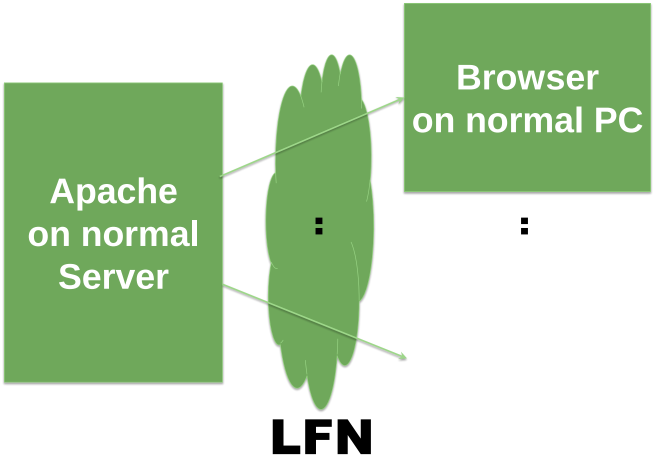
- ✓ Overhead of mem-copy & useless operation
- ✓ Inappropriate TCP tuning (Buffer Size, etc.)
- ✓ I/O Bottleneck (Harddisk/RAID)

Our Solution

- ✓ Modify Apache and Firefox
- ✓ Use Data-Reservoir Technology [Yoshino+ 2008]
- ✓ User SSD RAID0 (striping) system

Our Goal

Attain 80 % utilization on popularly used system



Bring our DR technology in general use!

Ultra-Speed file-Acquisition-system over Distance with Apache and fireFOX

Implementation Details

- **HTTP Server (Apache)**
  - Minimum modification for compatibility
    - Never modify source code
    - Omit unnecessary options for build
  - TCP tuning (DR-technology)
    - Adjust MTU, IPG, Buffer size, etc.
- **Client-Browser (modified Firefox)**
  - Optimize data operations
    - Reduce mem-copy & Omit useless operation
    - Adjust buffer size in Firefox
  - I/O optimization
    - Use mmap with madvice
    - EXT3 with write back
    - Background Frequent File-Cache flush

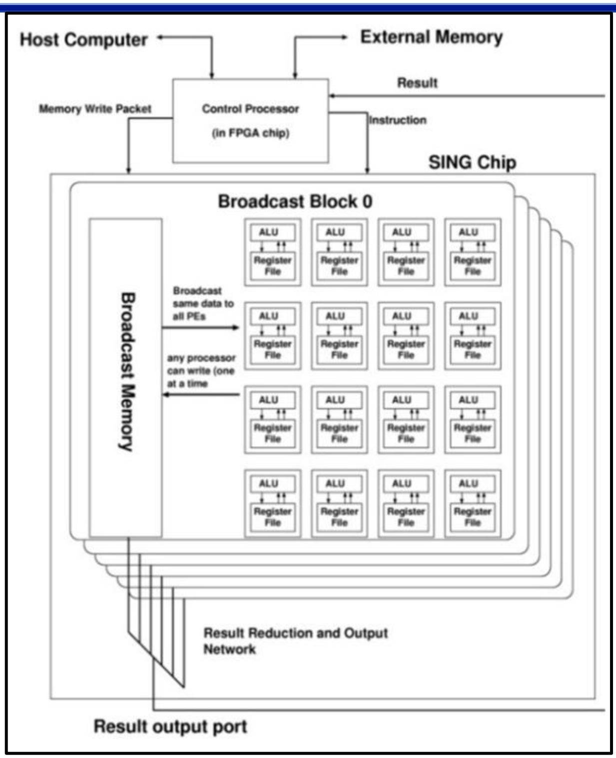
Evaluation

- **On 10G-LAN in laboratory (no delay)**
  - Non-modified Firefox : 1.7 Gbps
  - USADAFOX : **7 Gbps**
- **On pseudo LFN in laboratory (delay: 200ms)**
  - Non-modified Firefox : 3 Mbps
  - USADAFOX : **6.5 Gbps**
- **On Real LFN between Tokyo and Portland**
  - Non-modified Firefox : xxx Mbps
  - USADAFOX : x.xx Gbps

Specification of server and client

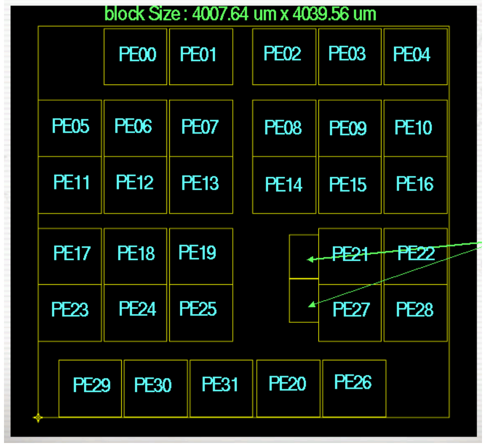
- Intel Core i7 940 3.00 GHz
- 6GB DDR3 SDRAM
- Chelsio S310E-CR
- Adaptec ASR-51245
- Intel X25-E 32 GB x 6 (RAID0)
- CentOS 5.3 (kernel: 2.6.18.128.el5)

GRAPE-DR Processor



- A "Many-core" SIMD processor with 512 processing elements (PEs)
- 512 PEs organized into 16 "Broadcast blocks"
- Fabricated with TSMC 90 nm process
- Chip size 18 mm x 18 mm
- Around 200 M transistors
- 500 MHz clock, 512 single-precision Gflops
- 256 Double-Precision Gflops
- 65 W power consumption
- Instruction bus, data input bus, data output bus

The "Broadcast block" (BB)

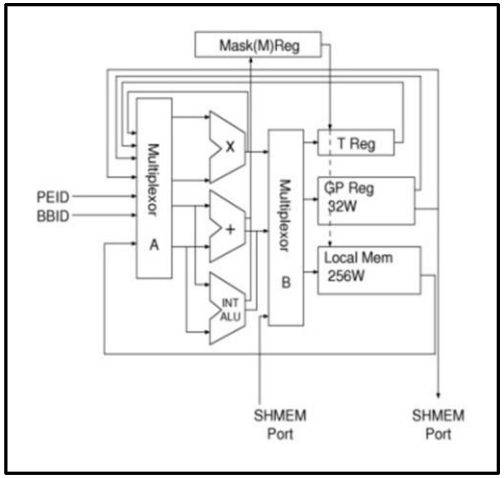


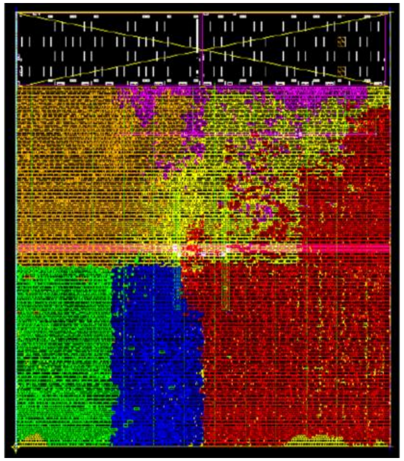
- Consists of 32 PEs and one "Shared" memory
- The shared memory can broadcast data to all PEs in the block
  Only one PE can write data to the shared memory at a time
- All BB receives the same data and same instruction from an off-chip control unit
- Data output is through programmable data-reduction network (summation, logic operation, max/min etc)

GRAPE-DR Processor

The processing element

- Double-Precision Add/Sub unit with throughput 1
- Double-Precision Multiplier unit with throughput 1/2, works also as Single-Prec. Multiplier with throughput 1
- A tri-port, 32-word register file
- A dual-port aux register
- A single-port, 256-word memory





The GRAPE-DR processor is optimized to

- ✓Computation-intensive applications
- ✓Calculation of particle-particle interactions in particle-based simulations
- ✓Dense-matrix operations
- ✓Quantum-chemistry applications

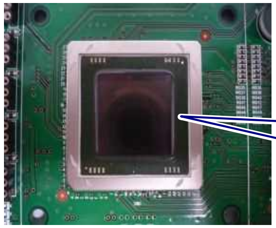
Importance of on-chip data reduction network

- ✓Particle-particle interactions – reduce necessary number of particle to achieve peak performance
- ✓Matrix operations – minimize the necessary matrix size to achieve the peak performance

GRAPE-DR board and system

GRAPE-DR board

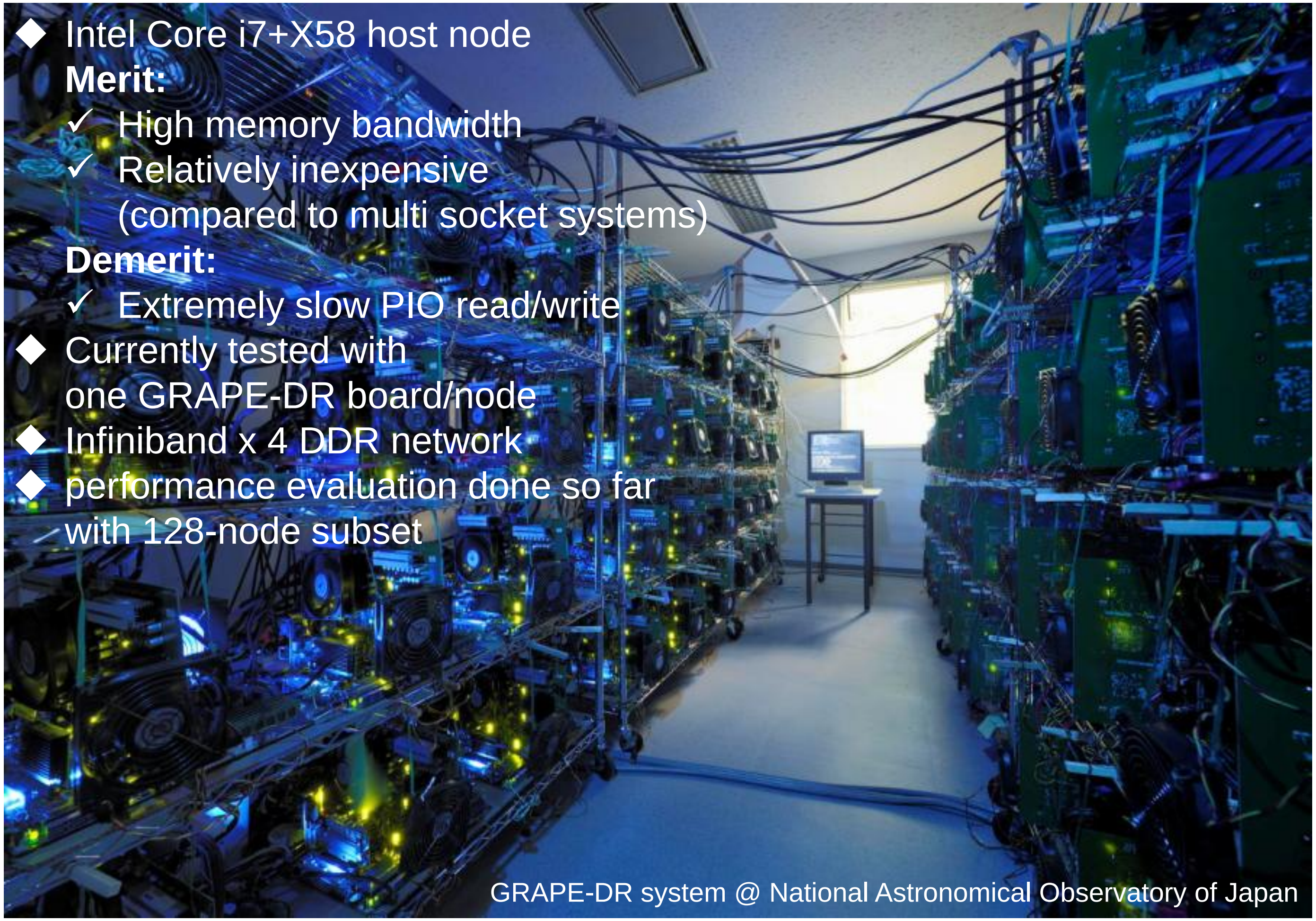
- 4 GRAPE-DR chips, each with FPGA control processor and DDR2 DRAM
- Total memory bandwidth 16 GB/s
- Theoretical I/O bandwidth: 4 GB/s bidirectional
- Measured I/O bandwidth: 2.5 GB/s write, 1.6 GB/s read
- Summation and other reduction operation can be done over multiple PEs



GRAPE-DR chips
Commercial version available from KFCR (www.kfcr.jp)

GRAPE-DR board and system

GRAPE-DR system – current setup



- Intel Core i7+X58 host node

  Merit:
  - ✓ High memory bandwidth
  - ✓ Relatively inexpensive (compared to multi socket systems)

  Demerit:
  - ✓ Extremely slow PIO read/write
- Currently tested with one GRAPE-DR board/node
- Infiniband x 4 DDR network
- performance evaluation done so far with 128-node subset

GRAPE-DR system @ National Astronomical Observatory of Japan

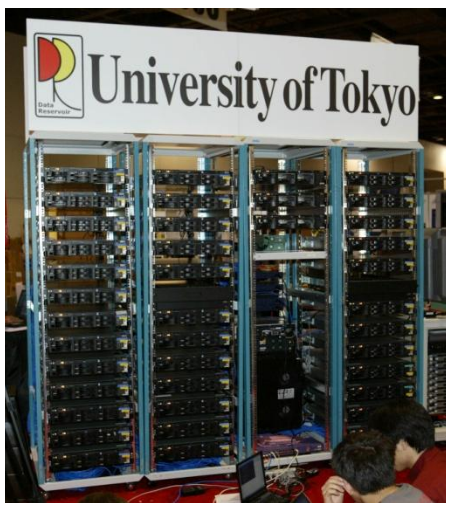
Generations of Data Reservoir

System for long-distance disk to disk data-transfer

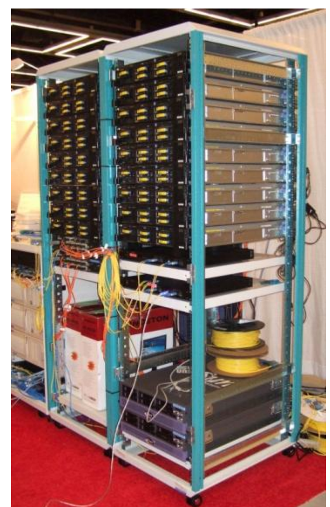
- 1st Generation – 26 servers, 26 disks for 500Mbps
- 2nd Generation – 16 servers, 64s for 10Gbps
- 3rd Generation – 8 servers, 32 disks for 10Gbps
- 4th Generation – 1 server, 32 disks for 10Gbps
- **5th Generation – 1 CPU PC, 6 SSDs for 10Gbps**

5th generation 2009

1st generation 2001



2nd generation 2003



3rd generation 2005

4th generation 2006

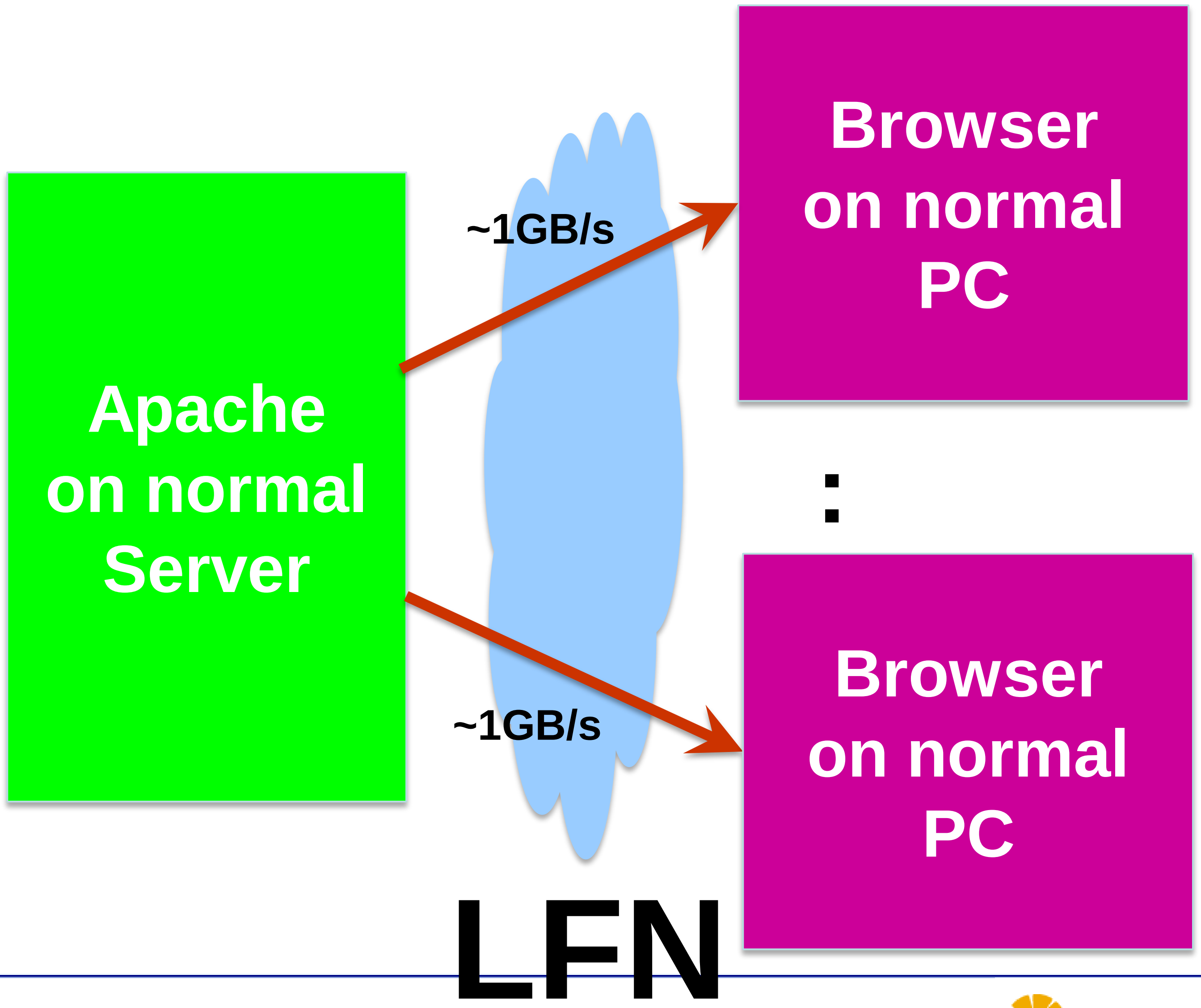
5^{th} Generation Data Reservoir

- Our Dream System of 2001
  - Key technology for utilizing 10Gbps by everyone
- Small, cheap, robust file to file transfer facility
  - **As small as we can carry as a carry on bag of a flight**
  - As cheap as a regular PC
  - Fully utilize 10Gbps long-distance network
  - Stable and robust to use on a shared internet
- IPv6
  - **We already showed IPv6 performance on latest LSR**
  - Both remote file accesses and Web accesses can be done on IPv4 and IPv6

We have more this year

- **Very high speed Web serice**
  - **Apache** server on a cube PC (Univ. of Tokyo)
  - Modified **Firefox** web client on a cube PC (Portland)
  - Linux OS, 6 x SSDs (RAID 0)
  - Http data transfer on single TCP stream
    - Zero-copy EXT3 file system
- Dynamic Pacing for robust use of Shared networks
  - (for BWC, this capability is off)
  - Estimation of available shared bandwidth by probing
  - Pacing by estimated BW
  - Control of traffic and removal of bursty behavior

Data Reservoir technology to Web service



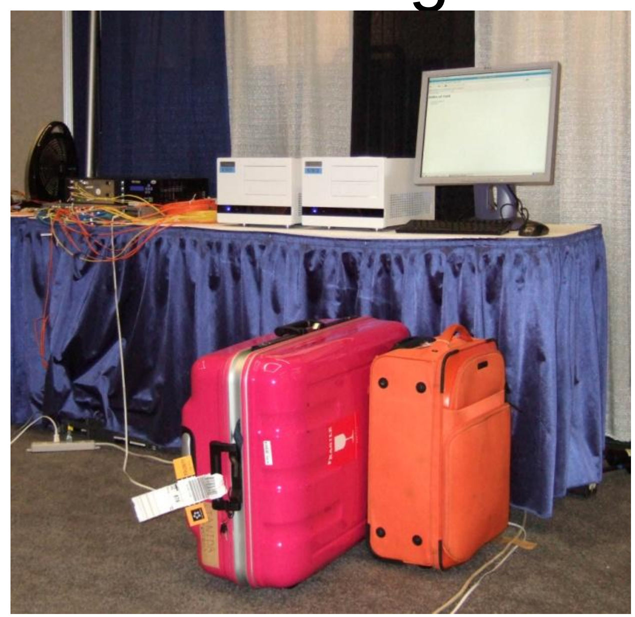
Server and Client

- **Same Nehalem PC**
  - **6 SSDs (Intel X25) support 10 Gigabit network**
  - **Single CPU (Intel Corei7)**
  - **Adaptec RAID card**
- Full bandwidth with access through a file-system

Apache Server in Univ. of Tokyo

Anue Delay Emurator

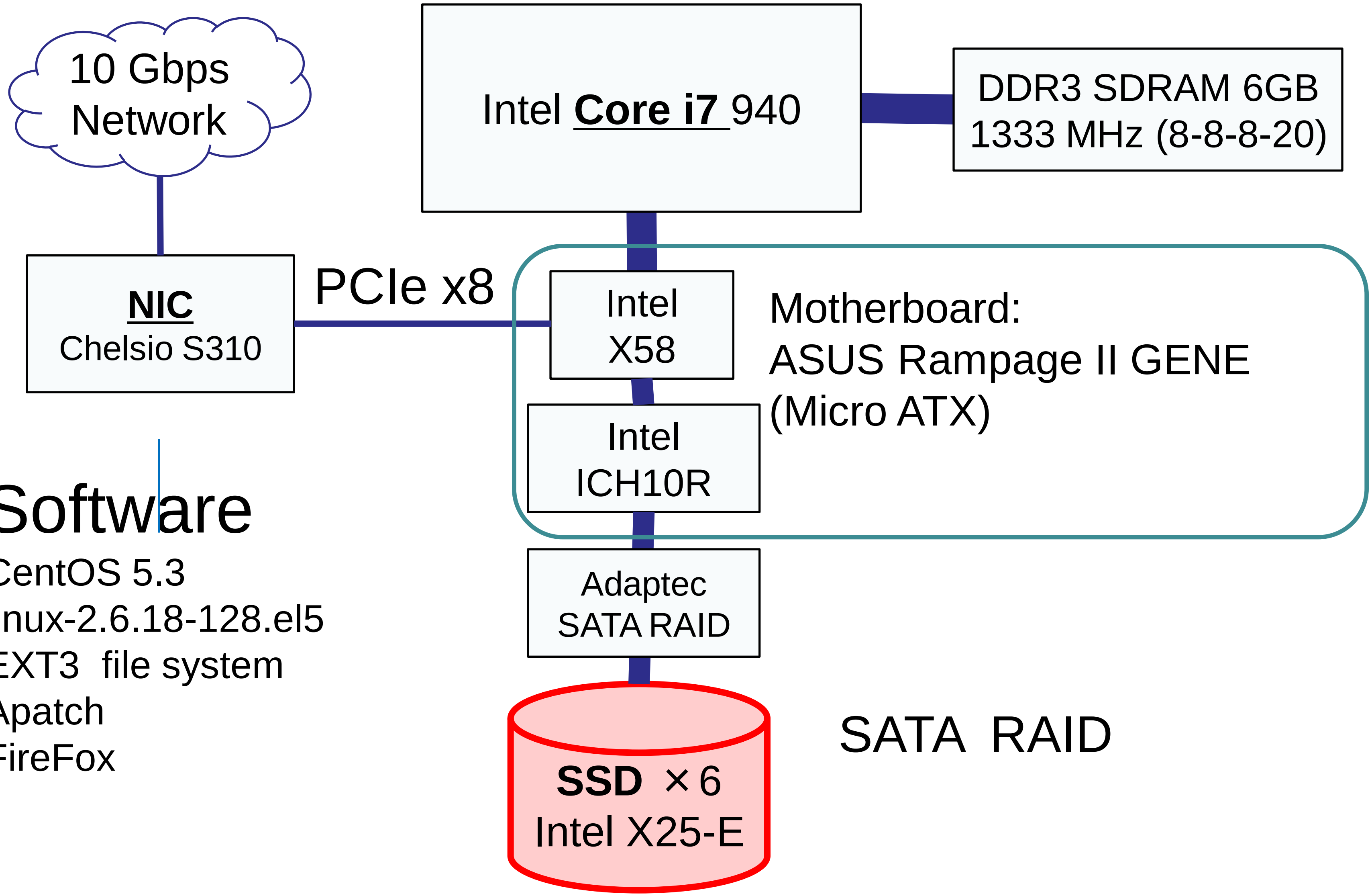
ARISTA 10G SW



Client PC

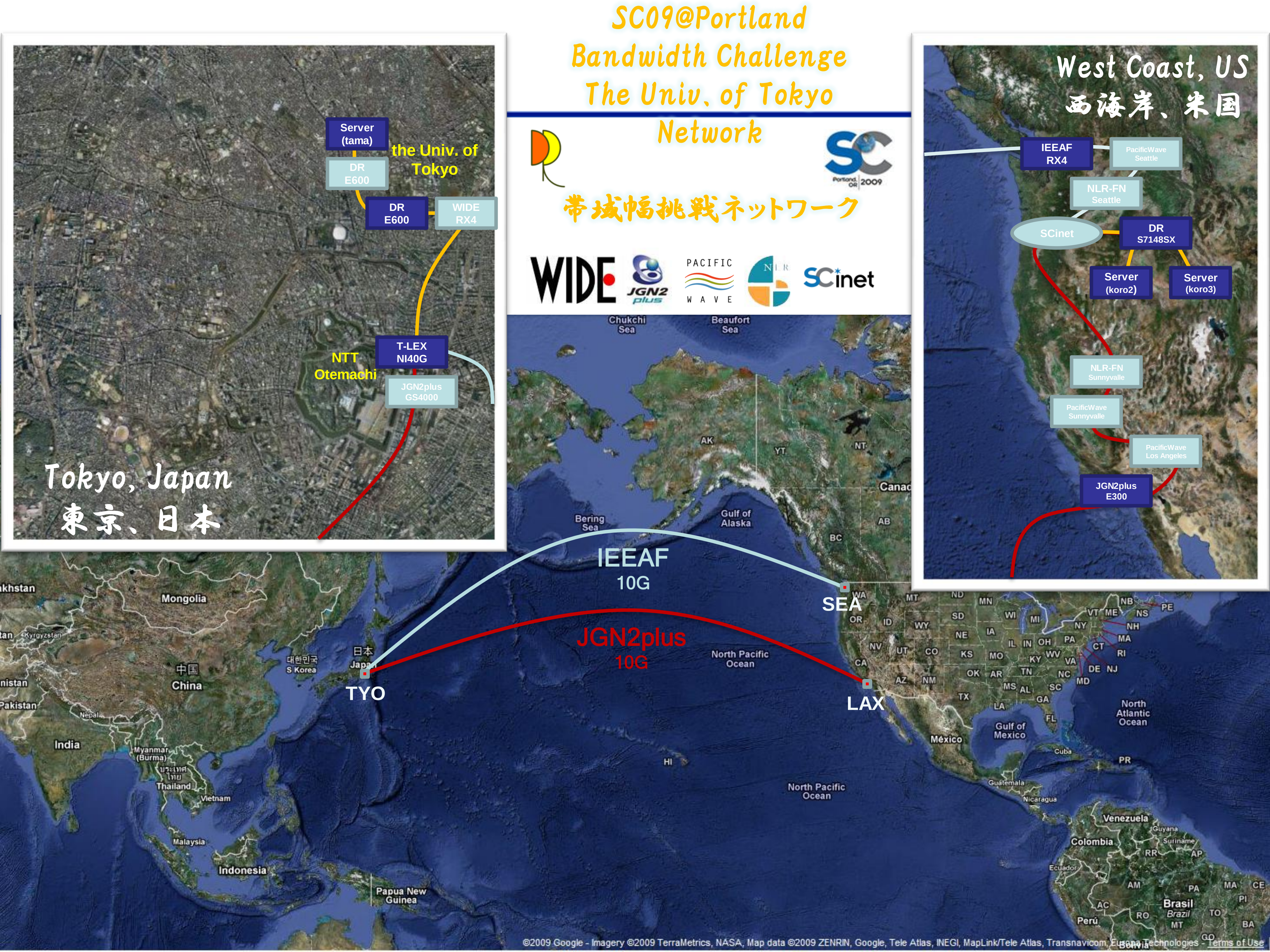
Suitcase for PC and fiber

FireFox Clients at SC09, Portland

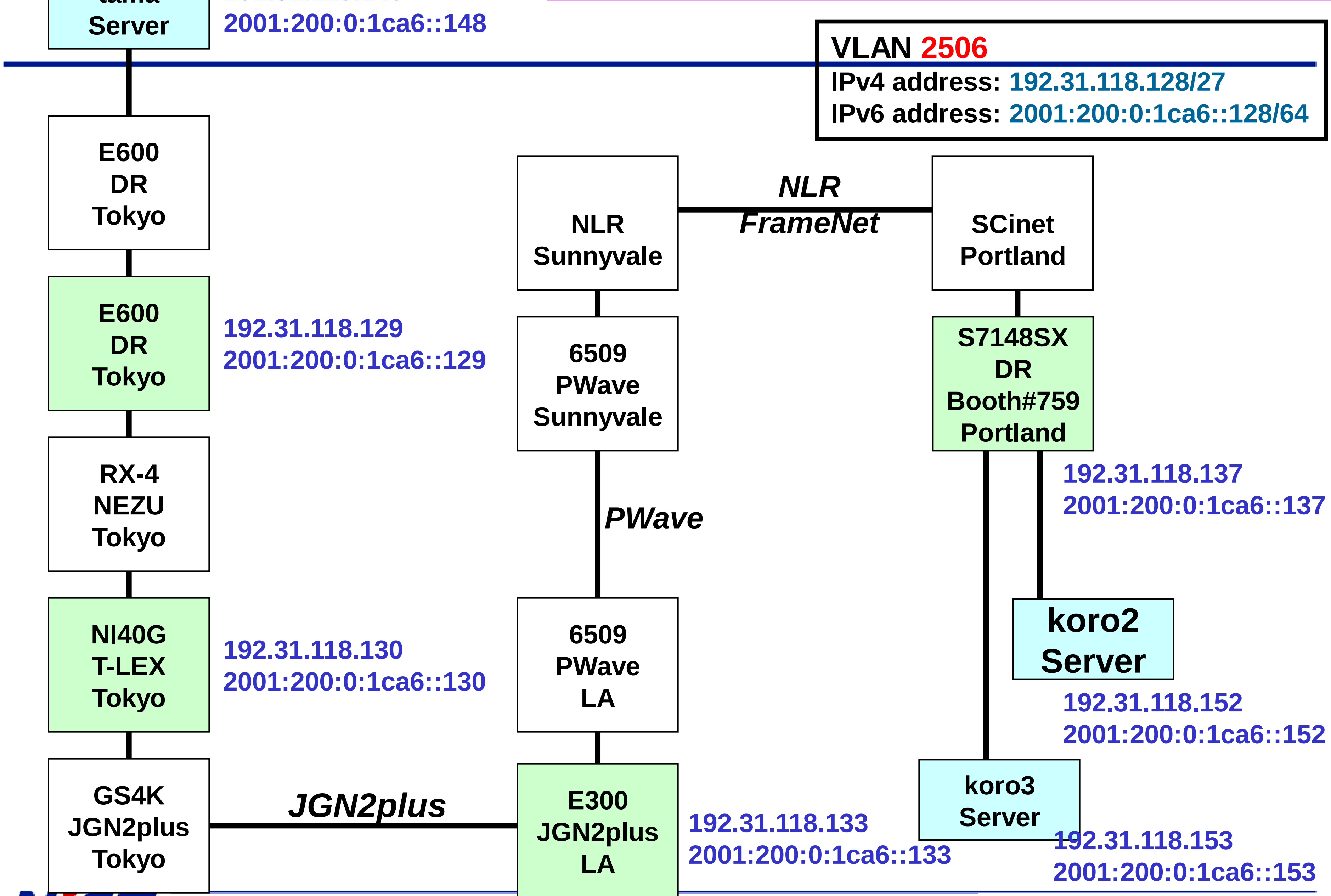


Software

CentOS 5.3
linux-2.6.18-128.el5
EXT3 file system
Apatch
FireFox


SC09@Portland
Bandwidth Challenge
The Univ. of Tokyo
Network
帯域幅挑戦ネットワーク
Portland, OR 2009
WIDE
JGN2plus
PACIFIC WAVE
NLR
SCinet
Tokyo, Japan
東京、日本
Server (tama)
the Univ. of Tokyo
DR E600
DR E600
WIDE RX4
NTT Otemachi
T-LEX NI40G
JGN2plus GS4000
West Coast, US
西海岸、米国
IEEAF RX4
PacificWave Seattle
NLR-FN Seattle
SCinet
DR S7148SX
Server (koro2)
Server (koro3)
NLR-FN Sunnyvale
PacificWave Sunnyvale
PacificWave Los Angeles
JGN2plus E300
IEEAF
10G
JGN2plus
10G
TYO
SEA
LAX
Chukchi Sea
Beaufort Sea
Bering Sea
Gulf of Alaska
North Pacific Ocean
North Atlantic Ocean
Gulf of Mexico
Mongolia
China
India
Indonesia
Papua New Guinea
México
Venezuela
Colombia
Brasil
Perú
©2009 Google - Imagery ©2009 TerraMetrics, NASA, Map data ©2009 ZENRIN, Google, Tele Atlas, INEGI, MapLink/Tele Atlas, Transnavicom, Europa Technologies - Terms of Use

SC09 Bandwidth Challenge Network Diagram



THE UNIVERSITY OF TOKYO

Enabling Technology

(1) Precise pacing --- Static and dynamic [Kamesawa+SC04]

(2) Optimization of TCP (buffers, windows, etc) [Yoshino+SC08]
(3) Modification on Apache and Firefox
- Apache Implementation
  - Omit unnecessary options for build
- Firefox Implementation
  - Optimize data operations
  - Reduce mem-copy & Omit useless operation
  - Adjust buffer size in Firefox
  - I/O optimization
  - Use mmap with madvice
  - EXT3 with write back
  - Background Frequent File-Cache flush

Performance(1)

File data transfer using EXT3 file system

- On pseudo LFN in laboratory (delay: 200ms)
  - File transfer rate : 8 Gbps
- On Real LFN between Tokyo and Portland
  - file transfer rate : xx Gbps
- Theoretical Maximum is 9.1 Gbps
  - JGN2plus is WAN PHY
- Performance is sensitive to packet loss ratio.

Performance(2)

- On 10G-LAN in laboratory (no delay)
  - Non-modified Firefox : 1.7 Gbps
  - USADAFOX : 7 Gbps
- On pseudo LFN in laboratory (delay: 200ms)
  - Non-modified Firefox : 3 Mbps
  - USADAFOX : 6.5 Gbps
- On Real LFN between Tokyo and Portland
  - Non-modified Firefox : 6.5 Mbps
  - USADAFOX : 6.5 Gbps

Measurement contents

(1)File transfer from Japan to SC09

- IPv4 and IPv6

(2)Web based data acces Japan to SC09

- IPv4 and IPv6

(3) File transfer by 2 PCs at the same time
Portland to Japan

Our contribution

- Apache + Firefox
  - enables everyone's utilization of 10G network
- Establish basic technology for robust use of 10G network
- 5th Generation Data Reservoir is a model case of the usage in next several years.

For more information

Visit
The University of Tokyo booth
#759