Performance limiting issues over high latency & high-bandwidth networks

Kei Hiraki

University of Tokyo

Computing System for real Scientists

- Fast CPU, huge memory and disks, good graphics
  - Cluster technology, DSM technology, Graphics processors
  - Grid technology
- Very fast remote file accesses
  - Global file system, data parallel file systems, Replication facilities

- Transparency to local computation
  - No complex middleware, or no small modification to existing software

↓

- **Real Scientists are not computer scientists**

- **Computer scientists are not work forces for real scientists**

Data Reservoir

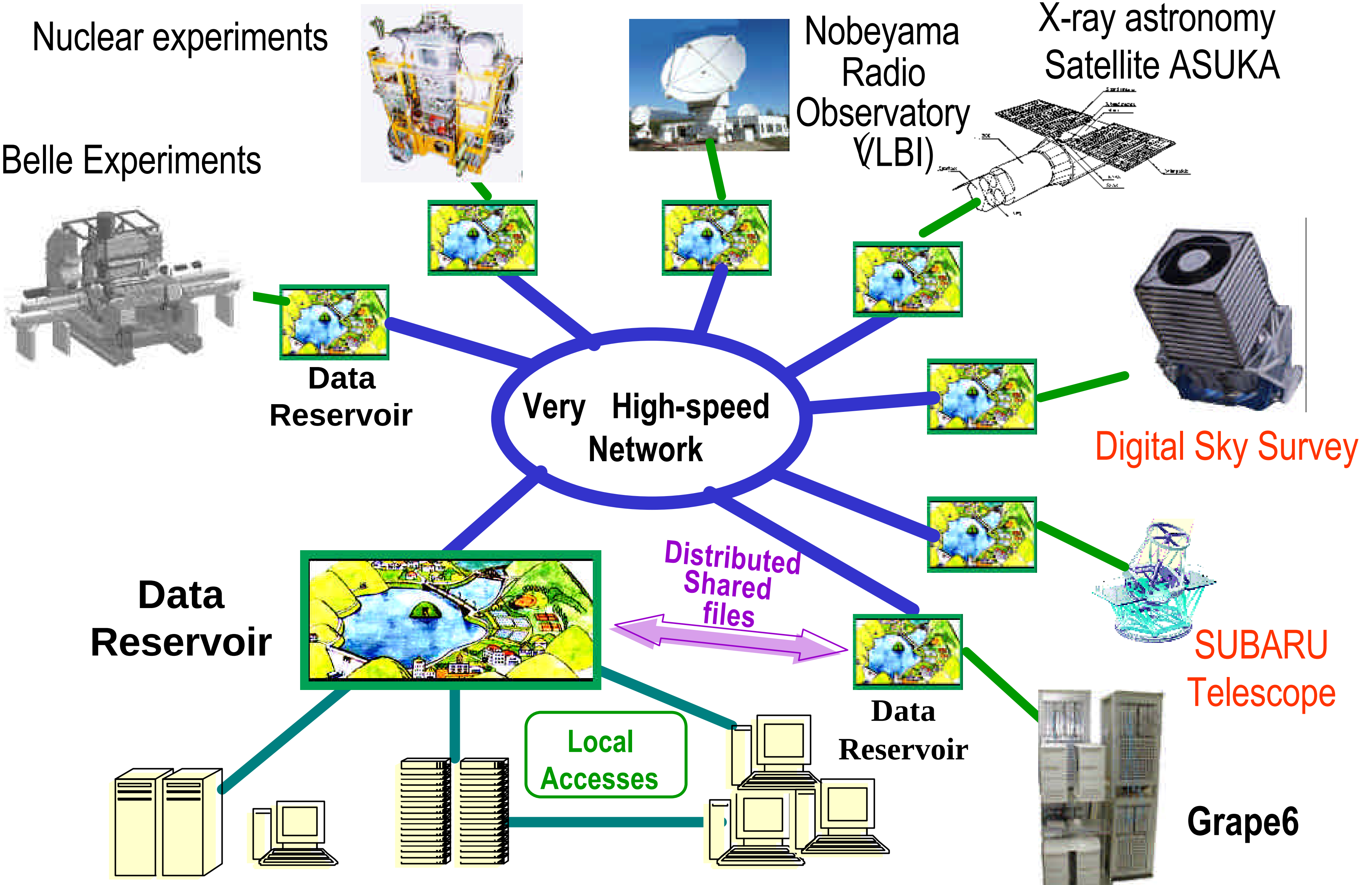
- Sharing Scientific Data between distant research institutes
  - Physics, astronomy, earth science, simulation data
- Very High-speed single file transfer on Long Fat pipe Network
  - > 10 Gbps,   > 20,000 Km,   > 400ms RTT
- High utilization of available bandwidth
  - Transferred file data rate > 90% of available bandwidth
    - Including header overheads, initial negotiation overheads
- OS and File system transparency
  - Storage level data sharing (high speed iSCSI protocol on stock TCP)
  - Fast single file transfer

GRAPE-DR

- GRAPE-DR:Very high-speed attached processor to a server
  - 2004 – 2008
  - Successor of Grape-6 astronomical simulator
- 2PFLOPS on 512 node cluster system
  - 1G FLOPS / processor
  - 512 processor / chip
  - 4 chips / PCI card
  - 2 PCI card / serer

  - 2 M processor / system
- Semi-general-purpose processing
  - N-body simulation, gridless fluid dyinamics
  - Linear solver, molecular dynamics
  - High-order database searching (genome, protein data etc.)

Data intensive scientific computation through global networks



Nuclear experiments

Belle Experiments

Nobeyama Radio Observatory (VLBI)

X-ray astronomy Satellite ASUKA

Data Reservoir

Very High-speed Network

Digital Sky Survey

SUBARU Telescope

Distributed Shared files

Data Reservoir

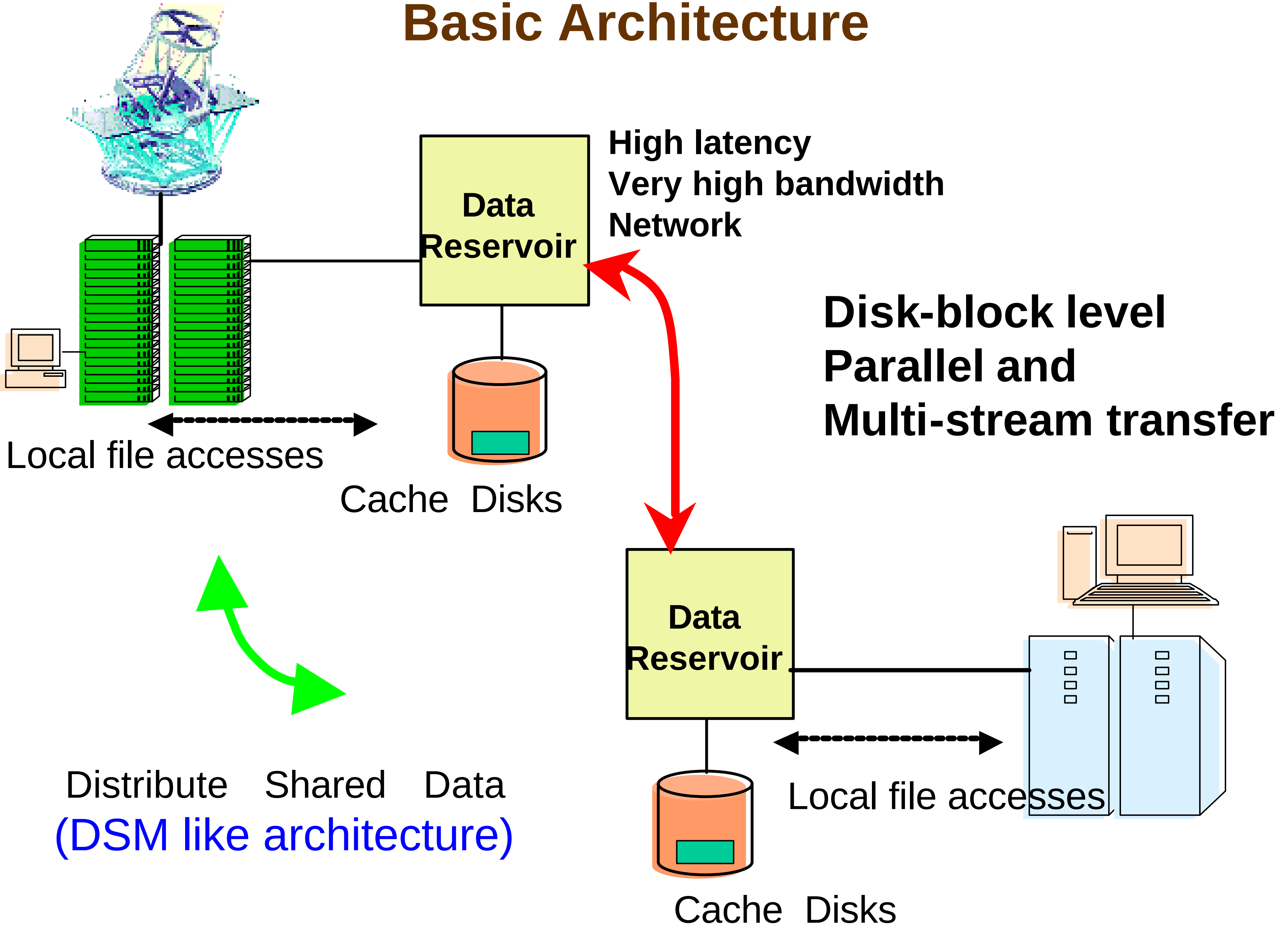
Data Reservoir

Local Accesses

Grape6

Data analysis at University of Tokyo

Basic Architecture



Data Reservoir

High latency
Very high bandwidth
Network

Disk-block level
Parallel and
Multi-stream transfer

Local file accesses

Cache Disks

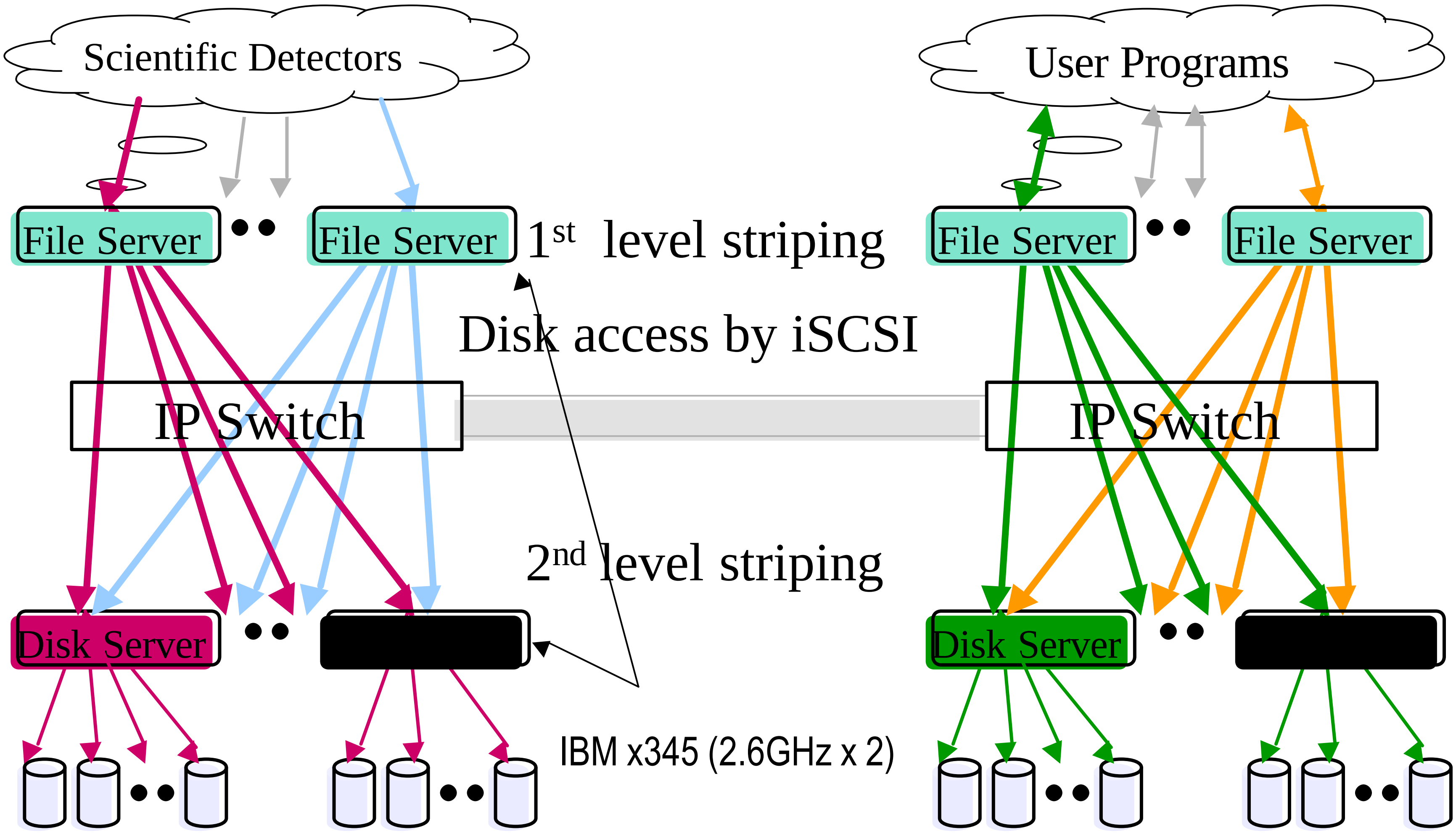
Data Reservoir

Local file accesses

Cache Disks

Distribute Shared Data
(DSM like architecture)

File accesses on Data Reservoir



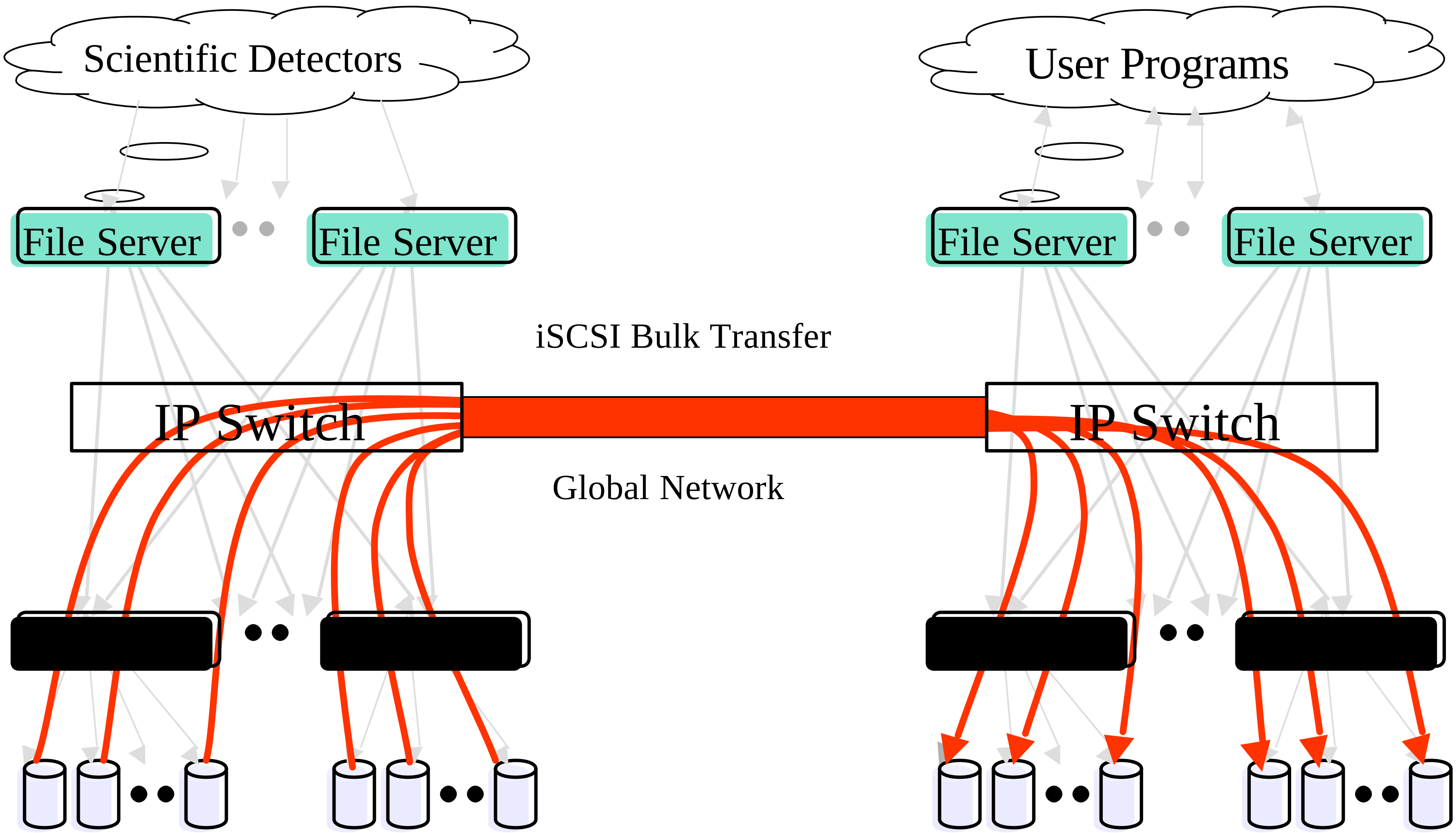
1st level striping

Disk access by iSCSI

2nd level striping

IBM x345 (2.6GHz x 2)

Global Data Transfer



1st Generation Data Reservoir

- **1st generation Data Reservoir (2001-2002)**
  - Efficient use of network bandwidth (SC2002 technical paper)
  - Single filesystem image (by striping of iSCSI disks)
  - Separation of local file accesses by users to remote disk to disk replication
  - Low level data-sharing (disk block level, iSCSI protocol)

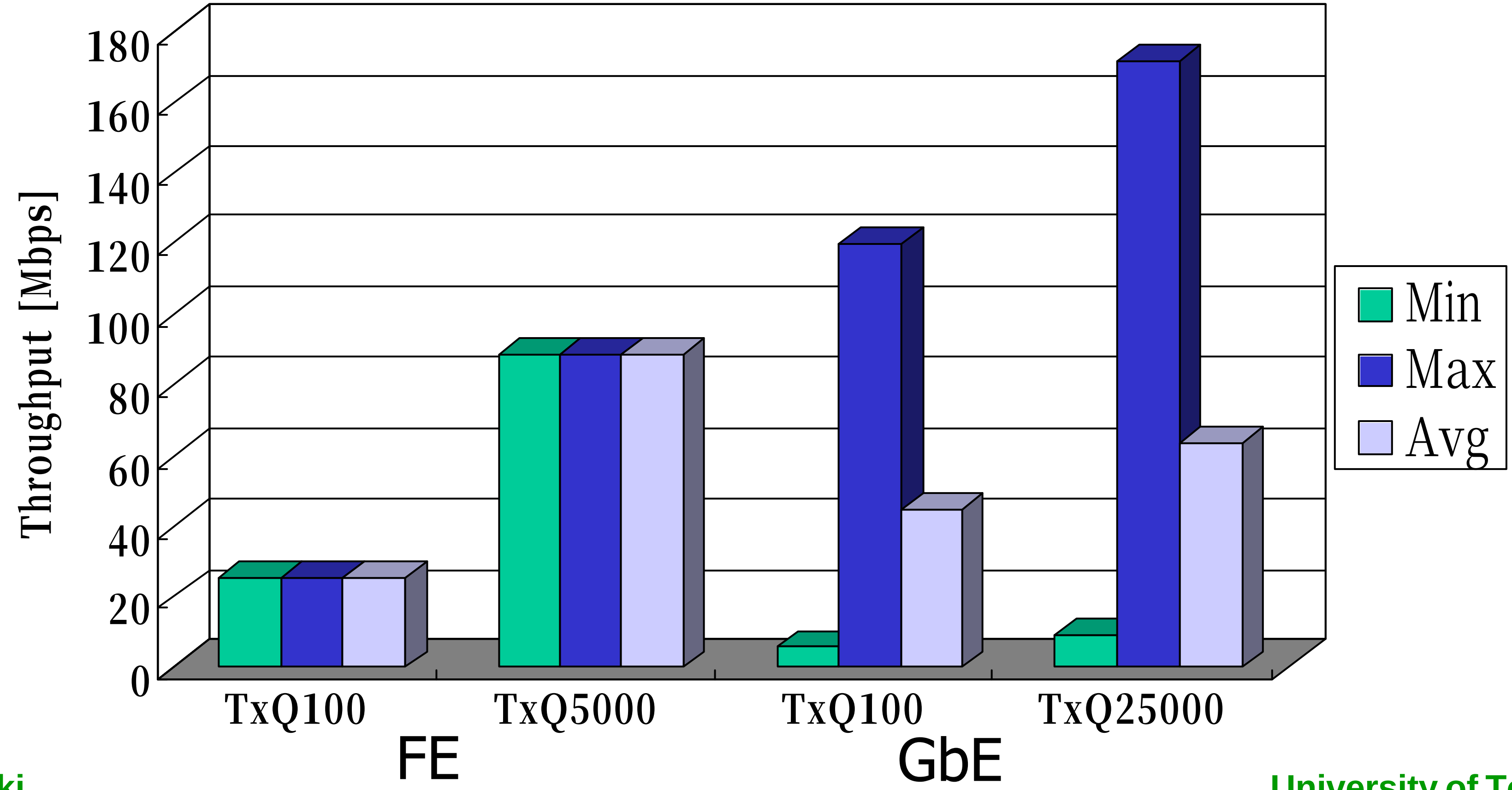
- **26 servers for 570 Mbps data transfer**
  - High sustained efficiency (93%）
  - Low TCP performance
    - 21Mbps/server --- very slow
  - Unstable TCP performance

Problems

- Low TCP bandwidth due to packet losses
  - TCP congestion window size control
  - Very slow recovery from fast recovery phase (>20min)
- Unbalance among parallel iSCSI streams
  - Packet scheduling by switches and routers
  - User and other network users have interests only to total behavior of parallel TCP streams
- Unstable network behavior from application soft

Our starting point (1)

- **Fast Ethernet vs. GbE**
  - ? Iperf in 30 seconds
  - ? Min/Avg: Fast Ethernet > GbE



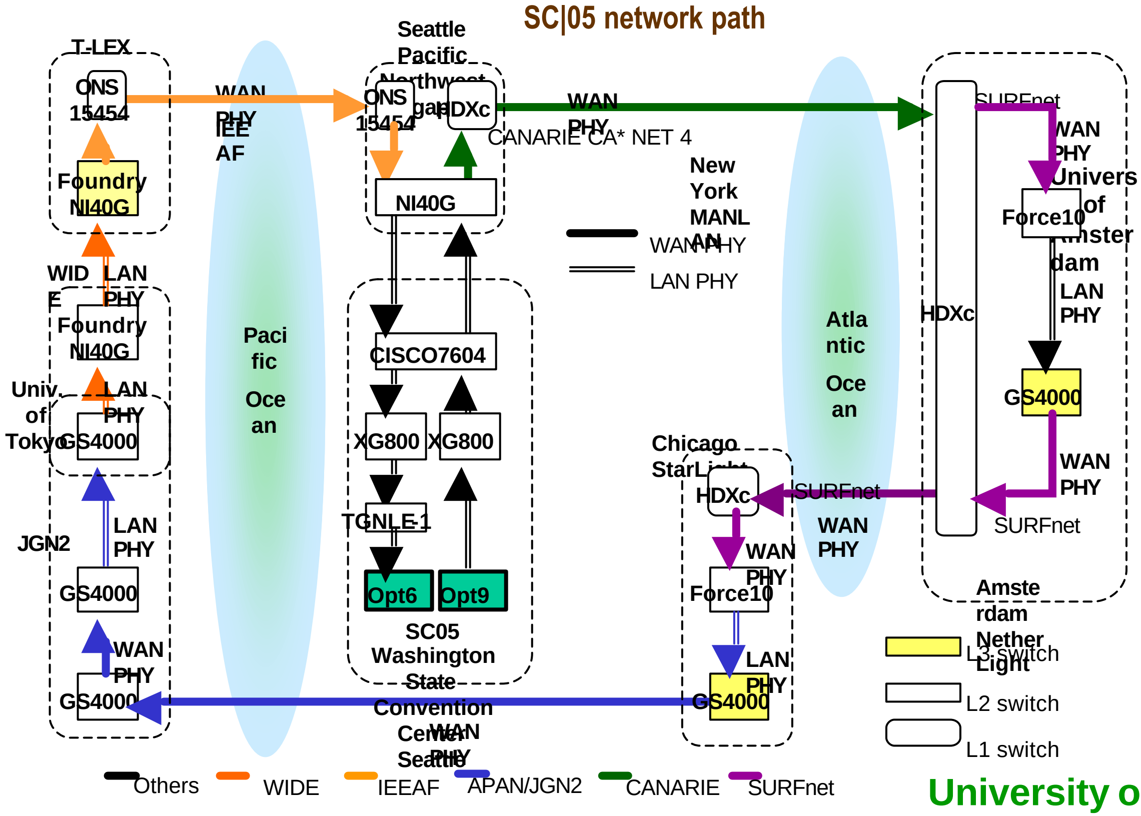
Our starting point (2)

- **Delay emulator v.s. actual network**
  - Performance on a delay emulator is much better than actual network

Delay emulator

Clock level accuracy

Real network
500ms RTT

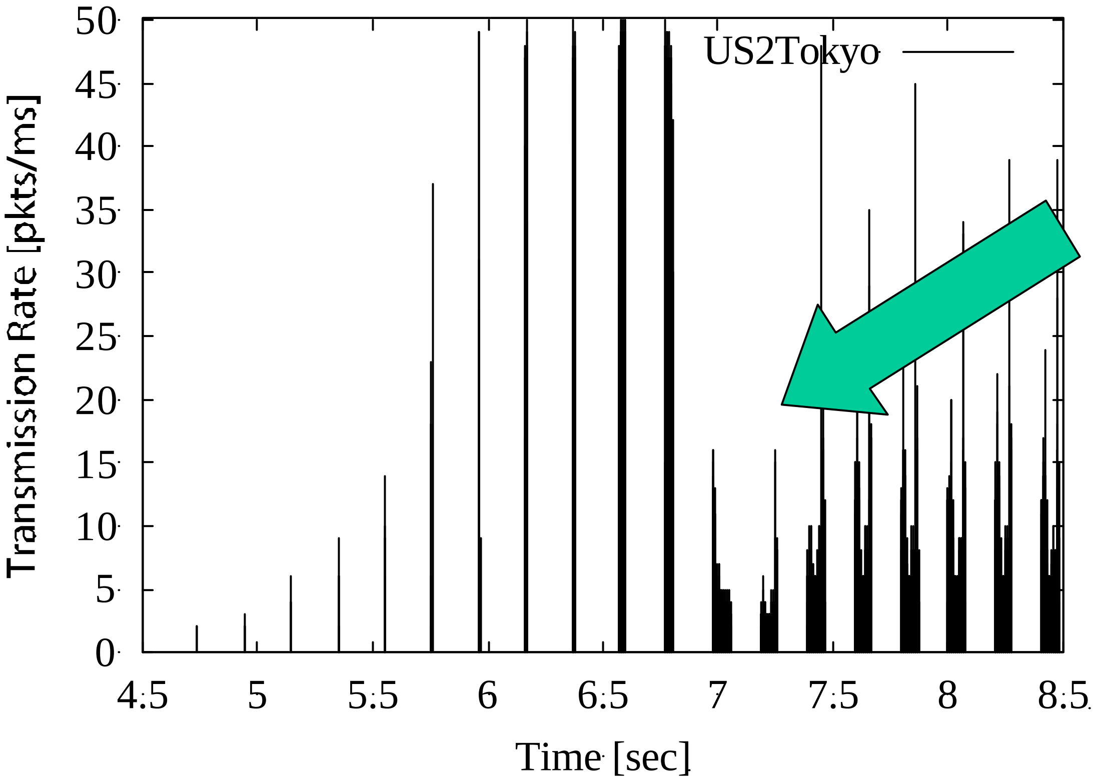


Packet Transmission Rate

? Bursty behavior

  ? Transmission in 20ms against RTT 200ms

  ? Idle in rest 180ms



Packet loss occurred

Unbalance within parallel TCP streams

? Unbalance among parallel iSCSI streams

  ? Packet scheduling by switches and routers

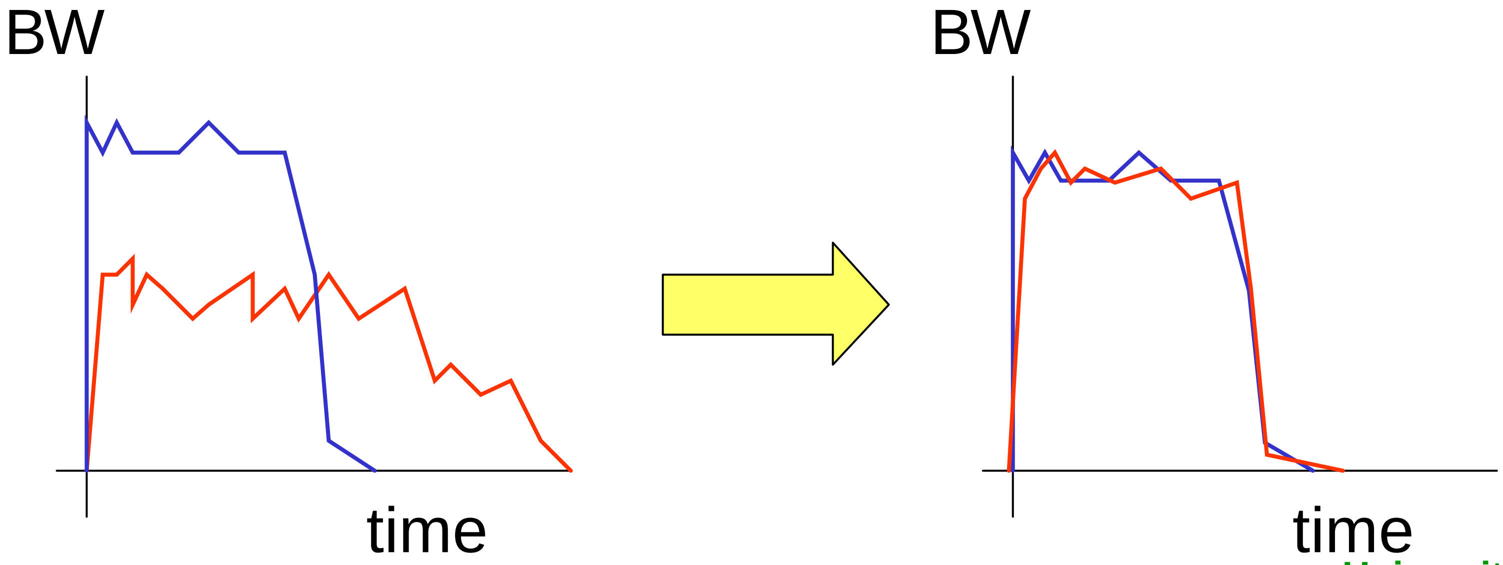
  ? Meaningless unfairness among parallel streams

  ? User and other network users have interests only to total behavior of parallel TCP streams

? Our approach

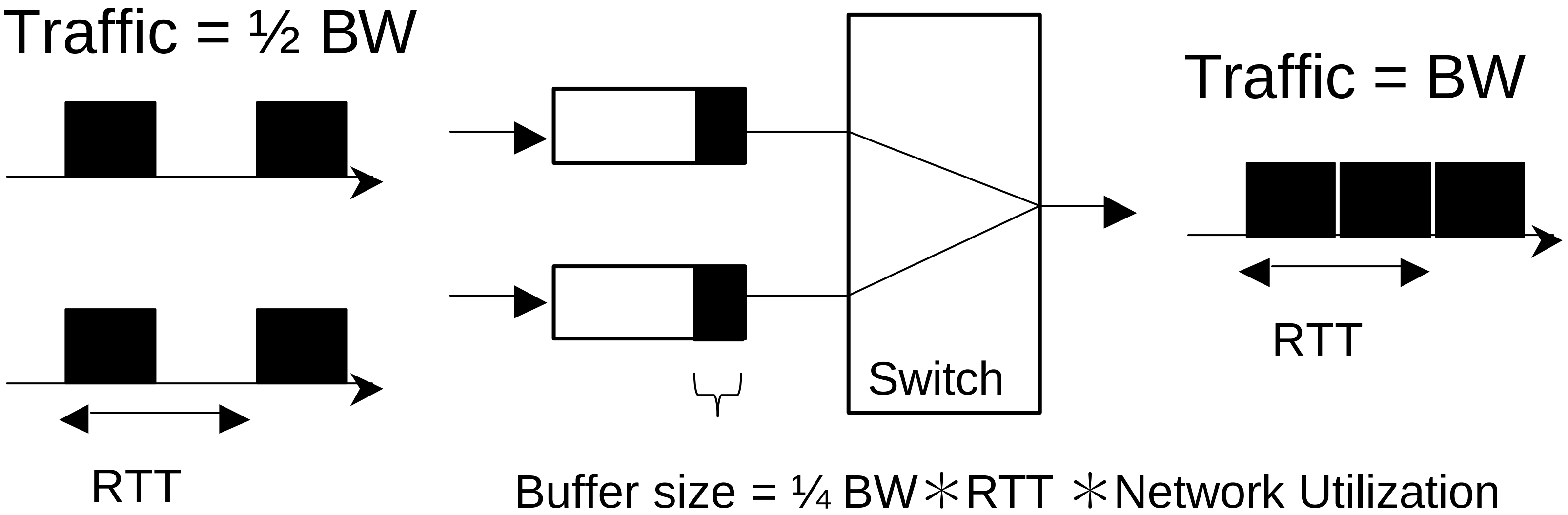
  ? Constant $\Sigma \boldsymbol{cwnd}_i$ for fair TCP network usage to other users

  ? Balance each $cwnd_i$ communicating between parallel TCP streams



Merging TCP streams

- Worst case is 2 TCP streams -> 1 TCP stream
- Bursty traffic just after slow start is the worst case



Buffer size = ¼ BW＊RTT ＊Network Utilization

10 Gbps, 200msRTT ⇒ 62.5 MB * Utilization factor

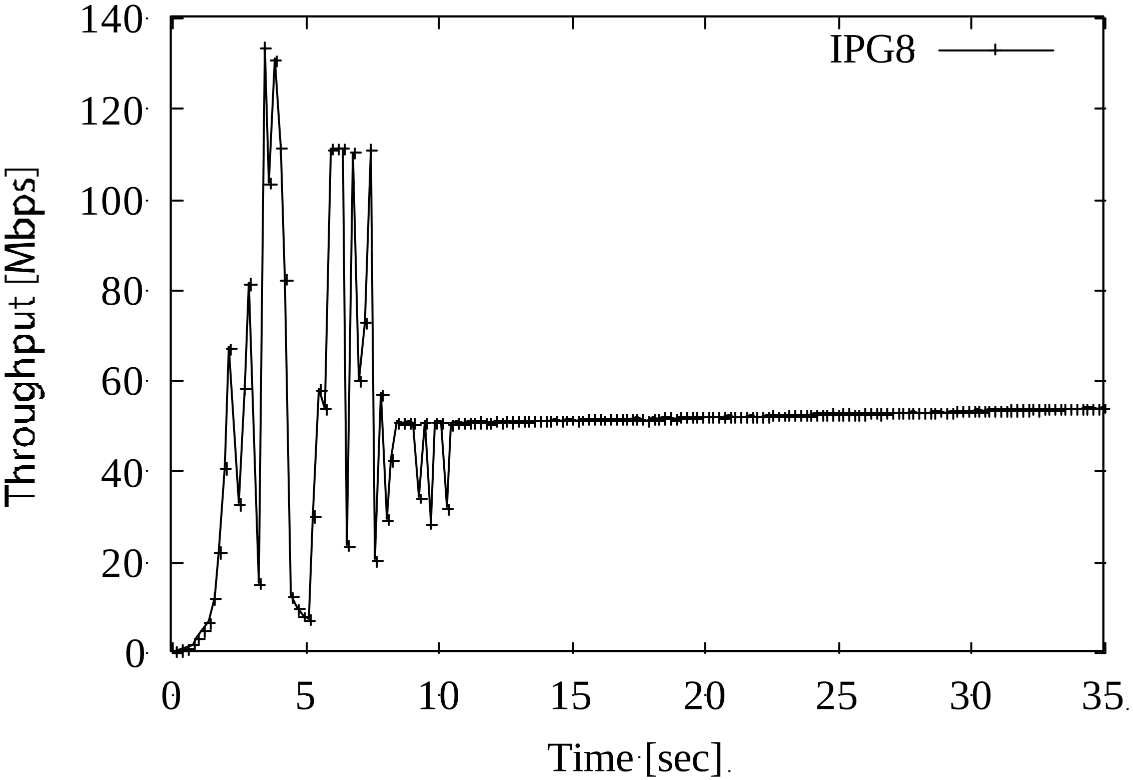
Many switches do not have large buffer
Flow control cannot apply at intermediate switches

Example Case of 8 IPG

? Success on Fast Retransmit

  ? Smooth Transition to Congestion Avoidance

  ? CA takes 28 minutes to recover to 550Mbps

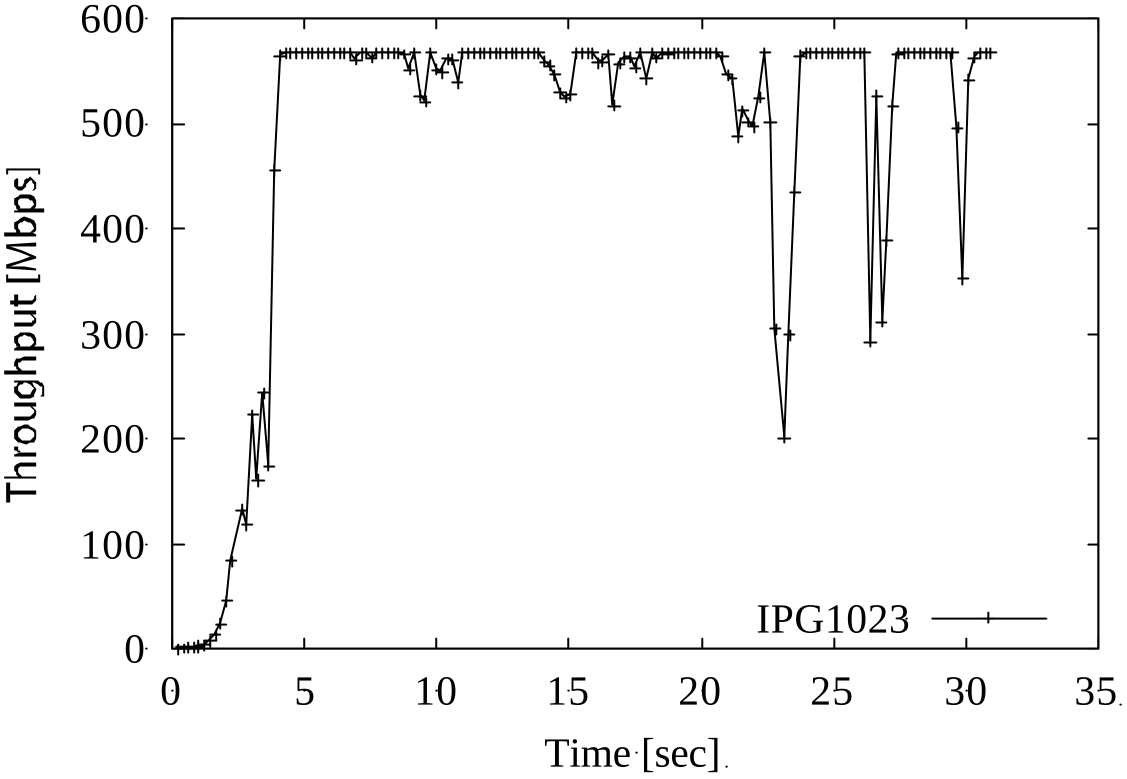


Best Case of 1023B IPG

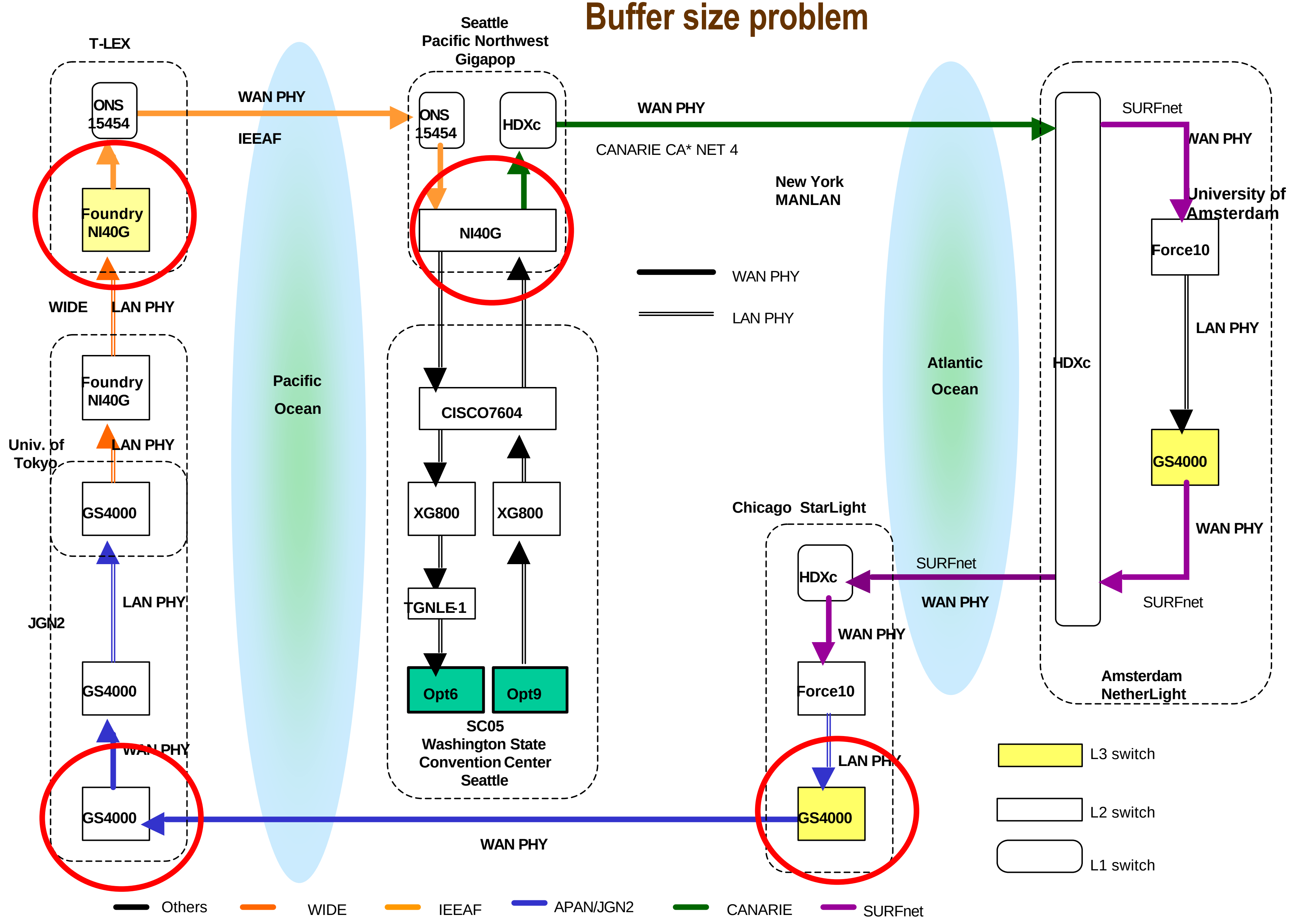
? Like Fast Ethernet case

  ? Proper transmission rate

? Spurious Retransmit due to Reordering

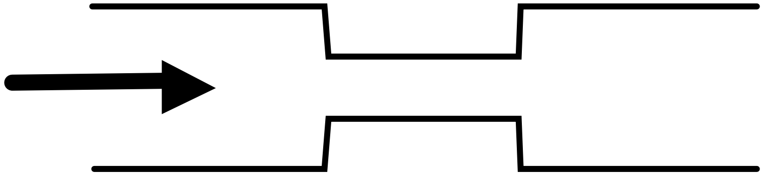


Buffer size problem

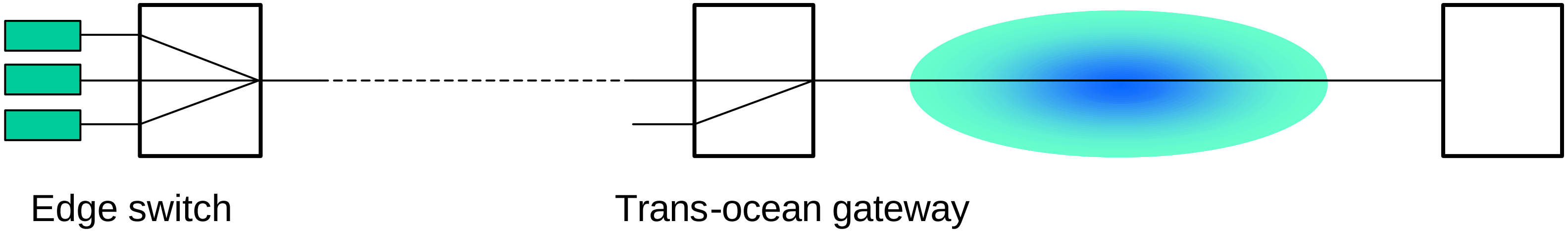


Sender side pacing

- BW bottleneck in the middle of network



- Too small buffer size of edge switch and trans-ocean gateway switch



- Coordination between MAC and TCP layer (variable packet pace)
  - Avoid unnecessary packet loss at slow start phase

Difficulty (1)

- Artificial packet losses
  - Average
- Bursty behavior of TCP traffic
  - Improvement in TCP or TCP-like protocol is not effective
  - Pacing works in some situation
- Unbalance between parallel TCP streams
  - Application performance may decided by the worst stream
  - No good balancing protocol below network layer
- Too small buffer size at switches
  - Merging TCP streams cause artificial packet losses
  - Minimum buffer size > ¼ RTT * BW is necessary

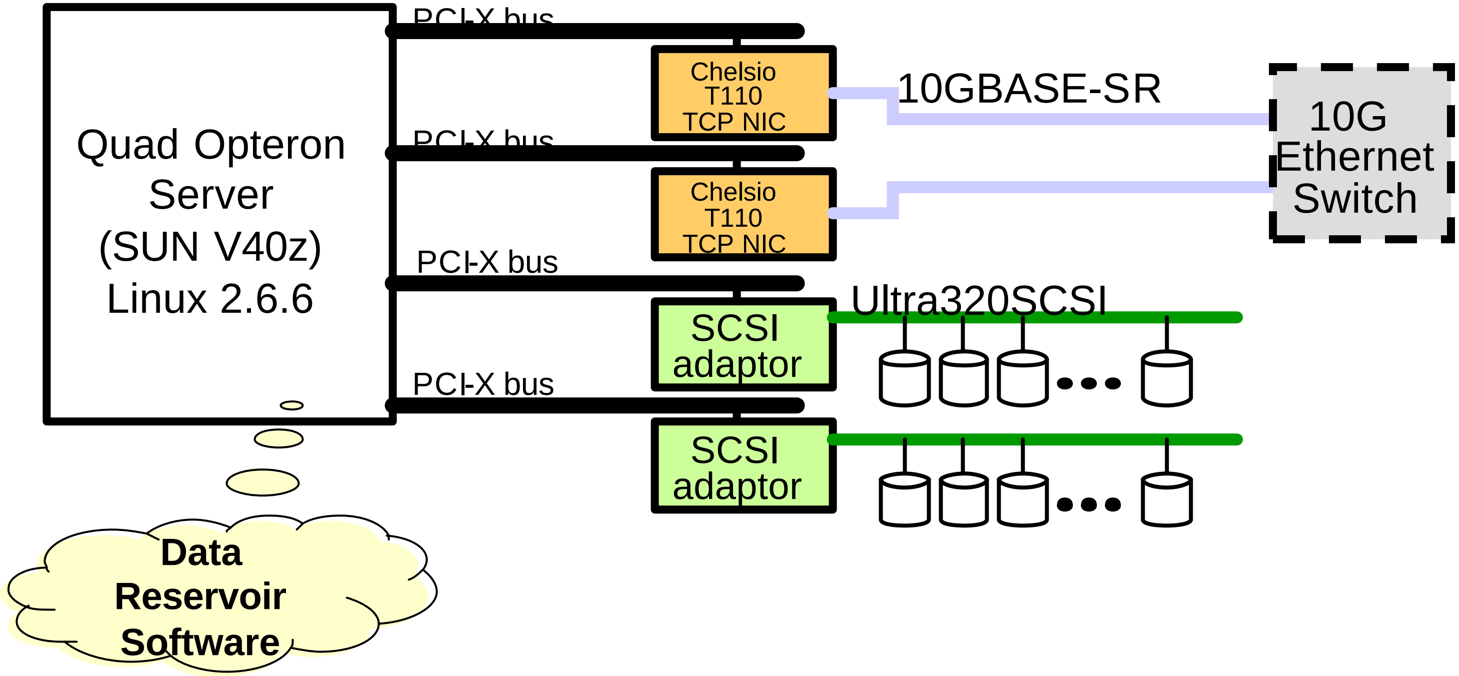
3rd Generation Data Reservoir

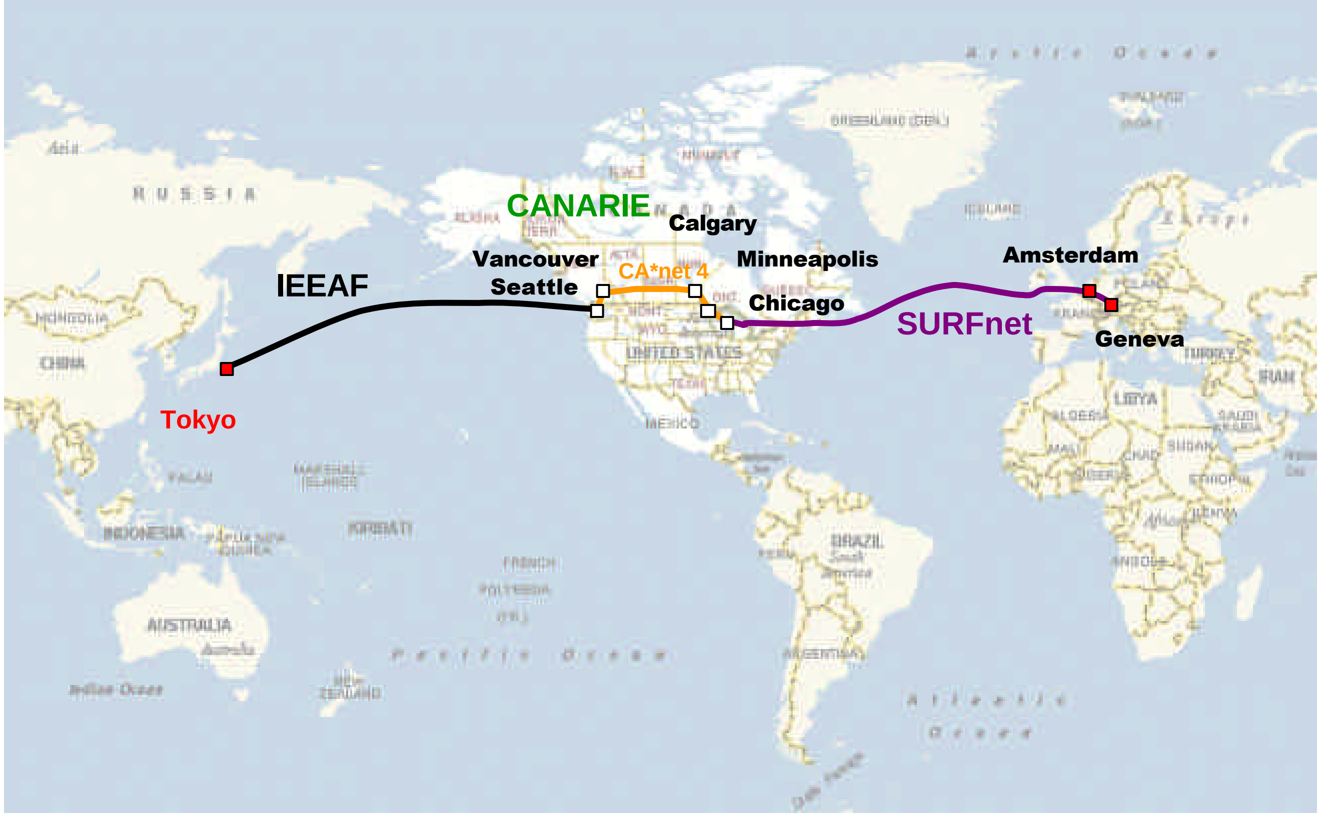
- **Hardware and software basis for 100Gbps Distributed Data-sharing systems**

- 10Gbps disk data transfer by a single Data Reservoir server
- Transparent support for multiple filesystems (detection of modified disk blocks)
- Hardware(FPGA) implementation of Inter-layer coordination mechanisms
- 10 Gbps Long Fat pipe Network emulator and 10 Gbps data logger

Utilization of 10Gbps network

- **A single box 10 Gbps Data Reservoir server**
  - Quad Opteron server with multiple PCI-X buses (prototype, SUN V40z server)
  - Two Chelsio T110 TCP off-loading NIC
  - Disk arrays for necessary disk bandwidth
  - Data Reservoir software (iSCSI deamon, disk driver, data transfer maneger)





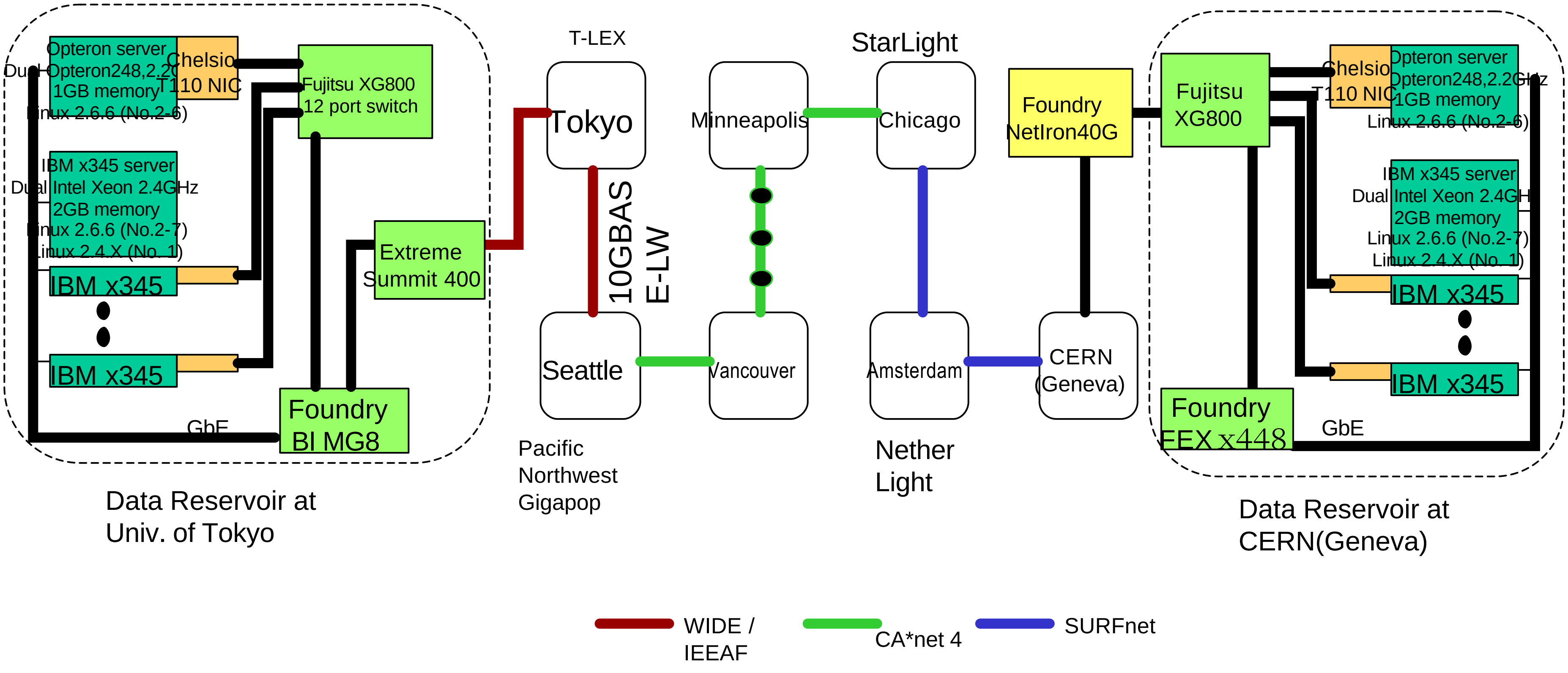
Network used in the experiment

🟥 End Systems

□ A L1 or L2 switch

Tokyo-CERN Network connection

Network topology of CERN-Tokyo experiment

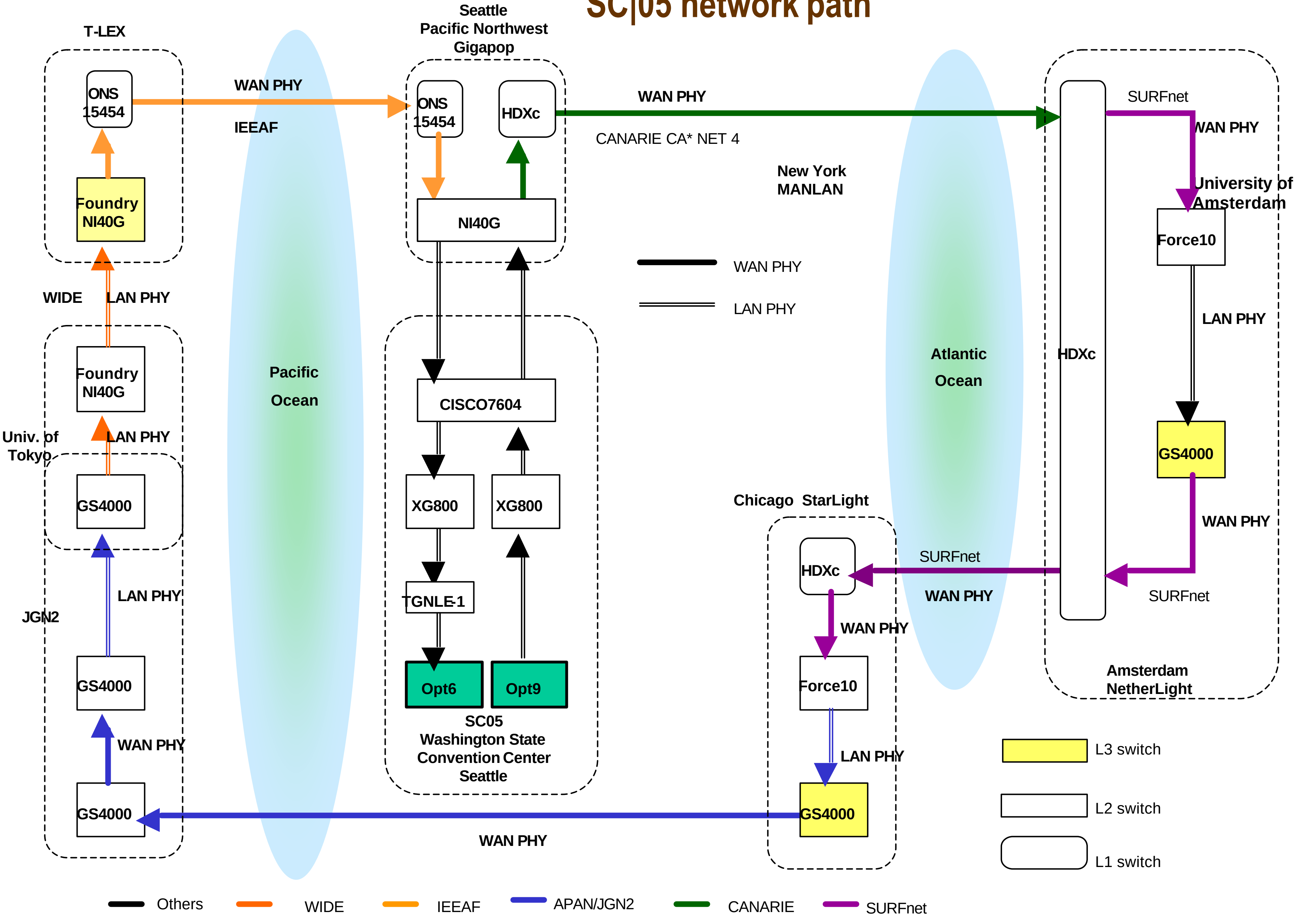


Difficulty (2)

- Long-distance L2 network
  - Difficulty in debugging
  - Unstableness due to Spanning Tree algorithm
  - Long latency MAC address detection

- Our opinion
  - Avoid long distance L2 network
  - Especially on trans-ocean connection

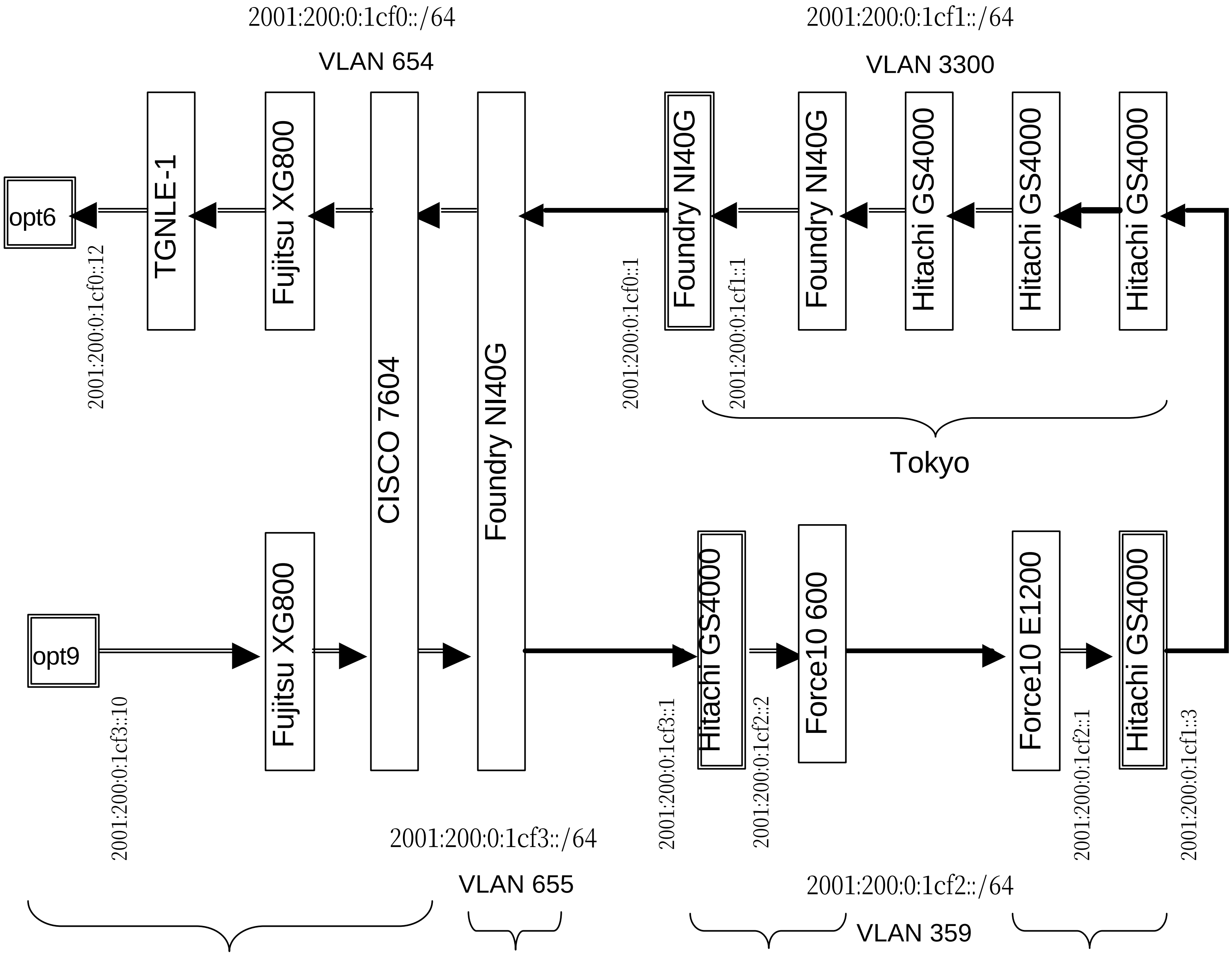
SC|05 network path



Kei Hiraki

University of Tokyo

SC|05 address arrangement



2001:200:0:1cf0::/64

VLAN 654

2001:200:0:1cf1::/64

VLAN 3300

opt6

2001:200:0:1cf0::12

TGNLE-1

Fujitsu XG800

CISCO 7604

Foundry NI40G

2001:200:0:1cf0::1

Foundry NI40G

2001:200:0:1cf1::1

Foundry NI40G

Hitachi GS4000

Hitachi GS4000

Hitachi GS4000

Tokyo

opt9

2001:200:0:1cf3::10

Fujitsu XG800

Hitachi GS4000

2001:200:0:1cf3::1

2001:200:0:1cf2::2

Force10 600

Force10 E1200

2001:200:0:1cf2::1

Hitachi GS4000

2001:200:0:1cf1::3

2001:200:0:1cf3::/64

VLAN 655

2001:200:0:1cf2::/64

VLAN 359

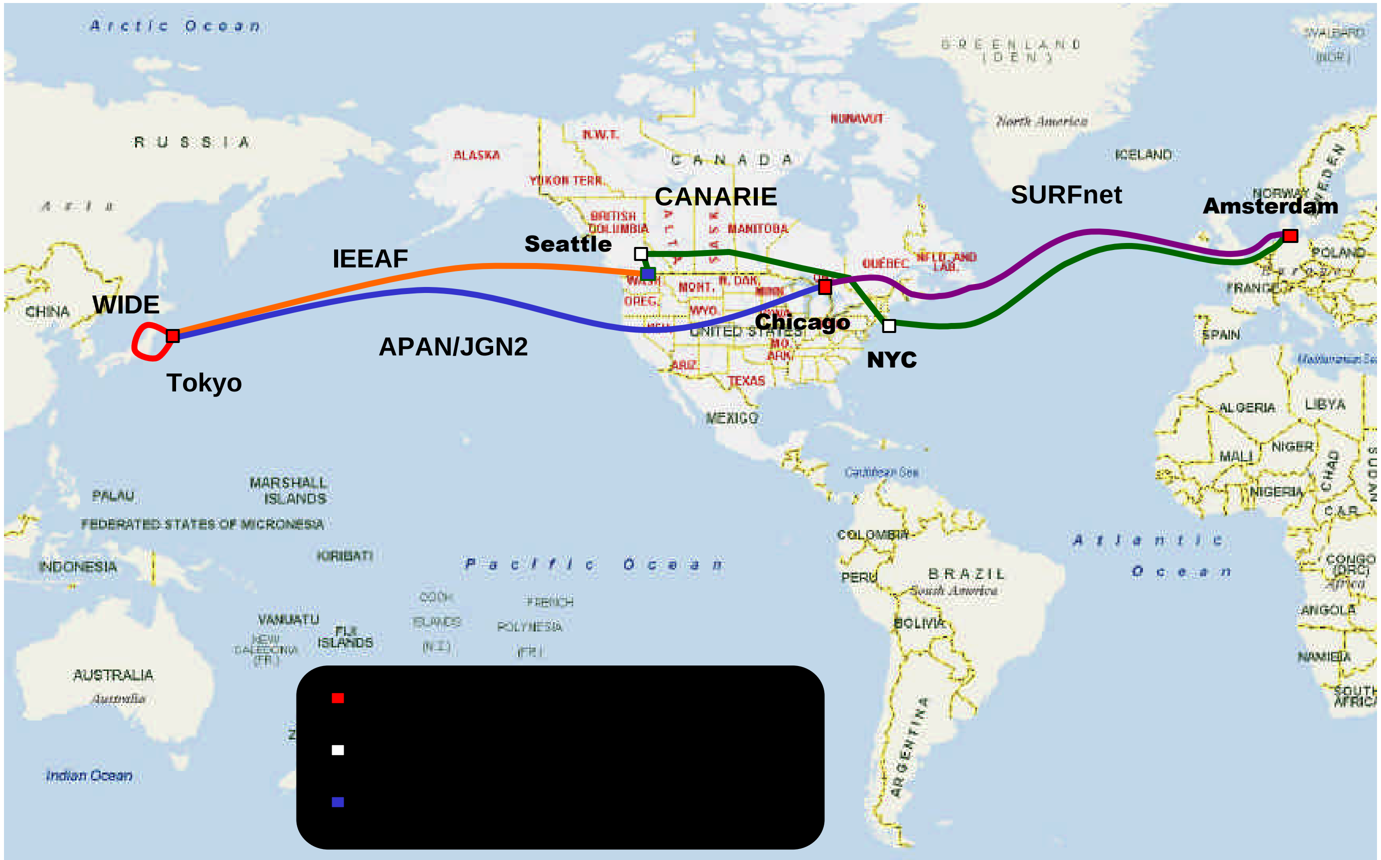
Seattle SC05 booth 233

Seattle Westin

Amsterdam

Chicago

SC|05 network path map



Difficulty (3)

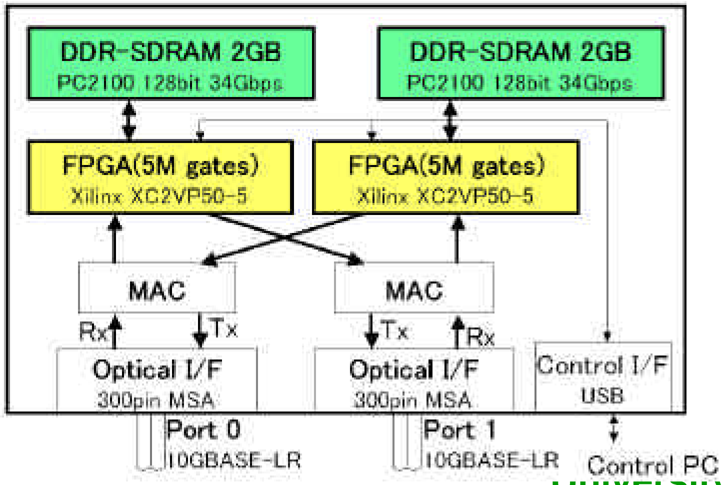
- Observation
  - 500ms precise delay shows much better performace than real network
    - Anue 10Gbps delay emulator ⇒ same level to local performance
    - Seattle -> Tokyo -> Chicago -> Amsterdam -> Seattle network

      ⇒ Unstable and less performance to local performance
  - Unnecessary packet loss at receiving server
    - Bursty packet stream at receiver
    - Bottleneck at I/O bus, CPU utilization, and amount of buffer
  - Performance difference by intermediate switch / router
    - Difference is about 10%
    - Sender side pacing is not effective

Receiver side pacing

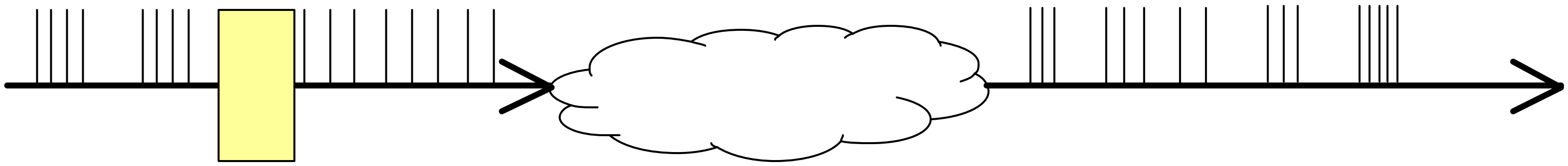
- Objective
  - Reduce packet losses due to receiver bottleneck
    - Act as large receiving buffer at receiving Network Interface Card
    - Set to the maximum receiving speed of the system

- Implementation
  - SANSEI-system TGNLE-1 FPGA based buffering system
  - FPGA is used to implement fine-grain pacing mechanism
  - 2GB buffer memory (DDR memory)

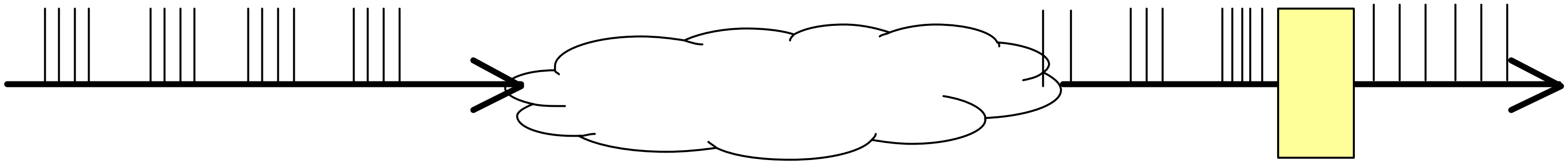


Pacing

- Sender side pacing
  – Effective for the bottleneck middle of the network
    - Complex hardware to cooperate with TCP protocol
    - Not effective to some 10G switches

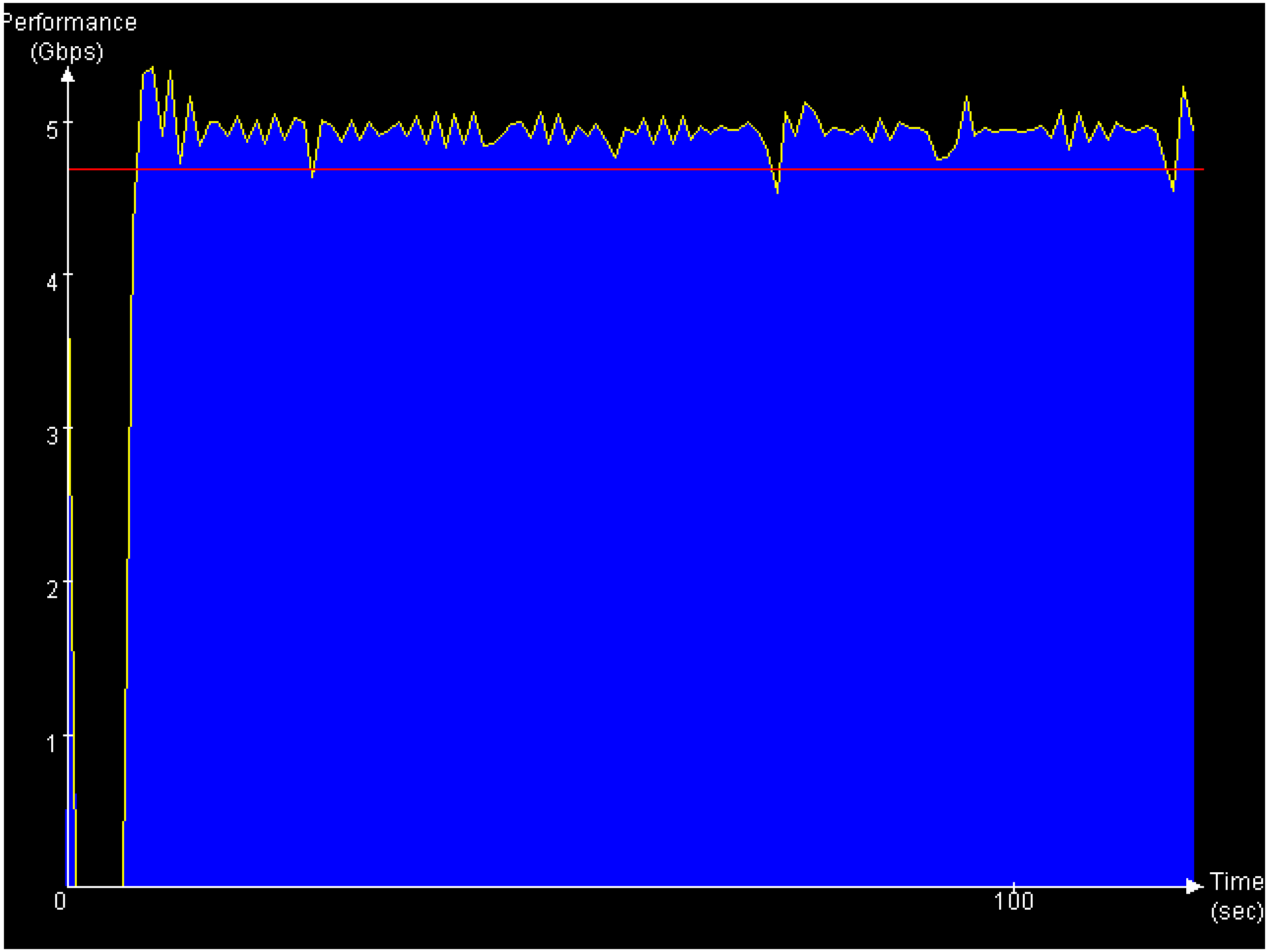


- Receiver side pacing
  - Effective to NIC whose maximum bandwidth is less than 10G
  - Simple hardware. It can be implemented in NIC
  - Increase latency when bandwidth change



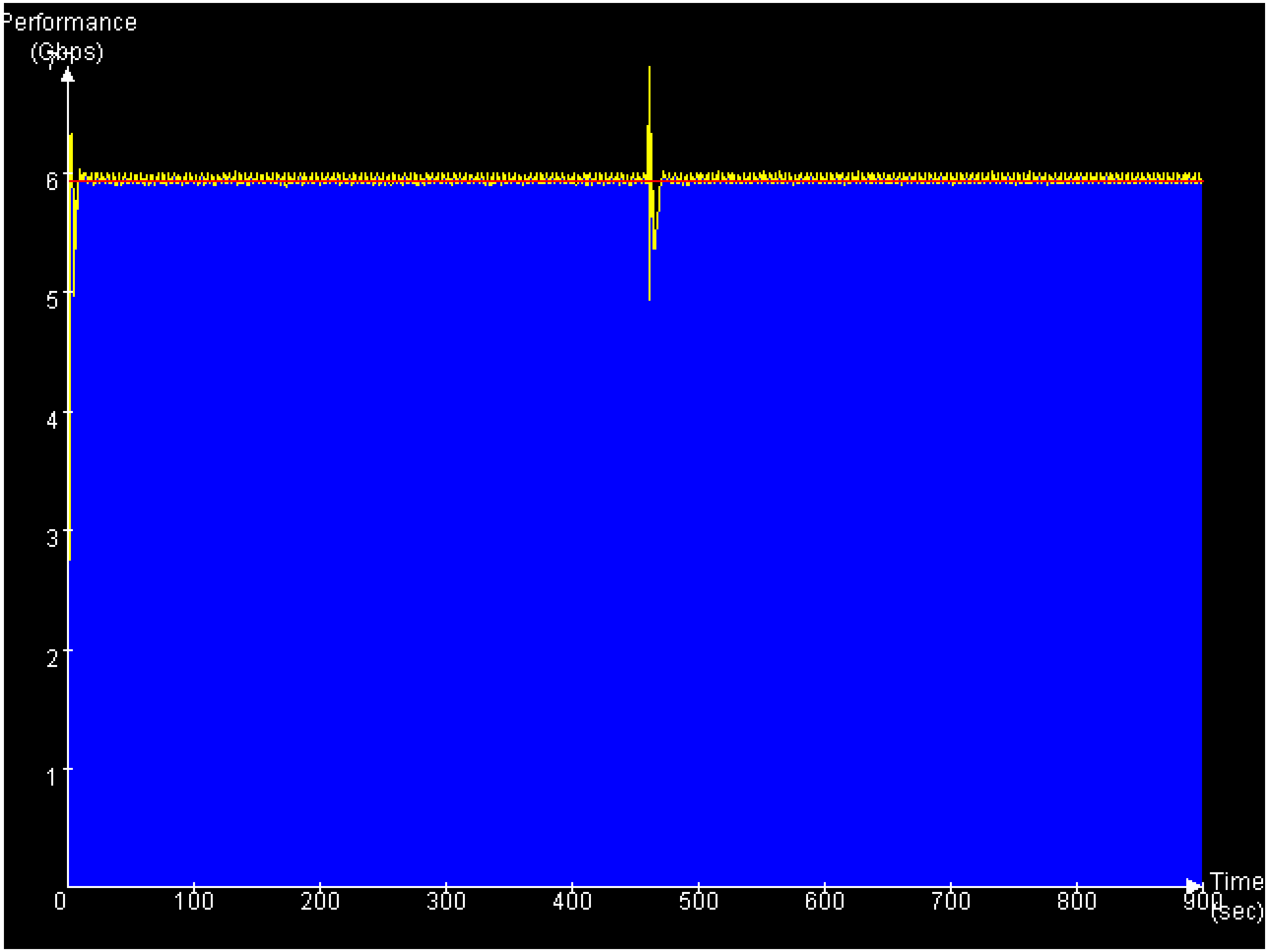
Preliminary Results

- Without receiver side pacing



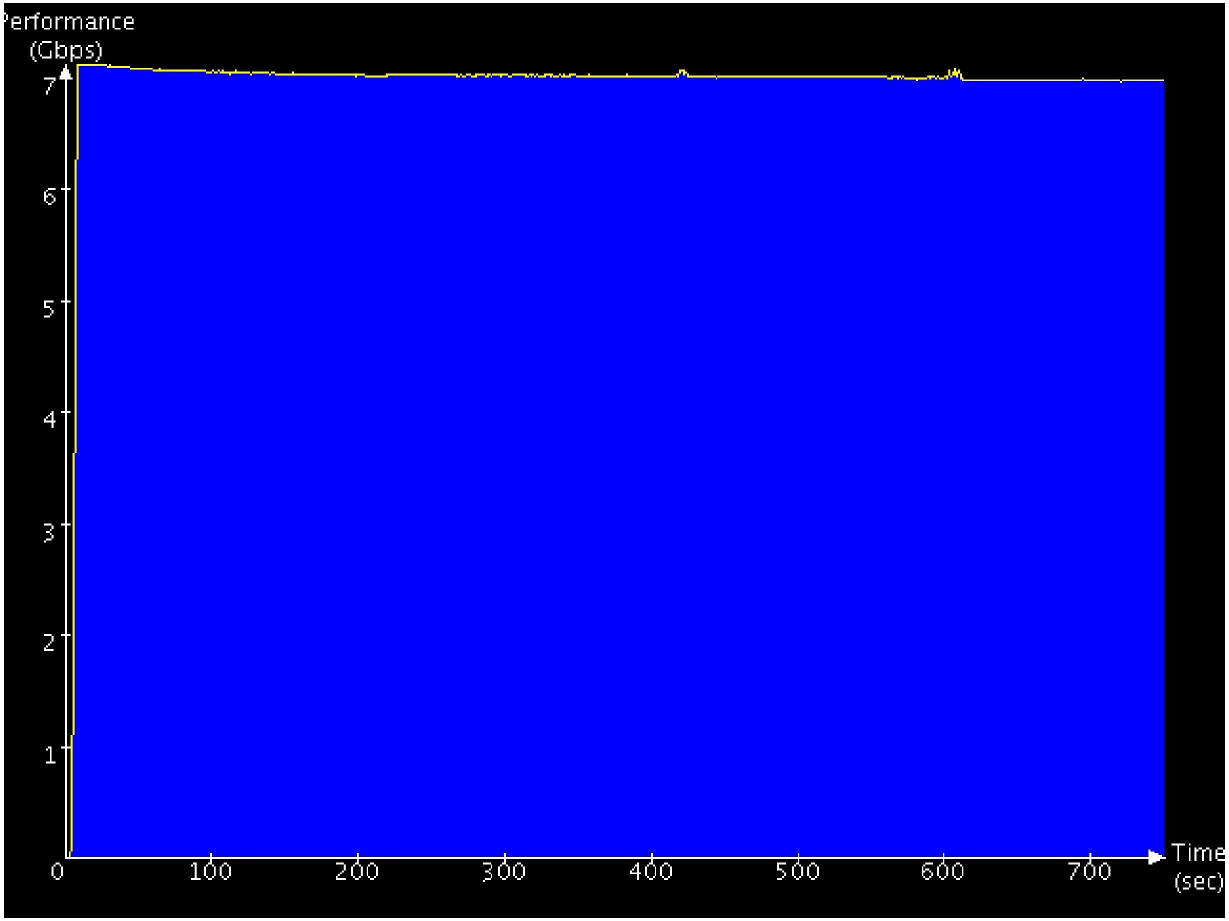
Preliminary Results

- Without receiver side pacing

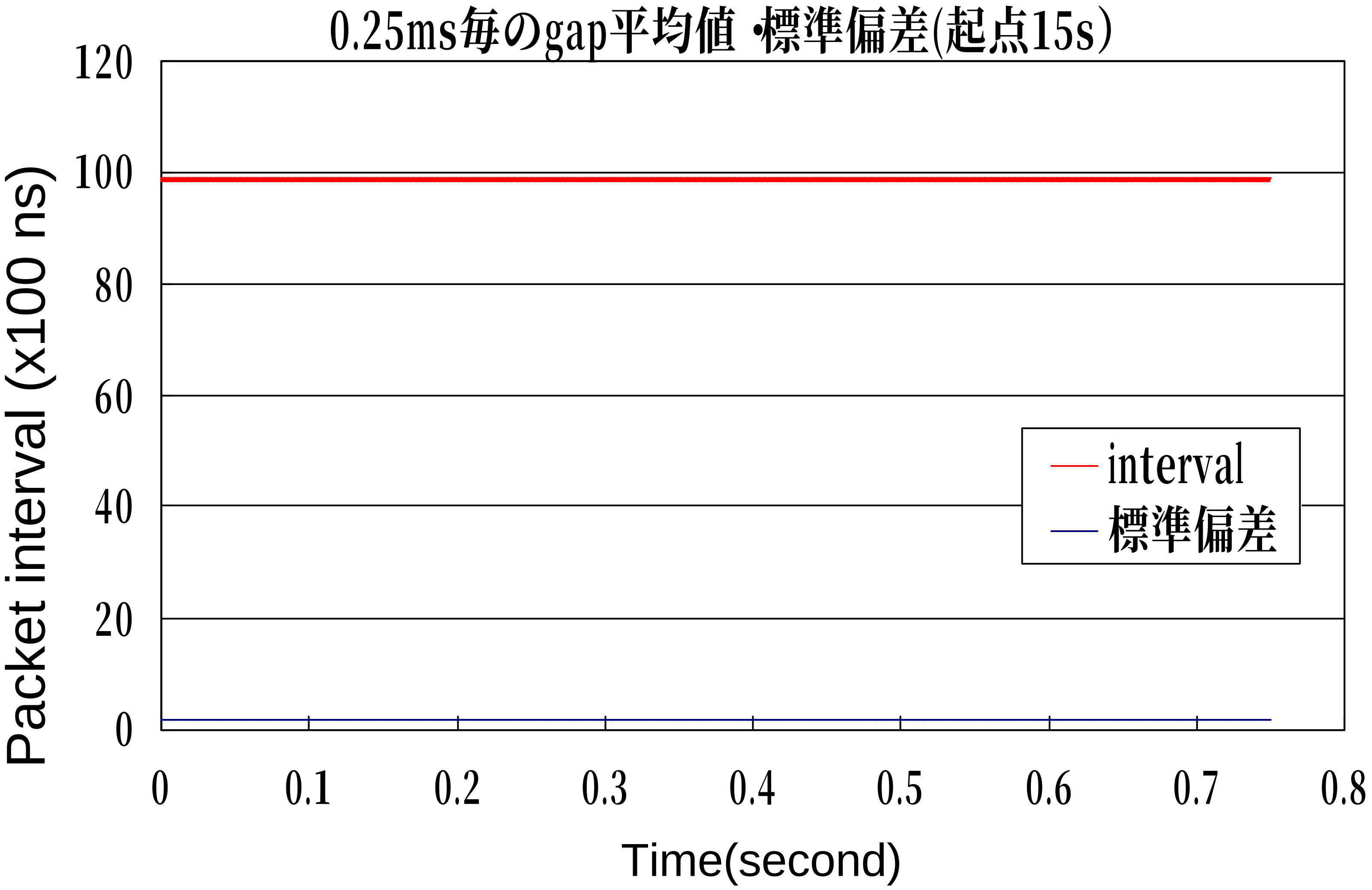


Preliminary Results

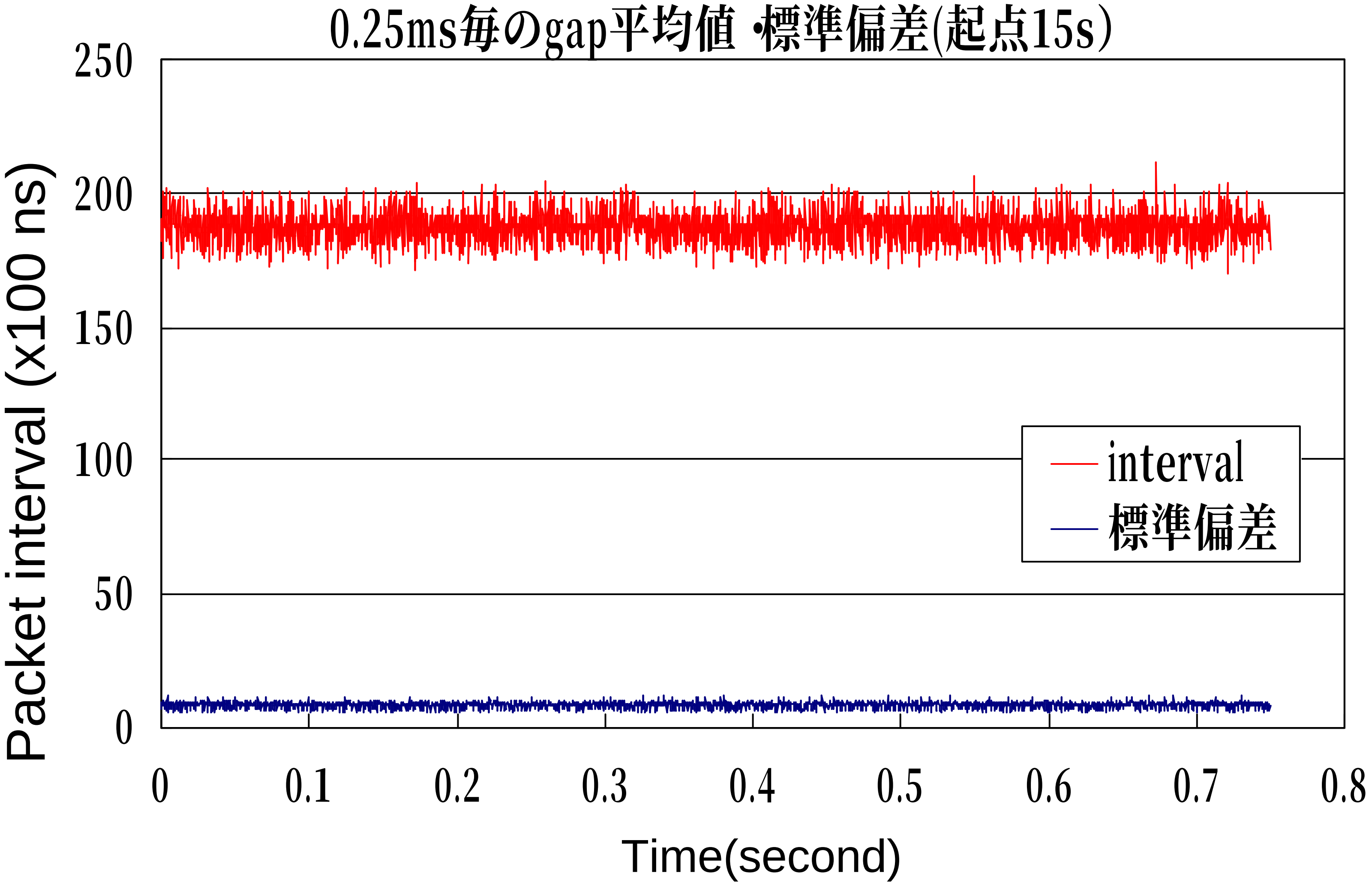
- IPv6 TCP single stream
  - 6.96Gbps (use of "receiver side pacing")
  - 5.58Gbps (without "receiver side pacing")



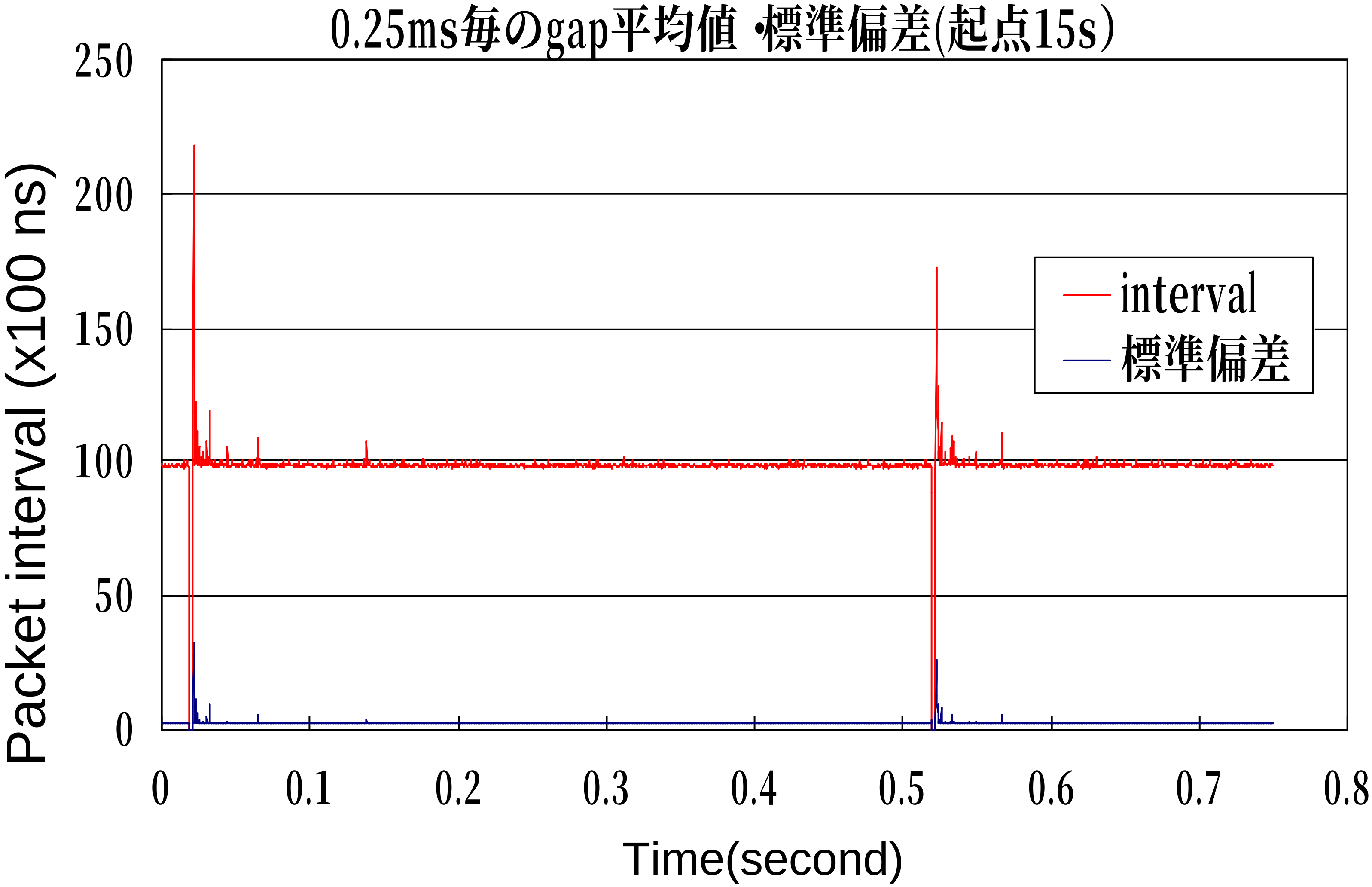
Local :Data packets@Sender-TX



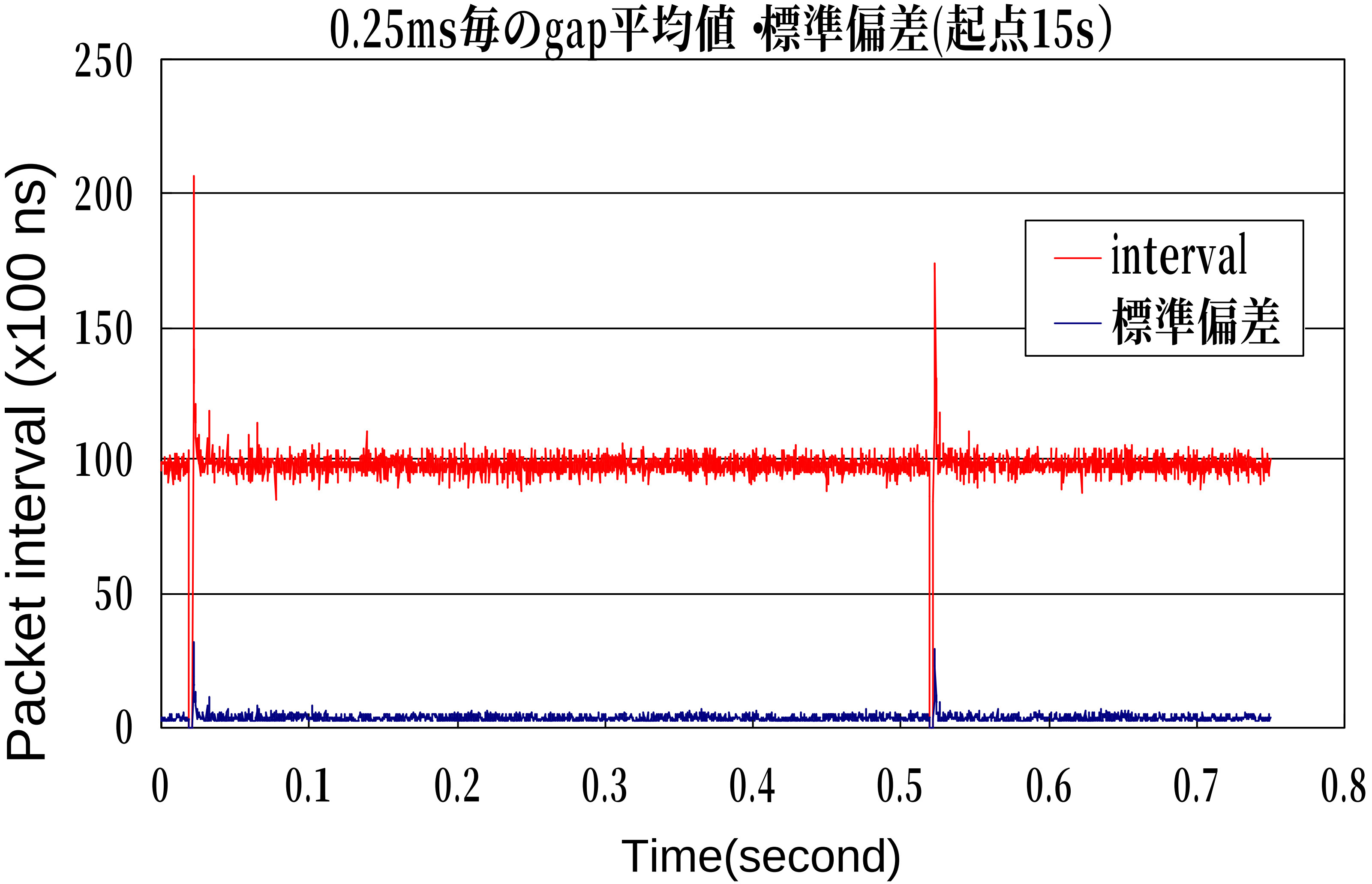
Local ：ACKpackets@Sender-RX



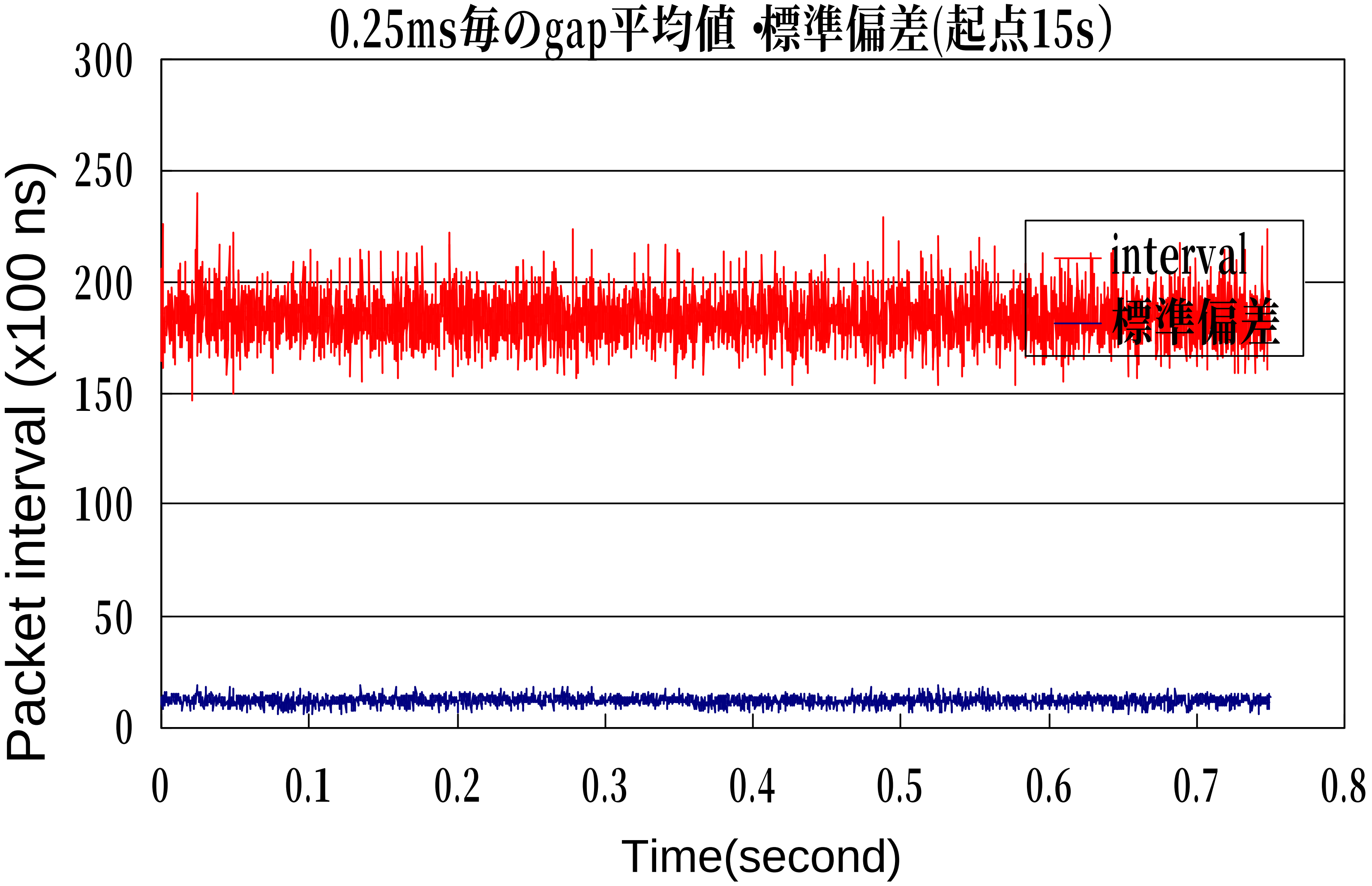
One way, 500ms network ：Data Packets@Sender-TX



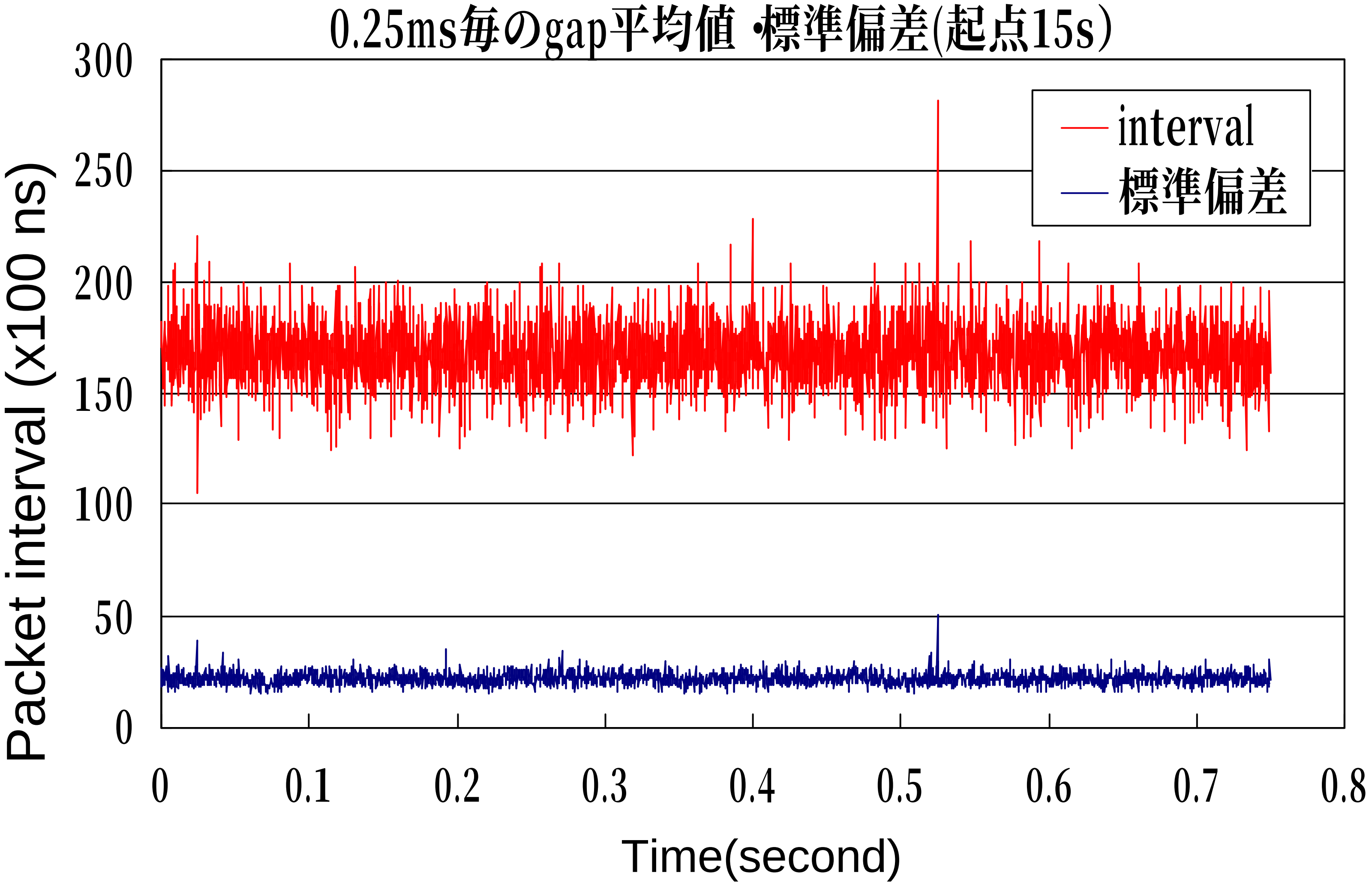
Data packets @Receiver-RX



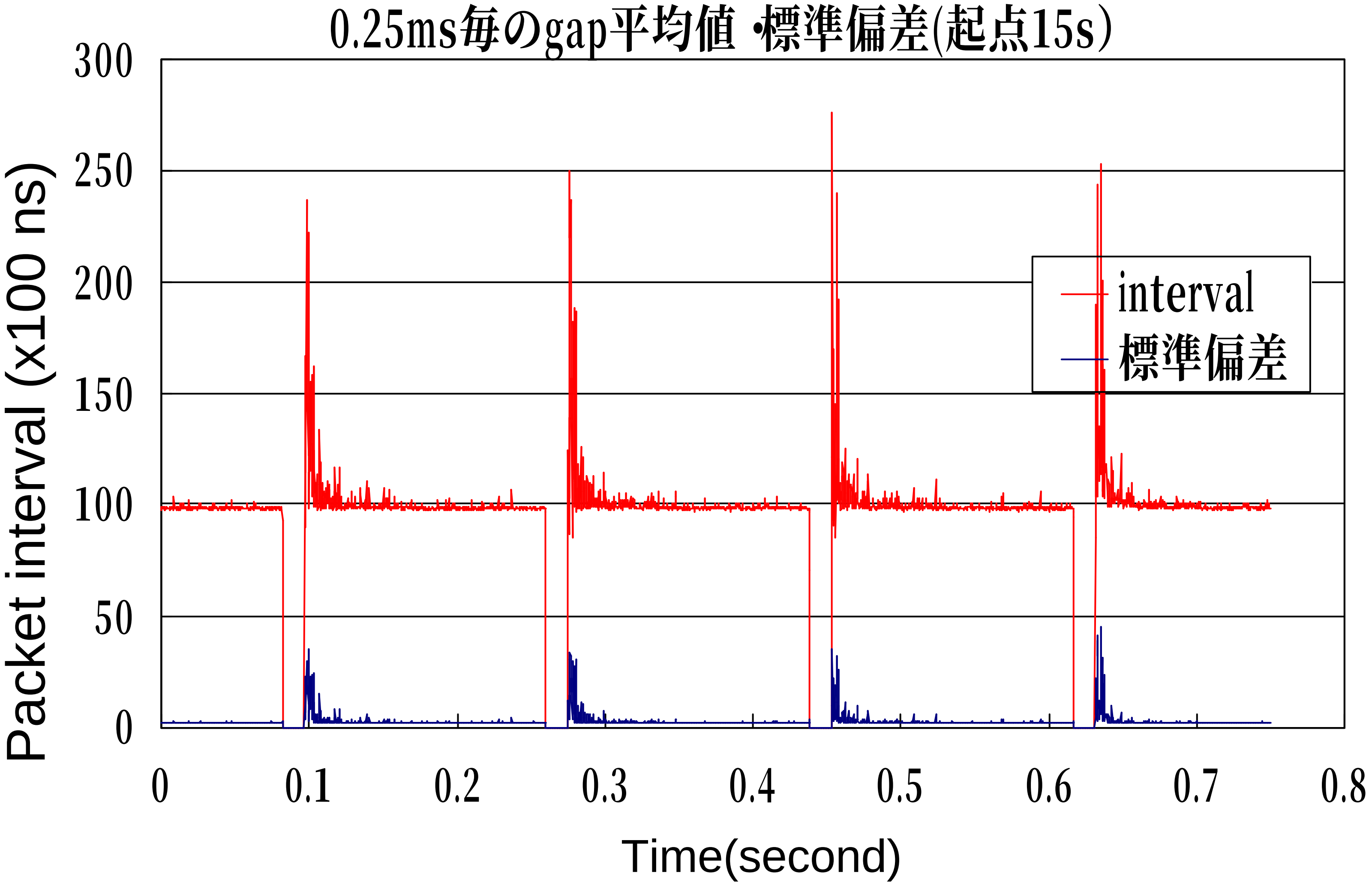
One way, 500ms ：ACKPackets@Receiver-TX



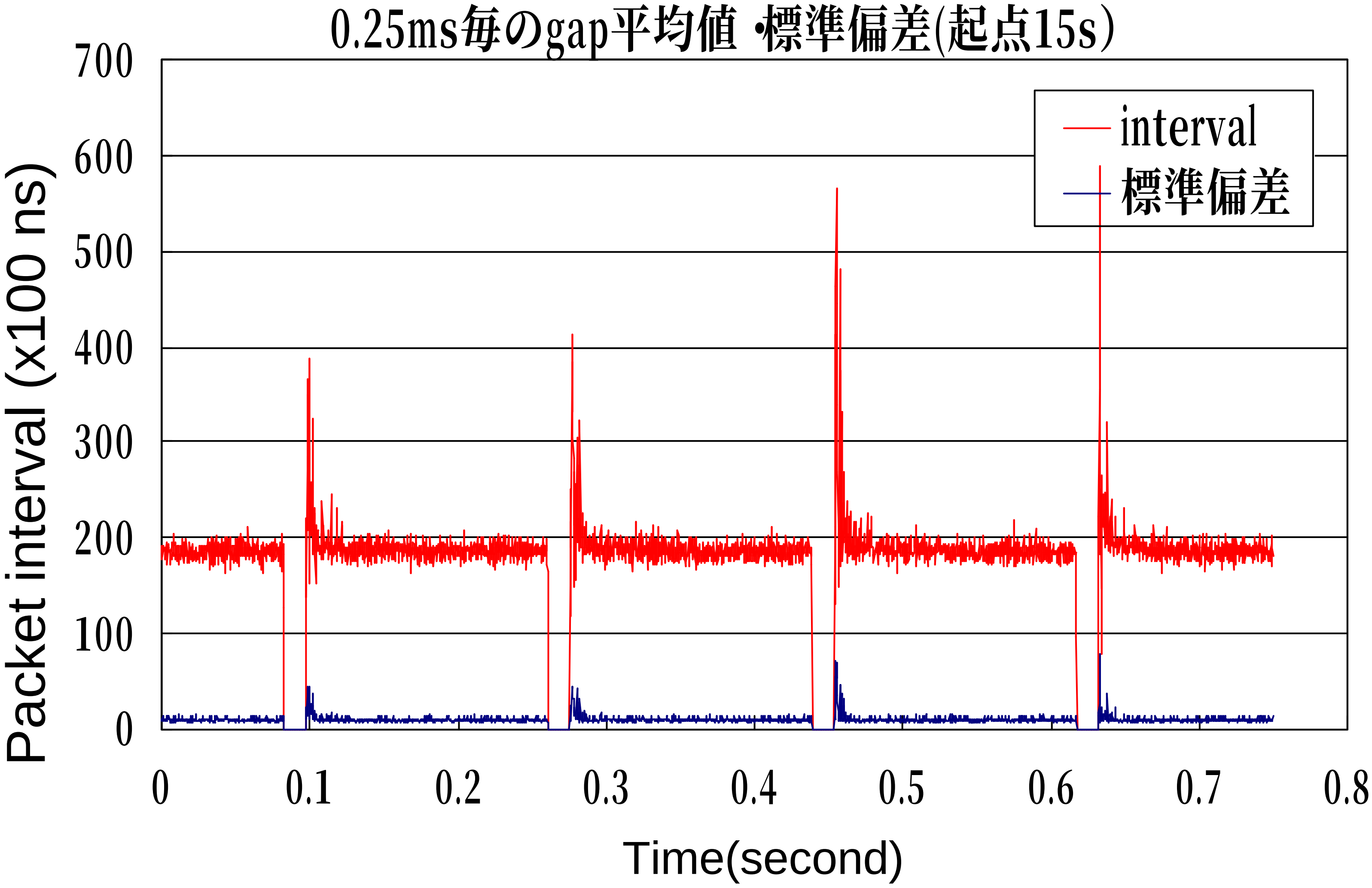
One way, 500ms RTT ：ACK packets@Sender-RX



Chicago loopback, 280ms :Data packets @Sender-TX

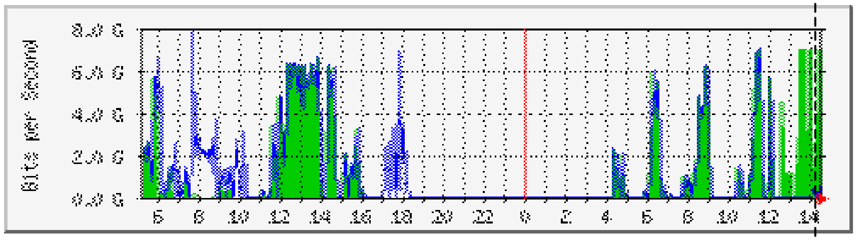


Chicago loopback ：ACK packets@Sender-RX

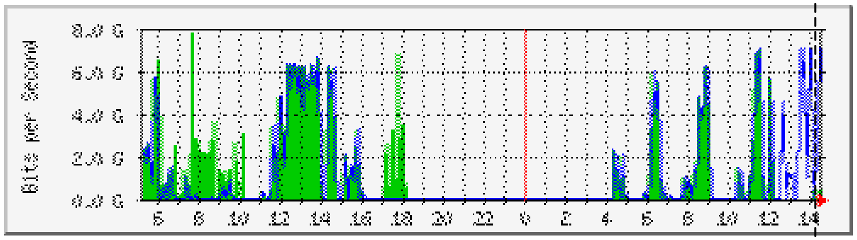


MRTG along the network path

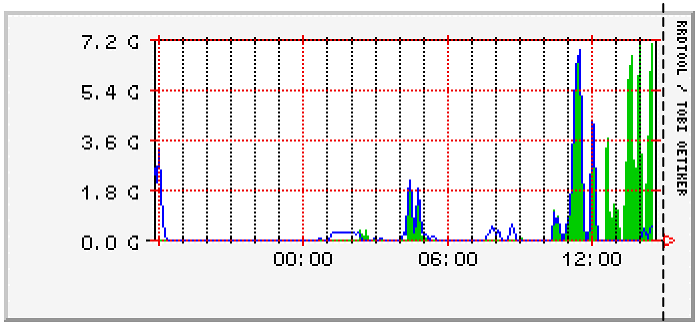
Measuing point



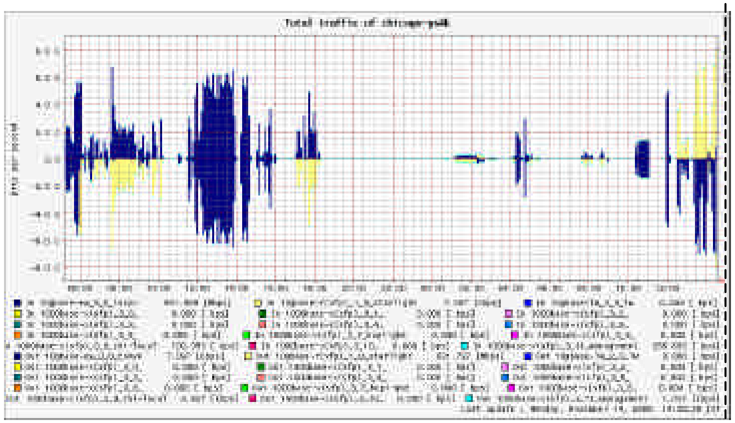
Seattle NI40G



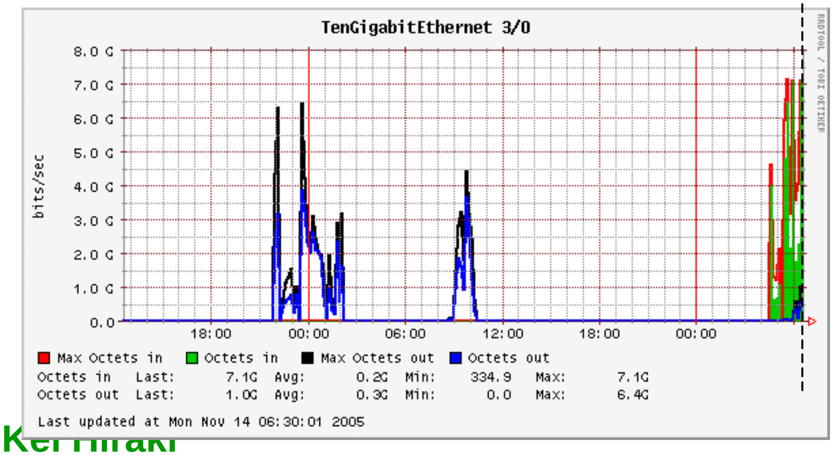
T-LEX output



NEZU NI40G
to JGN2



JGN2 Chicago GS4000

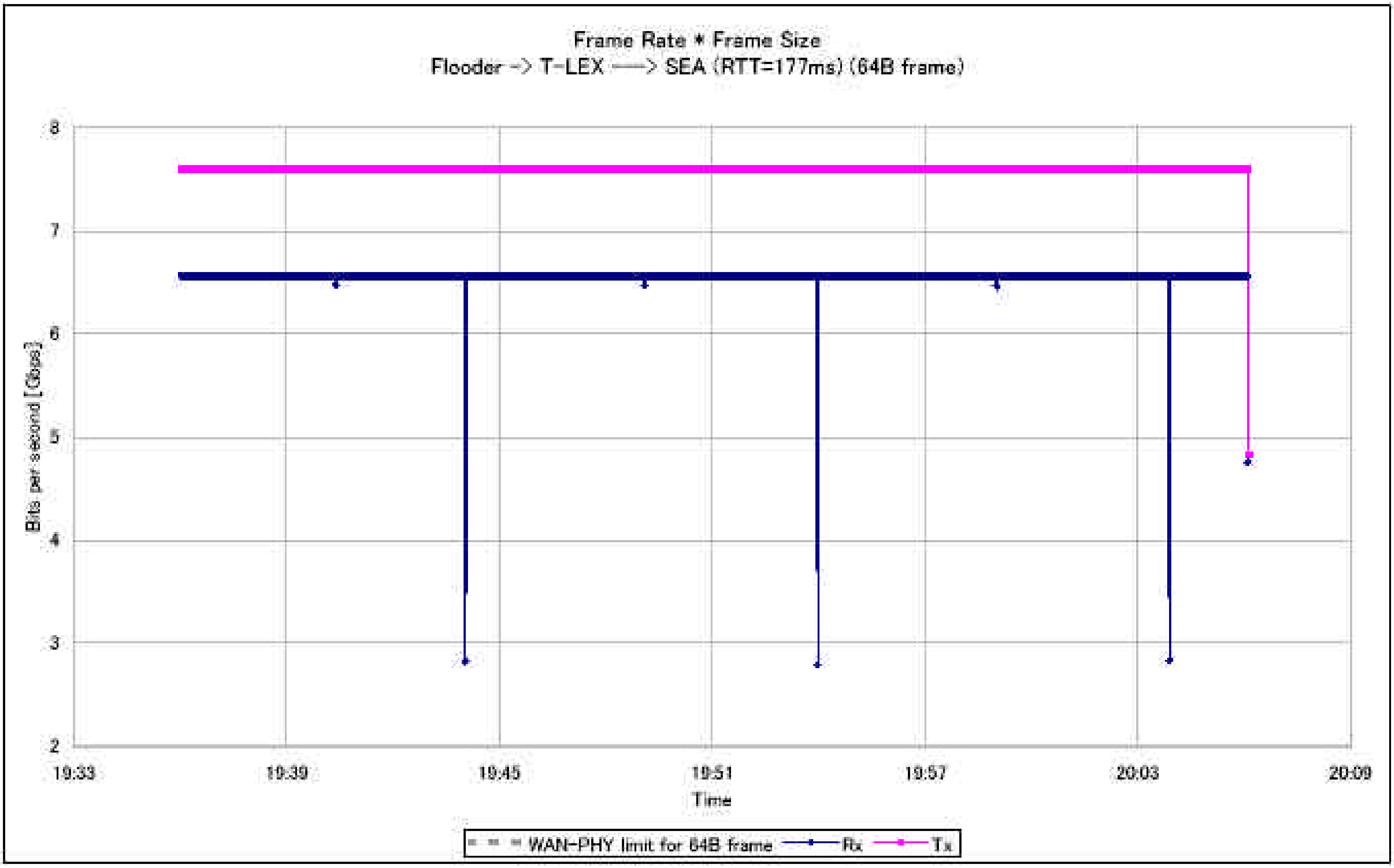


```
🟥 Max Octets in  🟩 Octets in  ■ Max Octets out  🟦 Octets out
Octets in    Last:      7.1G  Avg:       0.2G  Min:     334.9  Max:      7.1G
Octets out   Last:      1.0G  Avg:       0.3G  Min:       0.0  Max:      6.4G

Last updated at Mon Nov 14 06:30:01 2005
```

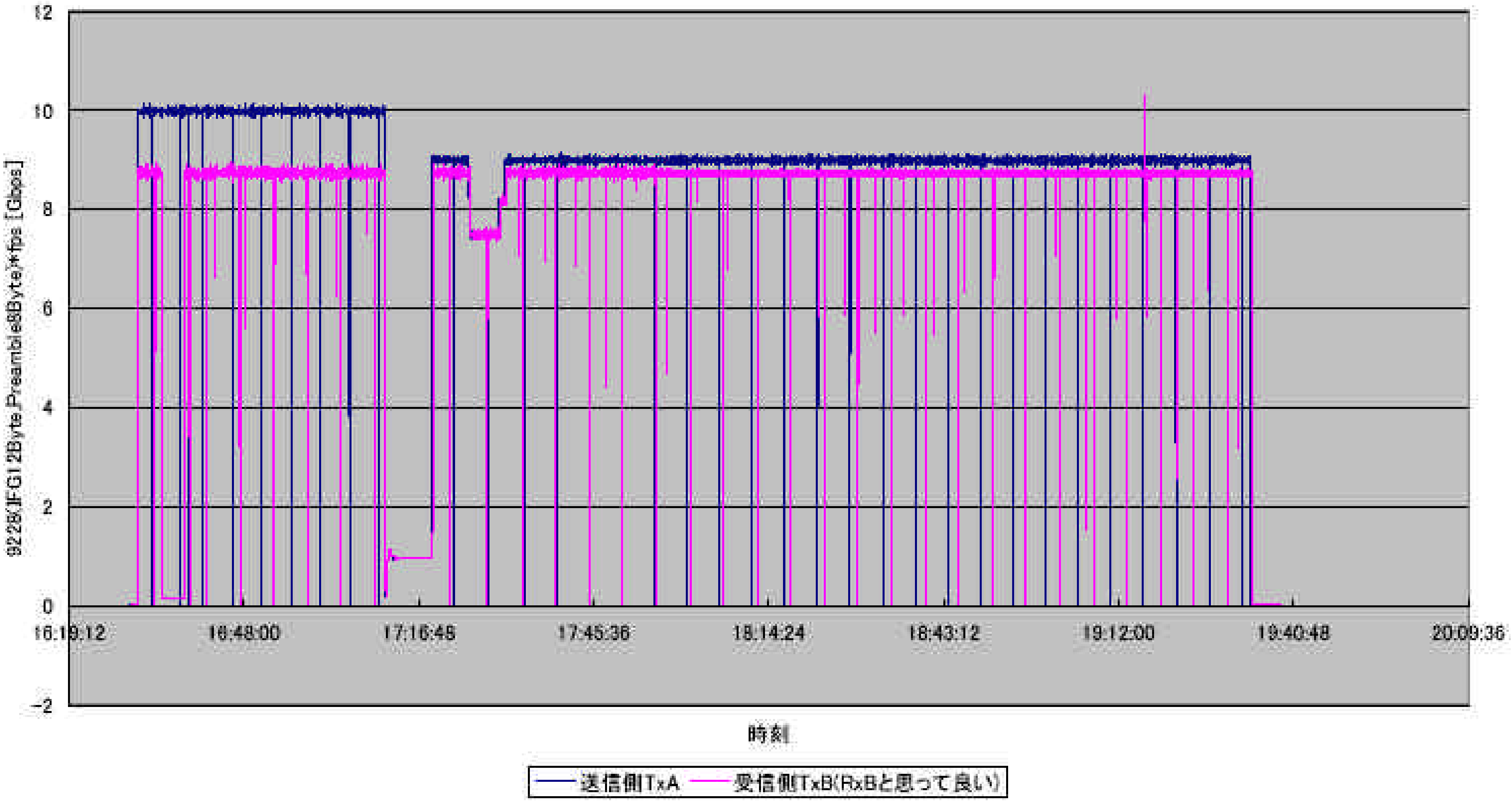
Amsterdam UvA Force10

Performance dip by MAC learning



Short(14+46+4=64Byte)フレームでの速度

Tx server -> NI40G -> F10 ——> Chicago F10



System for IPv6

- Server configuration
  - CPU: Dual AMD Opteron 248, 2.2GHz
  - Mother Board: RioWorks HDAMA rev. E
  - Memory: 1G bytes, Corsair Twinx CMX512RE-3200LLPT x 2, PC3200, CL=2
  - OS: Linux kernel 2.6.12
  - Disk Seagate IDE 80GB, 7200r.p.m. (disk speed not essential for performance)
- Network interface card
  - Chelsio T110 (10GBASE-SR), TCP offload OFF
  - PCI-X/133 MHz I/O bus connection

Technical points

- Traditional tuning and optimization methods
  - Tuning and optimization by Inter-layer coordination （Univ. of Tokyo)
  - Fine-grained pacing by offload engine
  - optimization at slow start phase
  - Utilization of flow control
  - Receiving side pacing

- Performance of single stream TCP on 30000 km circuit

  - 6.96 Gbps (TCP payload), standard Ethernet frame

    ⇒ 208,800 terabit meter / second

System for IPv4

- Server configuration
  - CPU: Quad Intel Xeon,3.6GHz
  - Server IBM x366
  - Memory: 32GB DDR2
  - OS: Windows Server 2003
  - Network interface card
  - Netrion Xframe II
  - PCI-X V2.0 266MHz I/O bus connection
  - No TOE function

Technical points

- Jumbo frame (9014B)

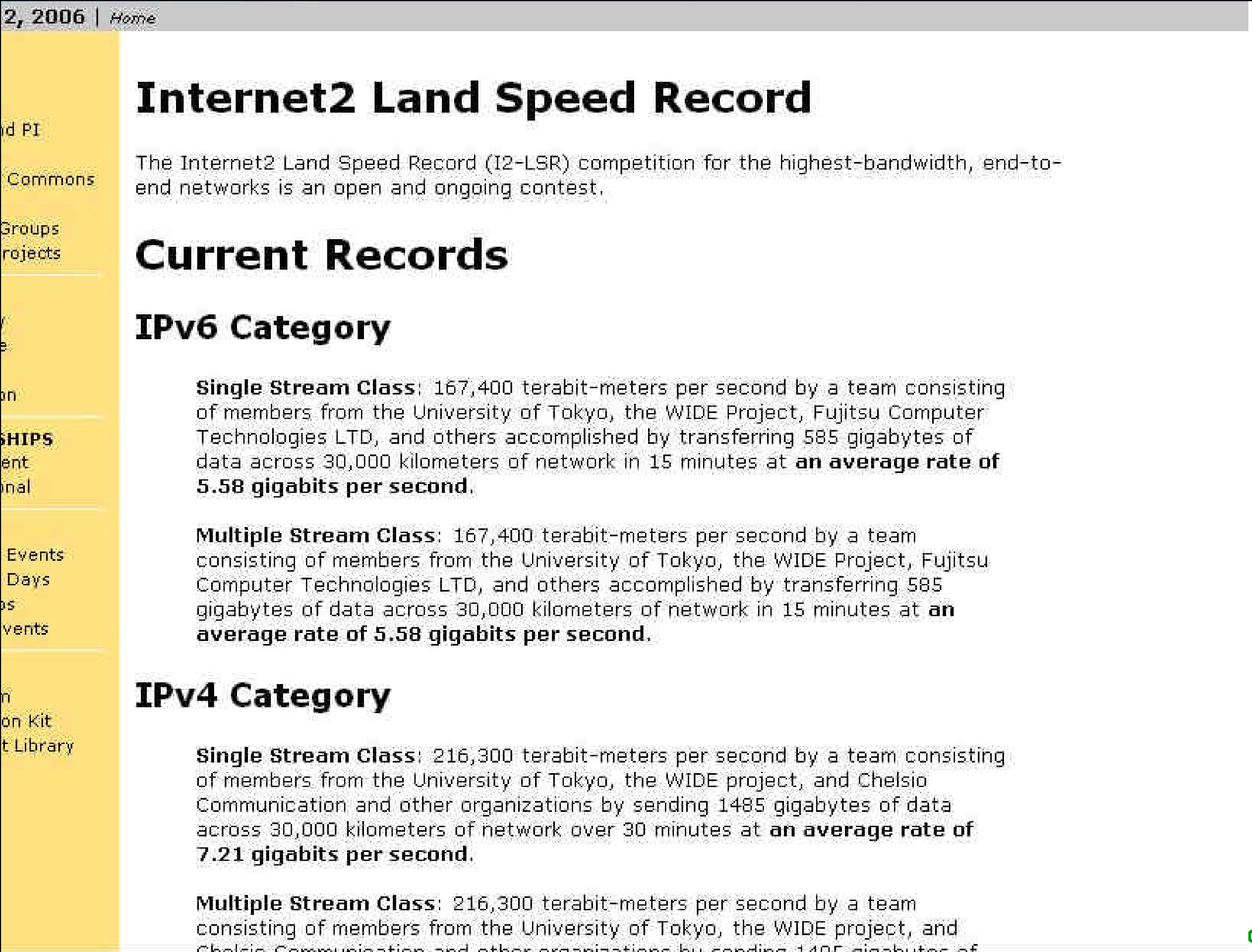
- Tuning and optimization by Inter-layer coordination (Univ. of Tokyo)
  - optimization at slow start phase
  - Utilization of flow control

- Performance of single stream TCP on 30000 km circuit

  - 7.96 Gbps (TCP payload), standard Ethernet frame

    ⇒ 239,820 terabit meter / second

  10% more than previous Land Speed Record



Internet2 Land Speed Record

The Internet2 Land Speed Record (I2-LSR) competition for the highest-bandwidth, end-to-end networks is an open and ongoing contest.

Current Records

IPv6 Category

Single Stream Class: 167,400 terabit-meters per second by a team consisting of members from the University of Tokyo, the WIDE Project, Fujitsu Computer Technologies LTD, and others accomplished by transferring 585 gigabytes of data across 30,000 kilometers of network in 15 minutes at **an average rate of 5.58 gigabits per second**.

Multiple Stream Class: 167,400 terabit-meters per second by a team consisting of members from the University of Tokyo, the WIDE Project, Fujitsu Computer Technologies LTD, and others accomplished by transferring 585 gigabytes of data across 30,000 kilometers of network in 15 minutes at **an average rate of 5.58 gigabits per second**.

IPv4 Category

Single Stream Class: 216,300 terabit-meters per second by a team consisting of members from the University of Tokyo, the WIDE project, and Chelsio Communication and other organizations by sending 1485 gigabytes of data across 30,000 kilometers of network over 30 minutes at **an average rate of 7.21 gigabits per second**.

Multiple Stream Class: 216,300 terabit-meters per second by a team consisting of members from the University of Tokyo, the WIDE project, and Chelsio Communication and other organizations by sending 1485 gigabytes of

Lessons learnt from LSR experiments

- Difficult combination -- WAN PHY, IPv6, Jumbo frame, L3 switching
  - There is no trouble-free switch
  - Unexpected packet losses by many reason
  - Further investigation on “Flow Control” is essential
- Unnecessary packet losses by packet clustering
  - Max receiving speed is normally less than wire speed
  - Behavior is different from switch to switch
  - Receiving side pacing is quite useful.
- Unnecessary packet losses by bursty TCP traffic
  - Sending side pacing is effective if the number of intermediate switch is small
  - Intermediate switches and routers erase effect of pacing
  - Difference between WAN PHY and LAN PHY may make trouble

Lessons learnt from LSR experiments

- Use of wide-area L2 network
  - Spanning Tree algorithm may make unstableness
  - MAC address learning may cause packet losses
  - Difficulty in debugging
  - Switches for trans-ocean (trans-pacific, etc.) should have very large buffers and pacing capability

- 10G network interface
  - Pacing capability is essential
  - Large input buffer (2*RTT*BW) or receiving side pacing is useful
  - Proper setting of window size, buffer size and queue length is essential

Toward end-to-end 10Gbps internet

- Purchase of 10Gbps NIC is always disappinting
  - Performance of 10 Gbps NIC may be worse than GbE
  - Packet losses!!

- Users want disk to disk, client to server performance
  - Disk performance
  - CPU bottleneck

- Importance of wire-rate switch
- Switches should be stable under any user packet sequences
- There may be a good switch and a bad switch. But the reason is unknown.

- Our next target is about 9Gbps.

Observation on Protocol

- We used traditional? TCP (loss-based)
  - If traffic is controlled, standard TCP works well
  - Selection of coefficients in AIMD algorithm may improve behavior
- Delay based TCP may not work properly
  - Major cause of variation of RTT is not congestion but meaningless jitter
  - Artificial packet clustering by intermediate switches/routers
  - Insufficient amount of intermediate buffers
- First several second may have problem
  - Experience from “Receiving side pacing”

Observation on Protocol

- Pacing is essential
  - Current switches/routers are not compatible to LFN
  - We need good cooperation algorithm between network layer and MAC layer

- Control of bursty behavior is also essential
  - Currently, users cannot control behavior of intermediate switches/routers

- Large buffer size at receiving NIC is very important
  - About RTT * BW size?

Conclusion

⑴We thanks all the people who support our experiments

Next target is 9Gbps through WAN PHY network

⑵Current 10Gbps devices and software technology is still far from satisfaction

Buffer size, burstness control, large-scale L2 network

⑶The next step is disk to disk transfer on single TCP stream

Optimization of cache memory and memory buses

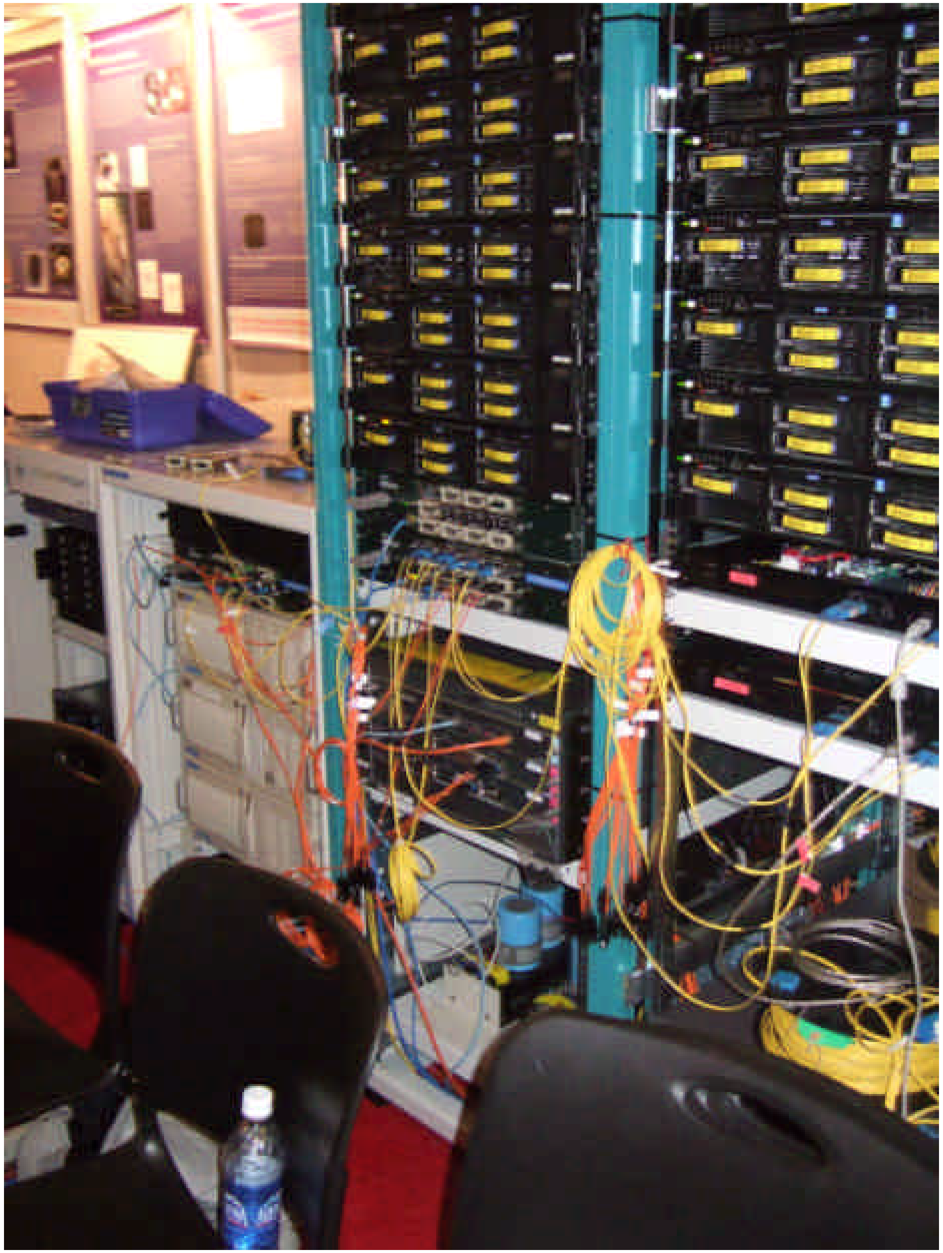
⑷After that, 10Gbps http service

⑸We are developing Hardware tools

- Wire-rate packet capture
- Traffic multiplier (for pushing networks)
- Packet filtering

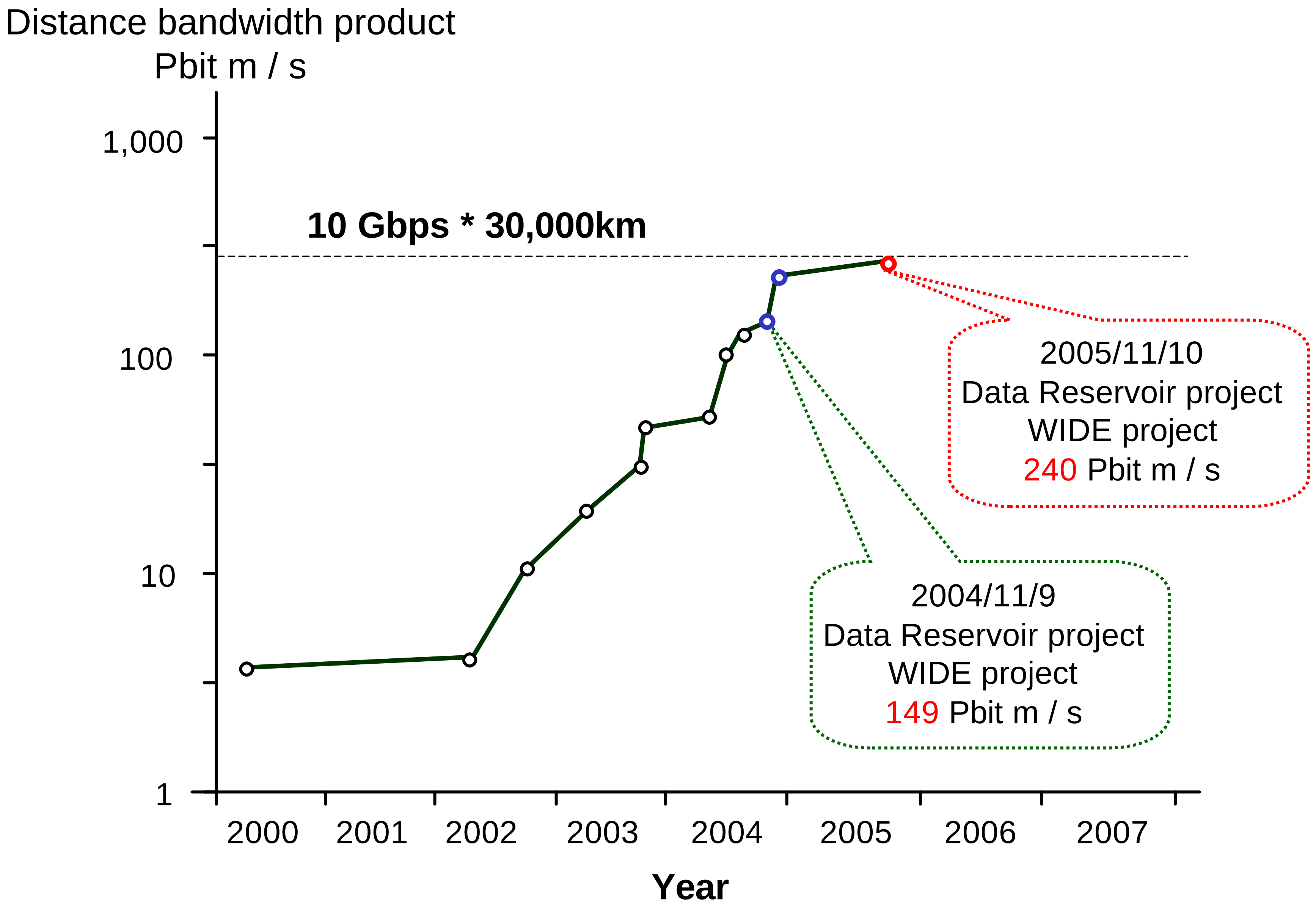
Photo of SC|05 University of Tokyo booth

Devices at SC|05



Devices at Pacific Northwest Gigapop

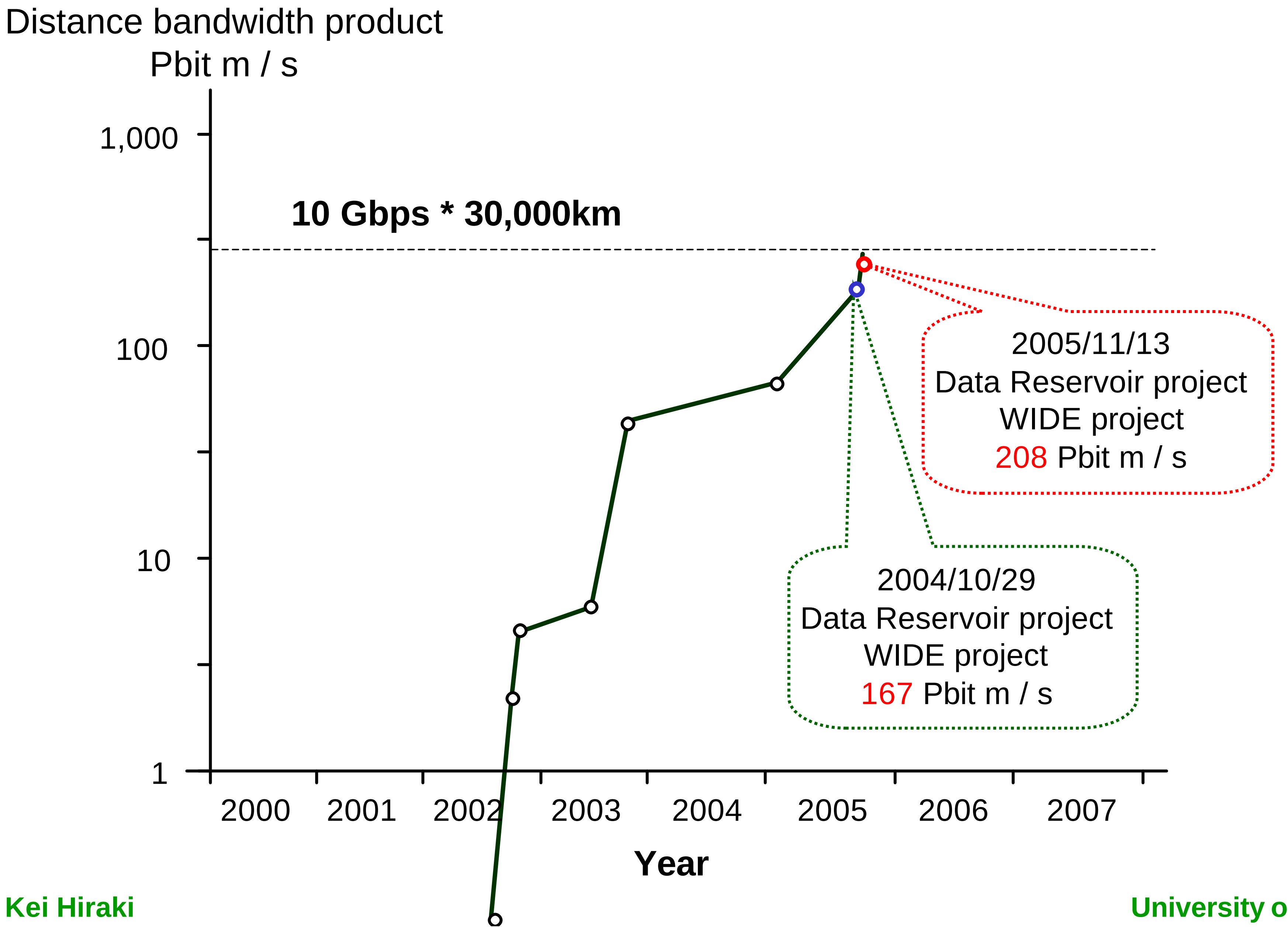
History of single-stream IPv4 Internet Land Speed Record



2005/11/10
Data Reservoir project
WIDE project
240 Pbit m / s

2004/11/9
Data Reservoir project
WIDE project
149 Pbit m / s

History of single-stream IPv6 Internet Land Speed Record



2005/11/13
Data Reservoir project
WIDE project
208 Pbit m / s

2004/10/29
Data Reservoir project
WIDE project
167 Pbit m / s

Thanks

JGNII

STARLIGHT℠

WIDE PROJECT

SURFnet

NTT Communications

Microsoft

TλEx Tokyo Lambda EXchange

SCinet

IEEAF

PACIFIC NORTHWEST GIGAPOP

CANARIE

sara Reken- en Netwerkdiensten

Chelsio Communications Accelerate

FOUNDRY NETWORKS

CISCO SYSTEMS

ANUE Network Emulators

PFU a Fujitsu company

Land Speed Recordの歴史

年月日	バンド幅距離積 テラビット・メートル／秒	研究グループ	クラス
2000年3月21日	4278	Microsoft Qwest Communications University of Washington USC Information Sciences Institute	IPv4　Single Stream
2000年3月29日	5384	Microsoft Qwest Communications University of Washington USC Information Sciences Institute	IPv4　Multiple Stream
2002年4月9日	4933	University of Alaska at Fairbanks Faculty of Science of the University of Amsterdam SURFnet	IPv4　Single Stream
2002年8月22日	40	Oregon Gigapop NYSERNet University of Oregon	IPv6　Multiple Stream
2002年9月27日	2517	ARNES DANTE RedIRIS	IPv6　Single Stream
2002年10月9日	5154	ARNES DANTE RedIRIS	IPv6　Single Stream
2002年11月19日	10136	Nationaal Instituut voor Kernfysica en Hoge-Energiefysica (NIKHEF) Universiteit van Amsterdam (UvA) Stanford Linear Accelerator Center (SLAC) California Institute of Technology (Caltech)	IPv4　Single and Multiple stream
2003年2月23日	23888	California Institute of Technology (Caltech) CERN Los Alamos National Laboratory (LANL) Stanford Linear Accelerator Center (SLAC)	IPv4　Single and Multiple stream

年月日	バンド幅距離積テラビット・メートル／秒	研究グループ	クラス
2003年5月6日	6947	California Institute of Technology (Caltech) CERN	IPv6　Single and Multiple stream
2003年10月10日	38420	California Institute of Technology (Caltech) CERN	IPv4　Single and Multiple stream
2003年11月11日	61752	California Institute of Technology (Caltech) CERN	IPv4　Single and Multiple stream
2004年2月22日	68431	California Institute of Technology (Caltech) CERN	IPv4　Multiple stream
2004年4月14日	69073	SUNET Sprint	IPv4　Single stream
2004年5月6日	77699	California Institute of Technology (Caltech) CERN	IPv4　Multiple stream
2004年6月25日	104529	California Institute of Technology (Caltech) CERN	IPv4　Multiple stream
2004年6月28日	103583	California Institute of Technology (Caltech) CERN	IPv4　Single stream

年月日	バンド幅距離積 テラビット・メートル／秒	研究グループ	クラス
2004年9月12日	124935	SUNET Sprint	IPv4　Single and Multiple stream
2004年11月8日	184877	California Institute of Technology (Caltech) CERN CENEC	IPv4　Multiple stream
2004年11月９日	148850	University of Tokyo Fujitsu Computer Technologies WIDE Chelsio Communications	IPv4　Single stream
2004年12月25日	216300	University of Tokyo Fujitsu Computer Technologies WIDE Chelsio Communications APAN JGN2 CANARIE SURFnet Universiteit van Amsterdam	IPv4　Single and Multiple stream

年月日	バンド幅距離積 テラビット・メートル／秒	研究グループ	クラス
2005年1月19日	72225	CERN CALTECH	IPv6　Single and Multiple stream
2005年10月29日	167400	University of Tokyo Fujitsu Computer Technologies WIDE Chelsio Communications JGN2	IPv6　Single and Multiple stream
2005年11月11日	239820	University of Tokyo Fujitsu Computer Technologies WIDE Microsoft JGN2 CANARIE SURFnet Universiteit van Amsterdam	IPv4　Single and Multiple stream
2005年11月13日	**208800**	University of Tokyo Fujitsu Computer Technologies WIDE Chelsio Communications JGN2 CANARIE SURFnet Universiteit van Amsterdam	IPv6　Single and Multiple stream

Future of Internet2 Land Speed Record

- 10Gbps era.
  - I/OBus bottleneck
    - PCI-express or PCI-X 266MH z
    - WAN PHY limit is about 9.1Gbps (TCP payload bandwidth)
- 40Gbps era.
  - 40Gbps WAN will be established in several years
    - At first, domestic network in Japan or US
    - 7500 km is necessary to make new record
  - 40GbpsLAN and NIC is still unpredictable
    - 40Gbps Ethernet or 100Gbps Ethernet
    - We need another new I/O bus
- 100Gbps era.
  - 100GbpsEthernet standard. When?
  - Yet another I/O bus and CPU will be definitely necessary
  - Year 2010? 2015?