# Quality of Service on Linux for the Atlas TDAQ Event Building Network

Y.Yasu(1), A.Manabe(1), Y.Nagasaka(2), H.Hasegawa(3), M.Shimojima (4), M.Nomachi(5), H.Fujii(1) and Y.Watase(1)
on behalf of the Atlas Trigger/DAQ group

(1)High Energy Accelerator Research Organization (KEK)
    Oho 1-1, Tsukuba, Ibaraki 305 Japan
(2) Department of Information and Intellectual Systems Engineering Faculty
    of Engineering Hiroshima Institute of Technology
    2-1-1 Miyake, Saeki-ku, Hiroshima 731-5193 Japan
(3)Faculty of Science, Shinshu University
    Asahi 3-1-1, Matsumoto, 390-8621 Nagano Japan
(4)Department of Electrical Engineering
    Nagasaki Institute of Applied Science (NIAS)
    536 Aba-machi, Nagasaki 851-01 Japan
(5)Graduate School of Science, Osaka University
    Machikaneyama 1-1, Toyanaka-city, 560-0043 Osaka Japan

# Contents

**Why QoS is necessary**

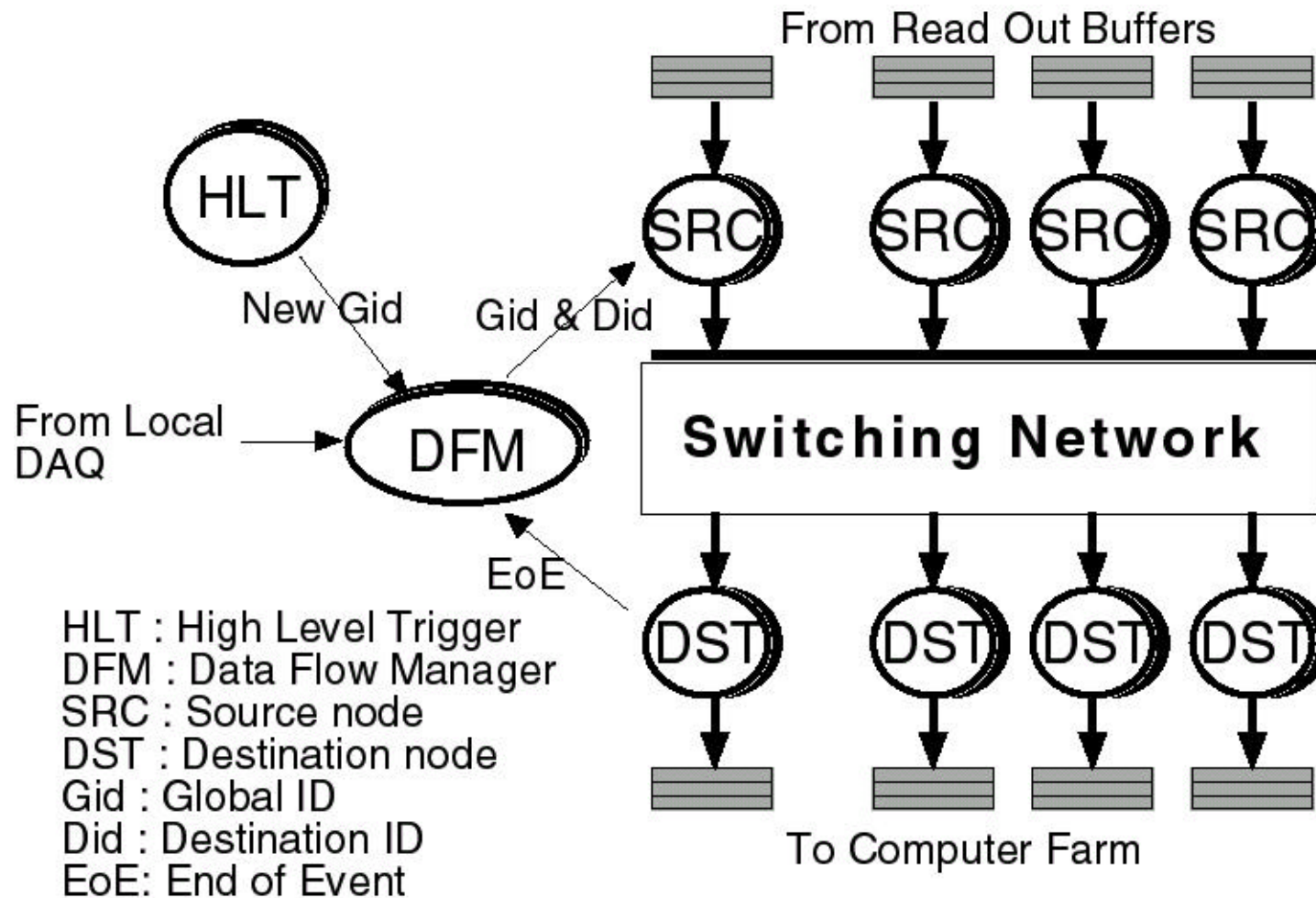**What QoS is**

**Performance Measurements**
    **Packet loss on UDP/IP multicast transfer**
    **QoS software overhead on TCP/IP transfer**
    **Packet distribution on TCP/IP transfer**

**Conclusion**

# ATLAS Event Builder System

From Read Out Buffers

HLT

New Gid          Gid & Did

SRC     SRC     SRC     SRC

From Local
DAQ                    DFM

Switching Network

EoE

HLT : High Level Trigger
DFM : Data Flow Manager
SRC : Source node
DST : Destination node
Gid : Global ID
Did : Destination ID
EoE: End of Event

DST     DST     DST     DST

To Computer Farm

# Why QoS is necessary

> UDP/IP multicast transfer is attractive for broadcasting messages
   to Event Builder Sources from scalability point of view,
   but UDP/IP may lose the messages(packets).



**Data Flow
Manager**

**Event Builder
Sources**
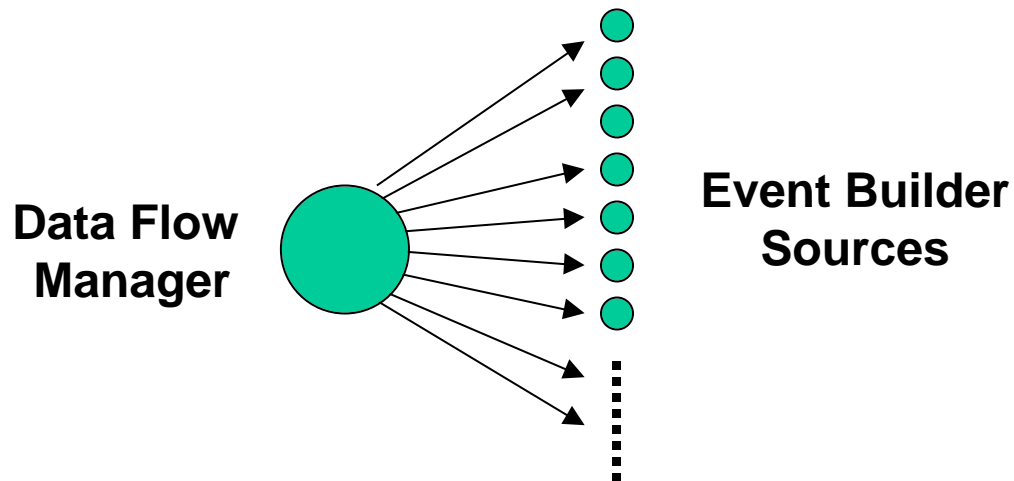
> Constant Bit Rate of ATM network can eliminate the packet loss
   by managing the bandwidth.
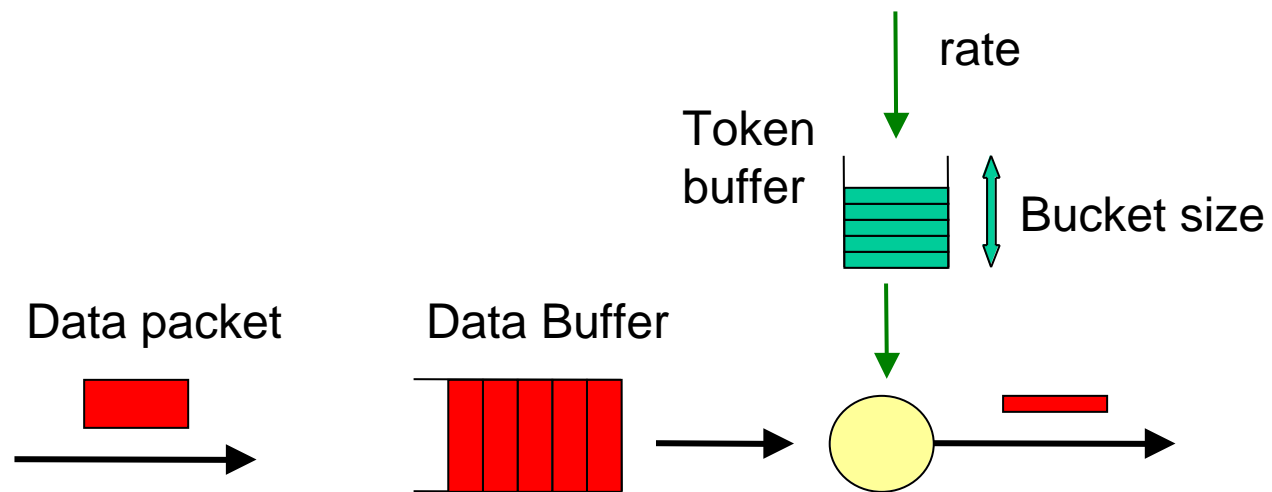> QoS in Linux kernel can manage the bandwidth.

Can the QoS eliminate the packet loss?
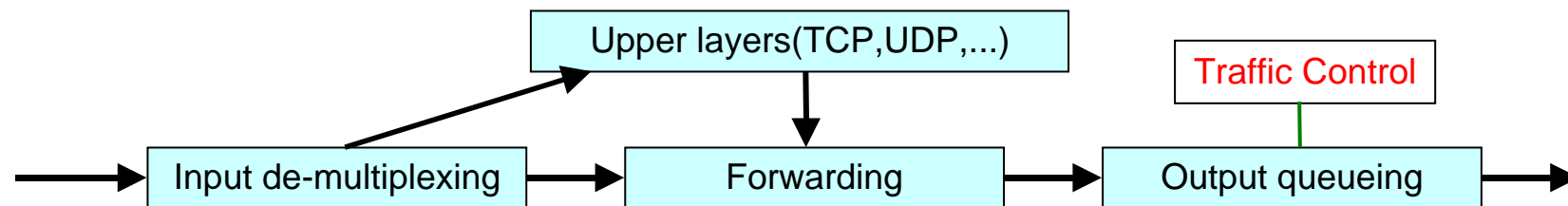
# What QoS is:Class-Based Queueing (CBQ)
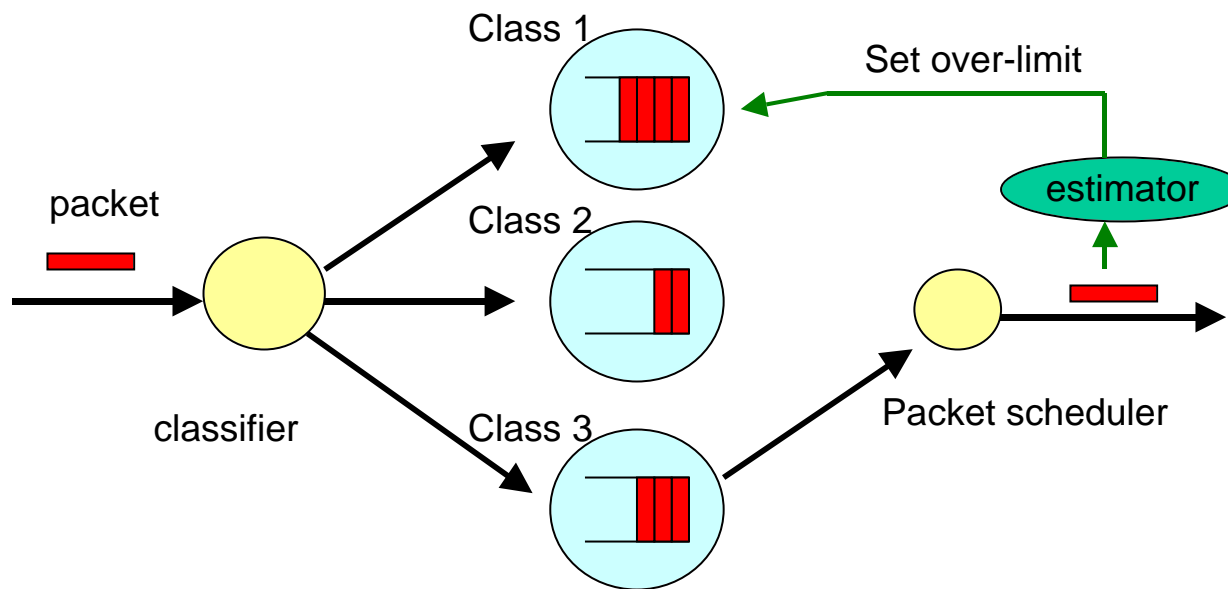


CBQ can classify incoming packets into multiple classes.
CBQ can share and limit the transfer rate.

# What QoS is :Token Bucket Filter (TBF)

rate

Token
buffer

Bucket size

Data packet

Data Buffer

TBF can limit the transfer rate according to the token rate

# What QoS is :Traffic control in Linux kernel

Upper layers(TCP,UDP,...)

Traffic Control

Input de-multiplexing → Forwarding → Output queueing

Traffic control is done only at output queueing.

Outgoing packet is only controlled.

# What QoS is :Queueing discipline for Linux
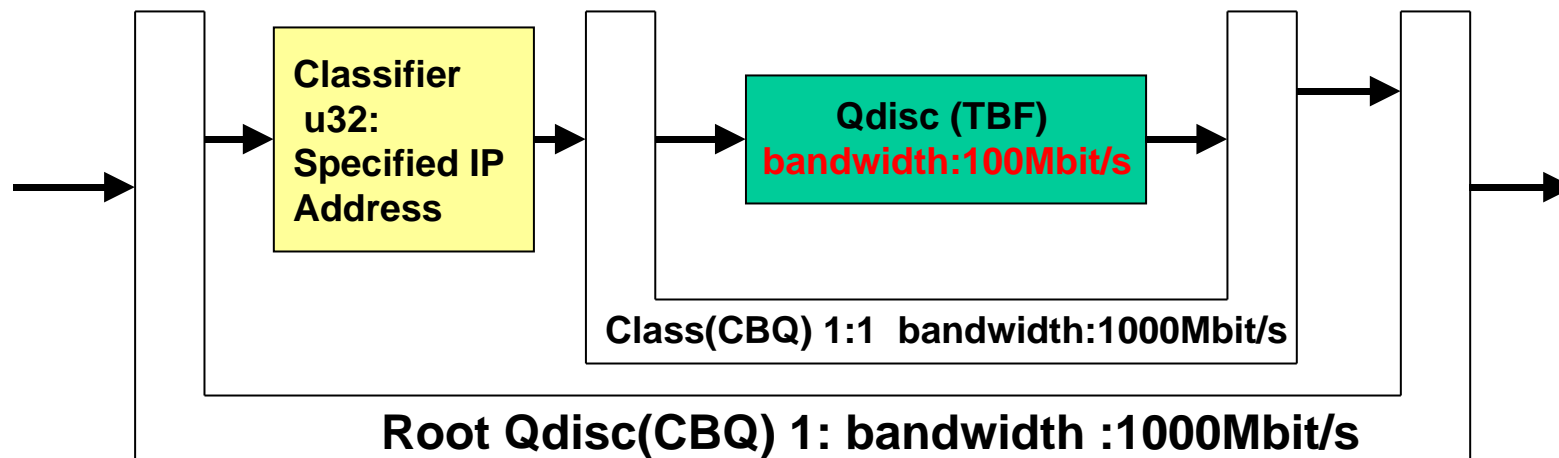


U32 can specify IP address to manage the bandwidth.

TBF limits the bandwidth to 100Mbit/s on 1000Mbit/s network
at this example.

# Performance measurement

**Packet loss measurement**

**Measurement of QoS software overhead**

**Packet distribution measurement**

# Testbed

3Com Super StackII Switch 9300
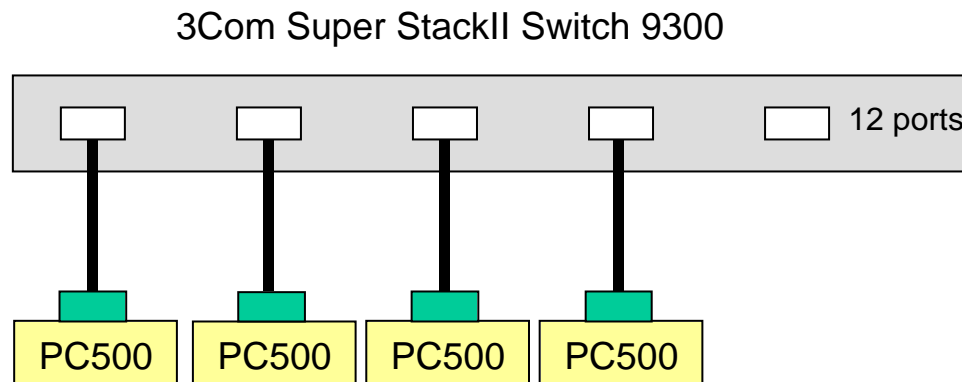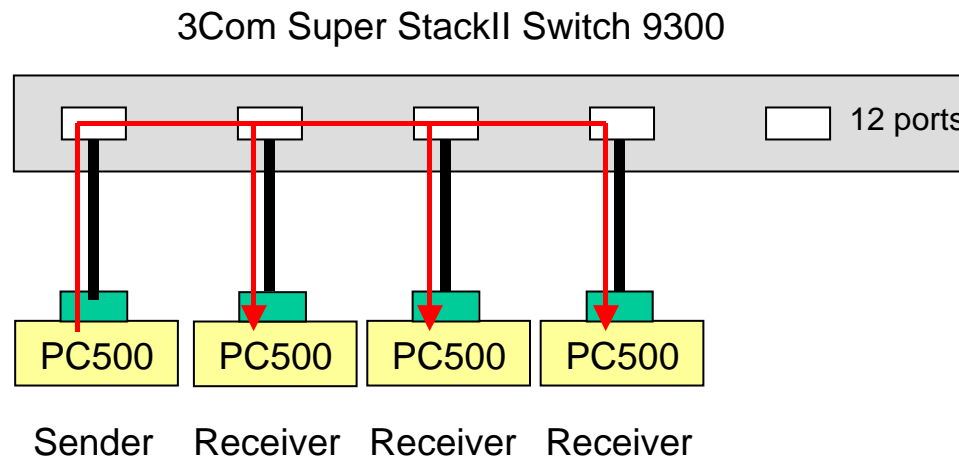


## Configuration of PC500 system

CPU:        PentiumIII/500MHz          Linux kernel version :   2.4.5
Chipset     440GX                      gcc version:             egcs-2.91-66
Memory:     100MHz/SDRAM/256MB         acenic driver version :  0.8
PCIbus:     32-bit/33MHz
NIC:        AceNIC(1MB)
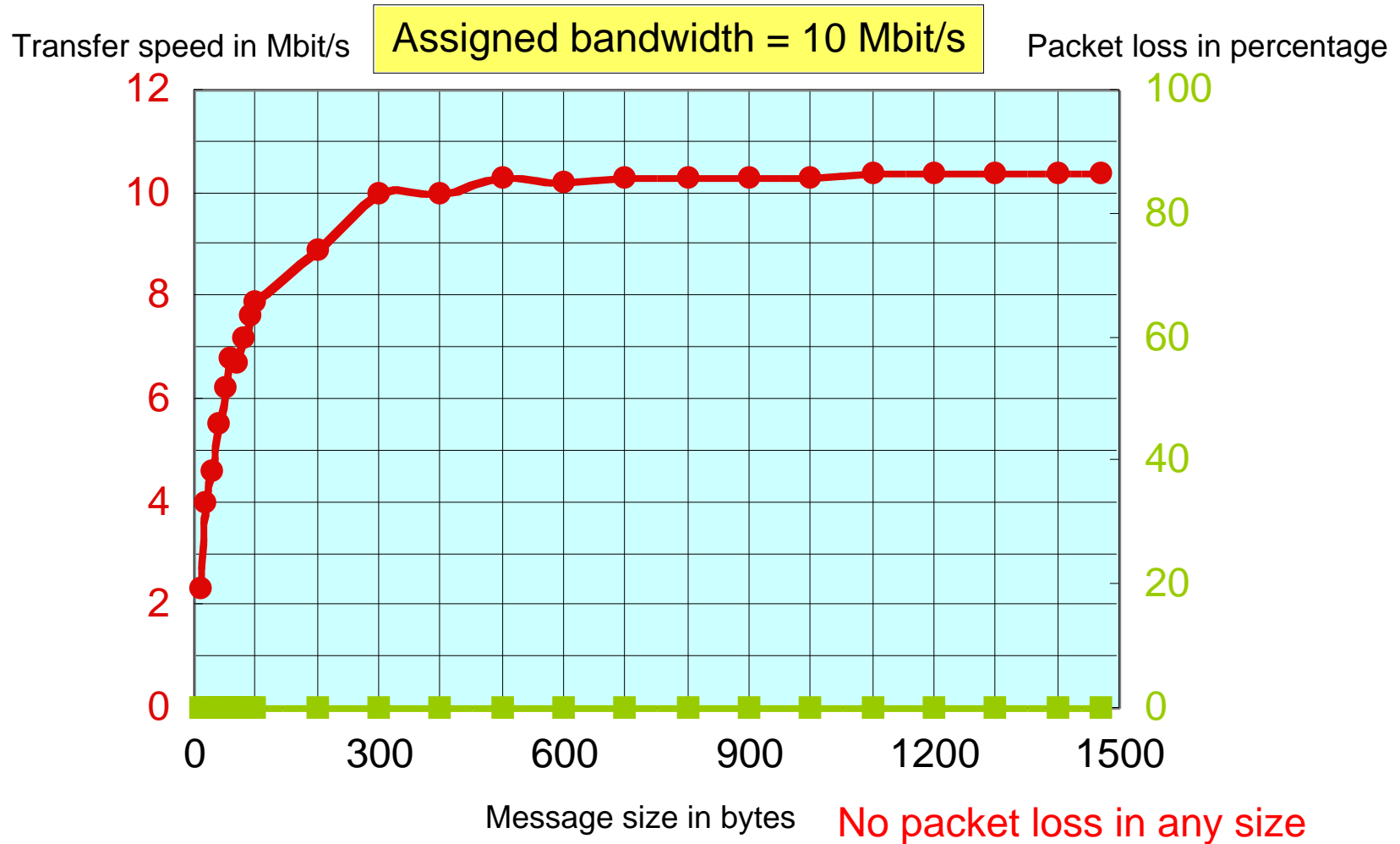
# Packet loss on
# UDP/IP multicast transfer

3Com Super StackII Switch 9300



12 ports

PC500    PC500    PC500    PC500

Sender    Receiver    Receiver    Receiver

Sender sends message with 10 bytes to receivers in best effort.

# Packet loss on UDP/IP multicast transfer (Cont.)
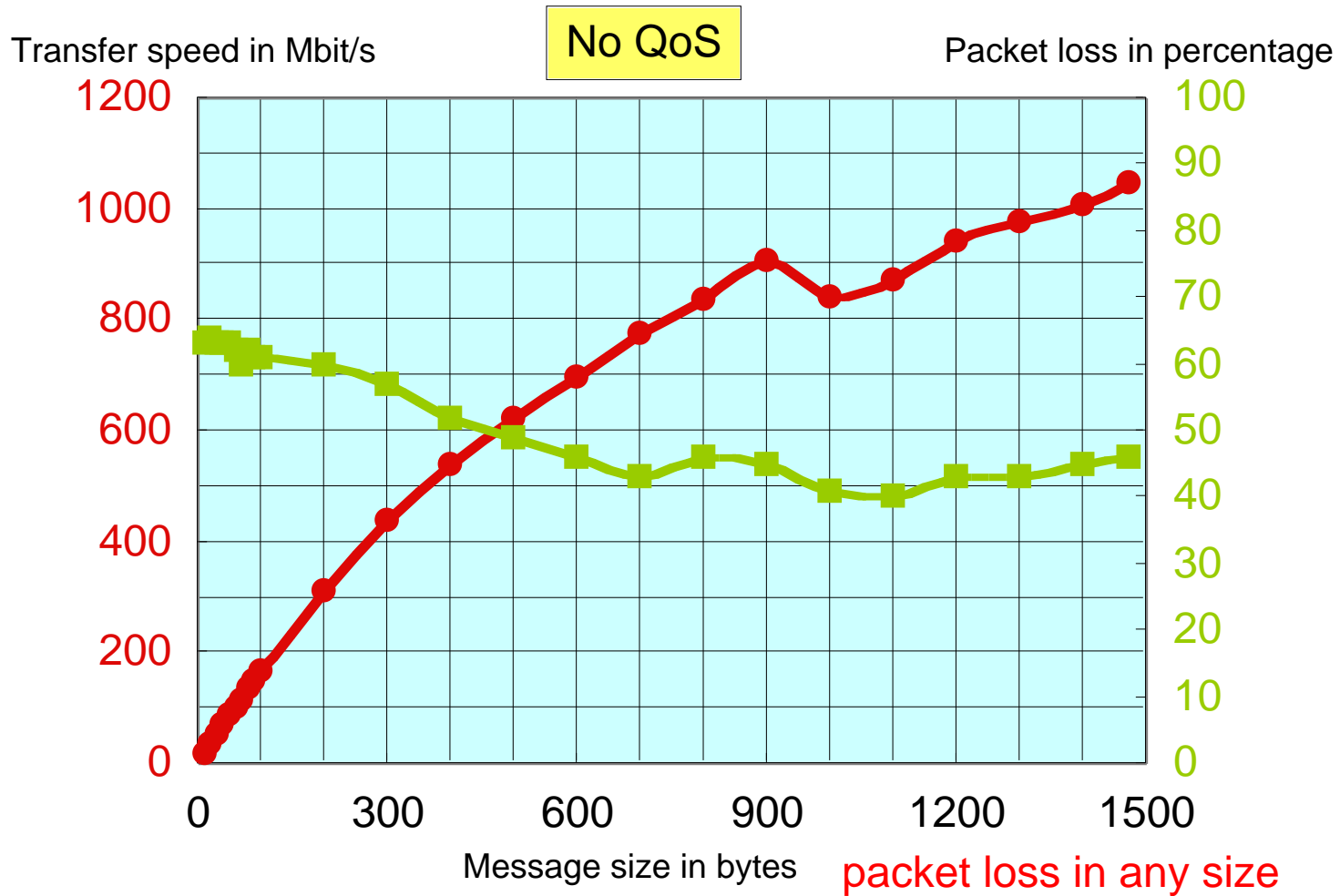
Transfer speed in Mbit/s

No QoS

Packet loss in percentage

Message size in bytes

packet loss in any size

# Packet loss on
# UDP/IP multicast transfer(Cont.)

Transfer speed in Mbit/s

Assigned bandwidth = 10 Mbit/s

Packet loss in percentage



Message size in bytes

No packet loss in any size

# Packet loss on UDP/IP multicast transfer(Cont.)

Transfer speed in Mbit/s

Assigned bandwidth = 50 Mbit/s

Packet loss in percentage
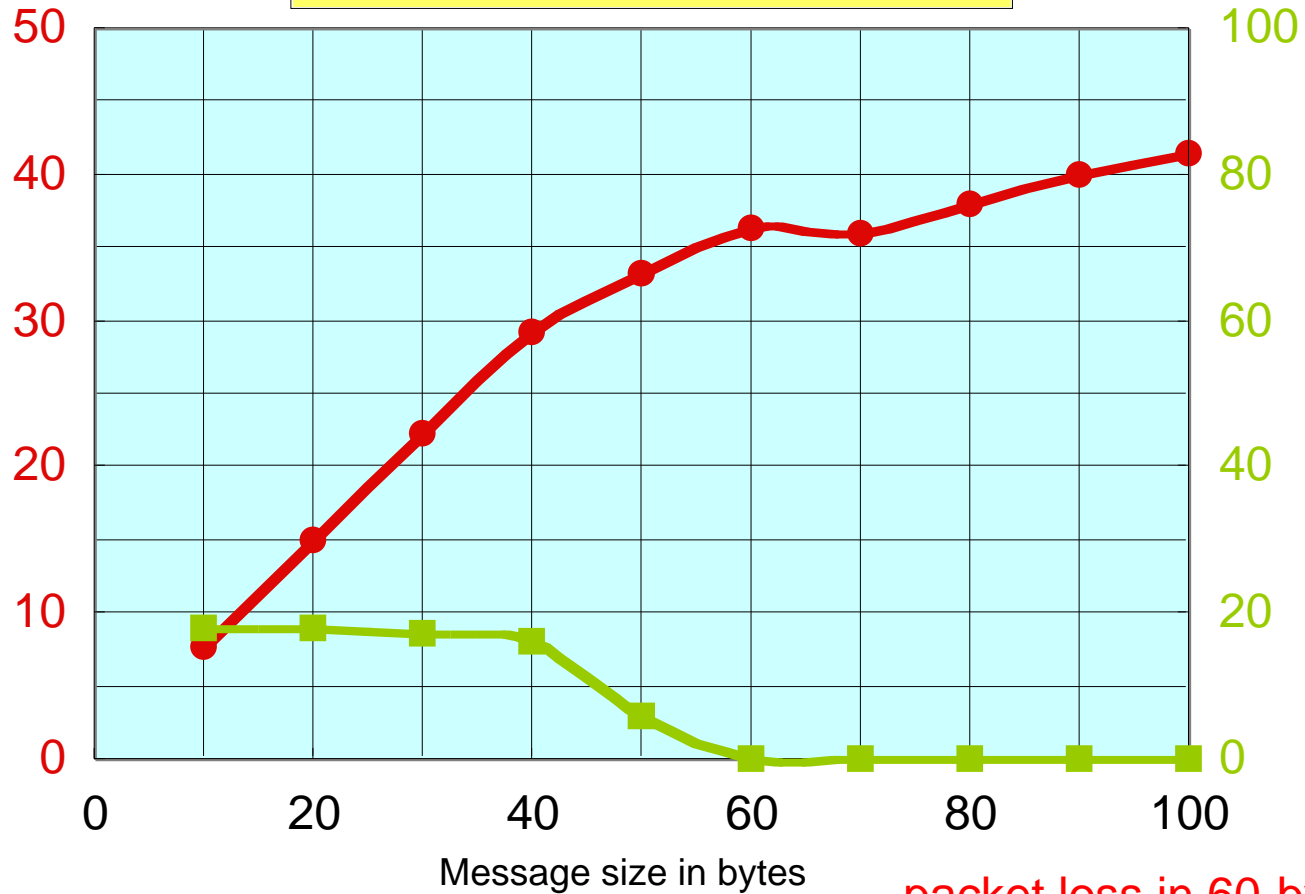
Message size in bytes

packet loss in 60-byte size

# Packet loss on
# UDP/IP multicast transfer(Cont.)

Transfer speed in Mbit/s

Assigned bandwidth = 100 Mbit/s

Packet loss in percentage

Message size in bytes   No packet loss out of 200-byte size

# QoS Software Overhead

3Com Super StackII Switch 9300
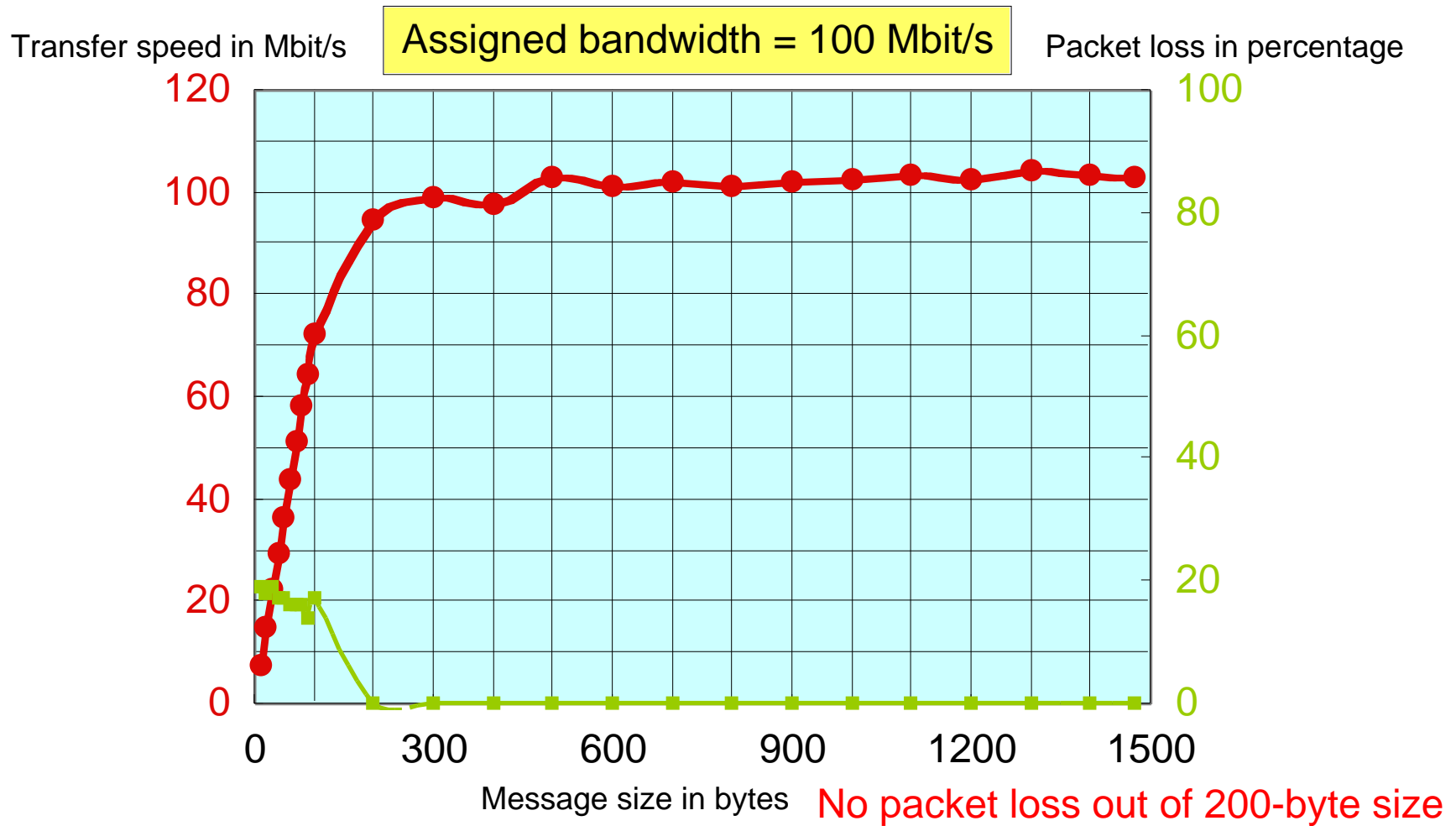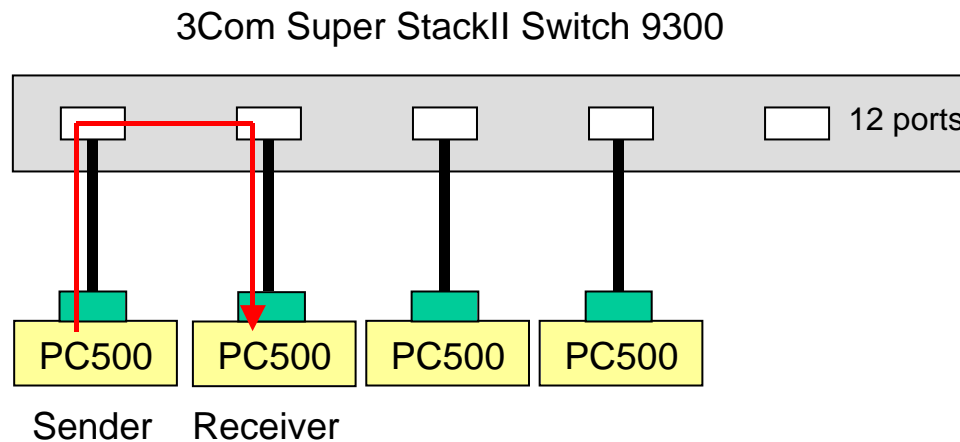


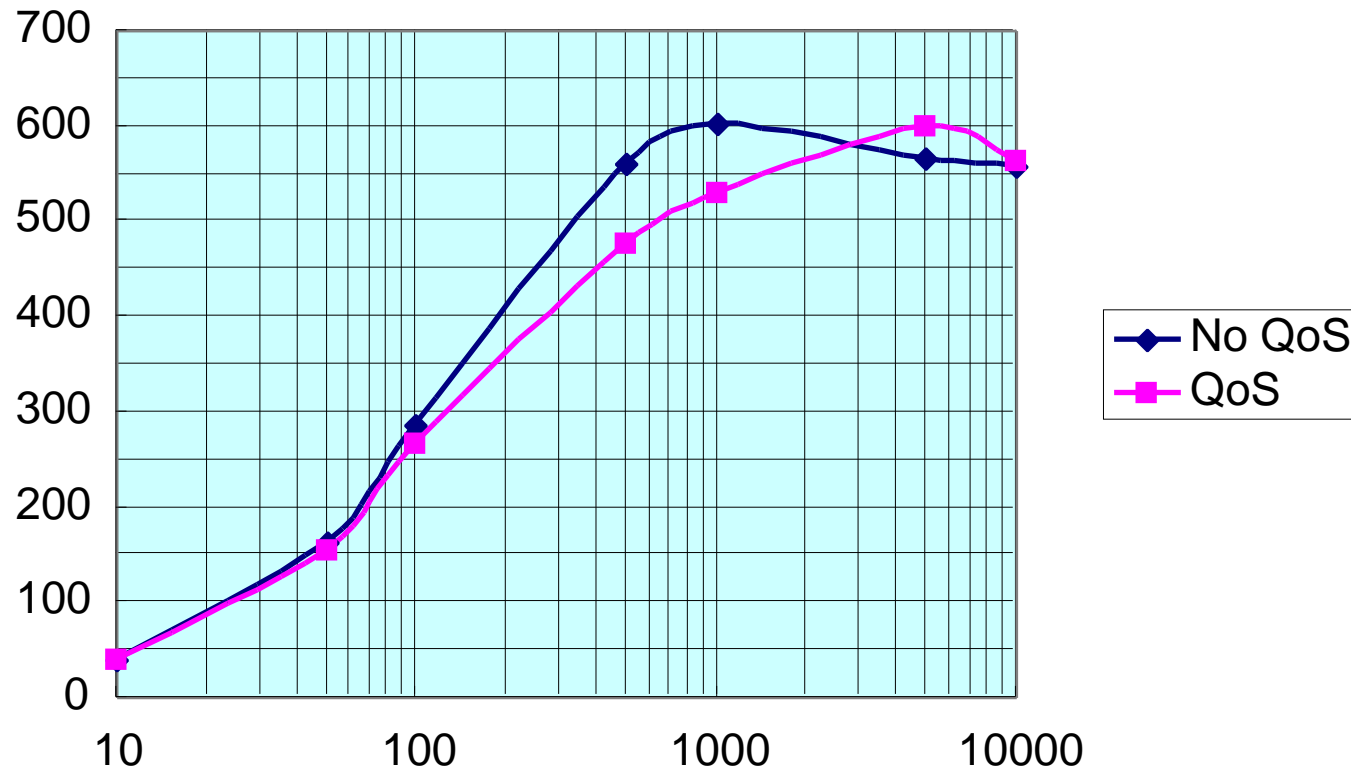Sender    Receiver

Assigned bandwidth is 1000Mbit/s.

Sender sends message with 10 bytes to a receiver in best effort.

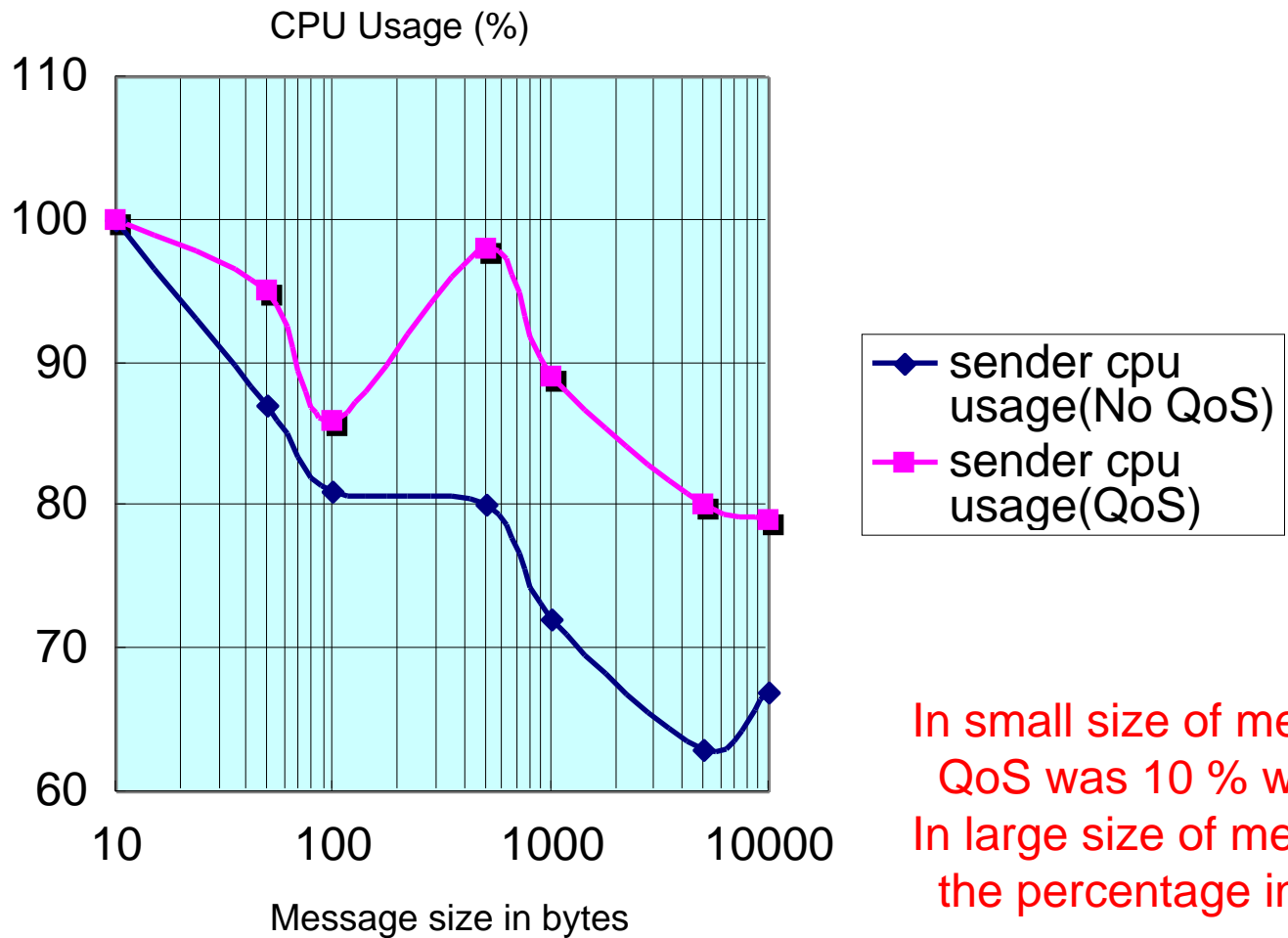This bandwidth does not limit the transfer, but it consumes CPU time.

# QoS Software Overhead: TCP/IP transfer speed
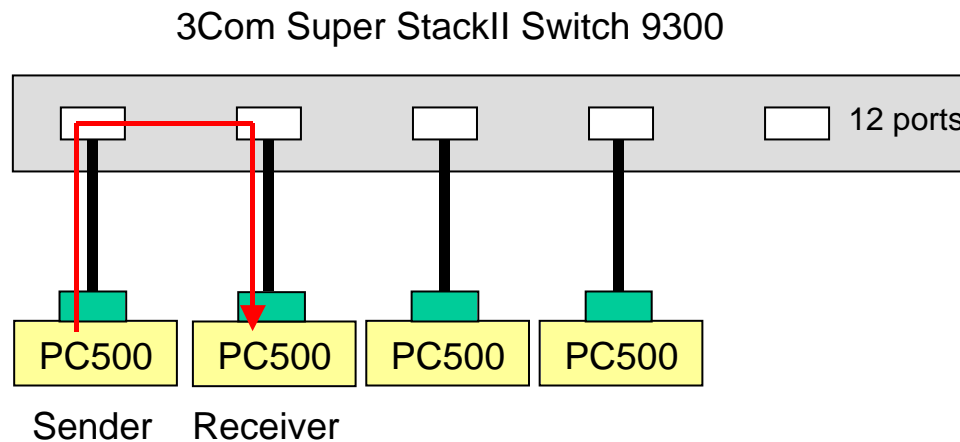
Transfer speed in Mbit/sec.



Message size in bytes

Peak speed was same.

# QoS Software Overhead: the CPU Usage

CPU Usage (%)



sender cpu usage(No QoS)

sender cpu usage(QoS)

In small size of message, QoS was 10 % worse. In large size of message, the percentage increases.

Message size in bytes

# TCP/IP Packet Distribution

3Com Super StackII Switch 9300



Sender    Receiver
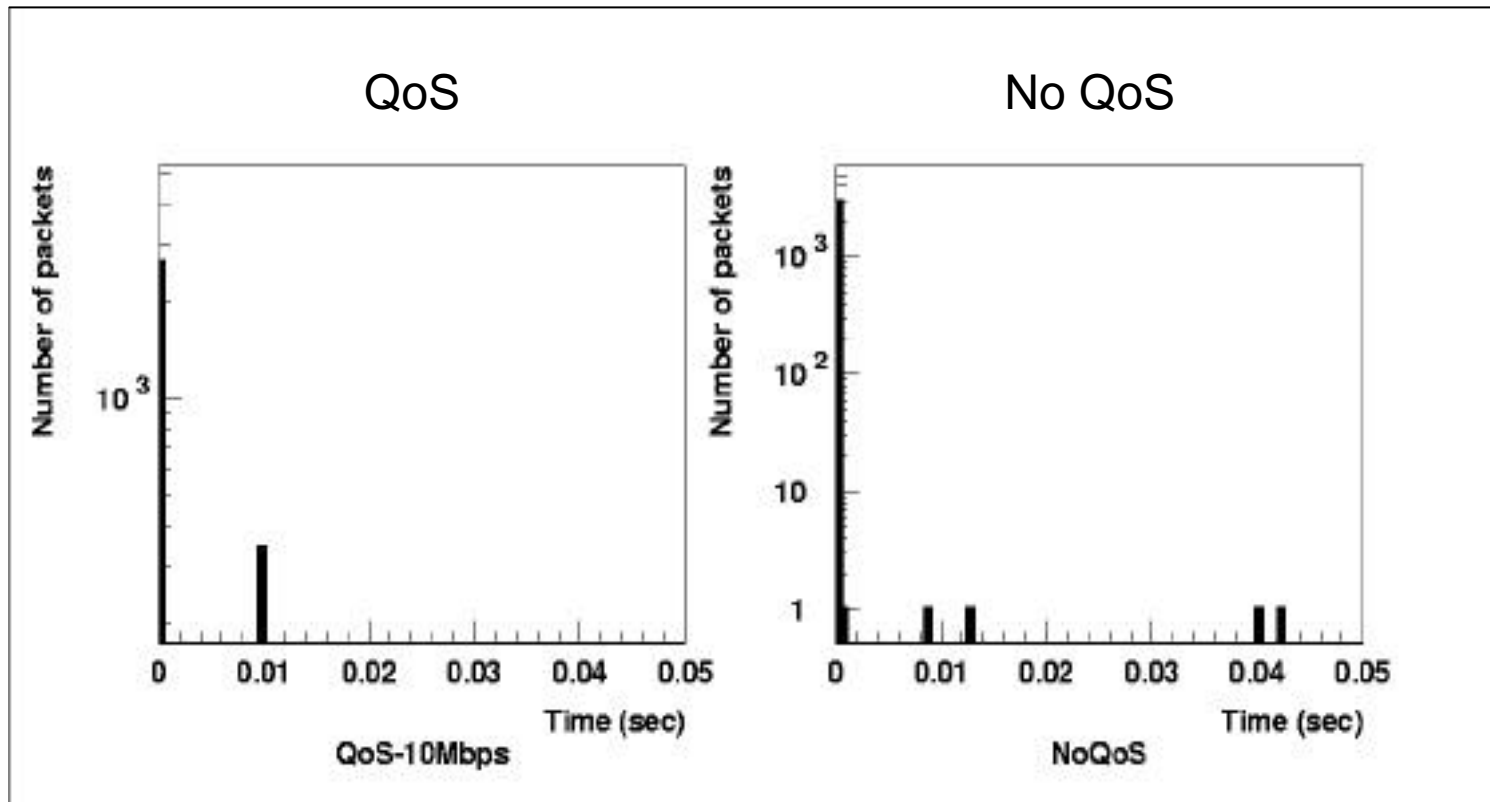
Assigned bandwidth is 10Mbit/s.

Sender sends message with 10 bytes to a receiver.

tcpdump utility is used for capturing packets

QoS

No QoS
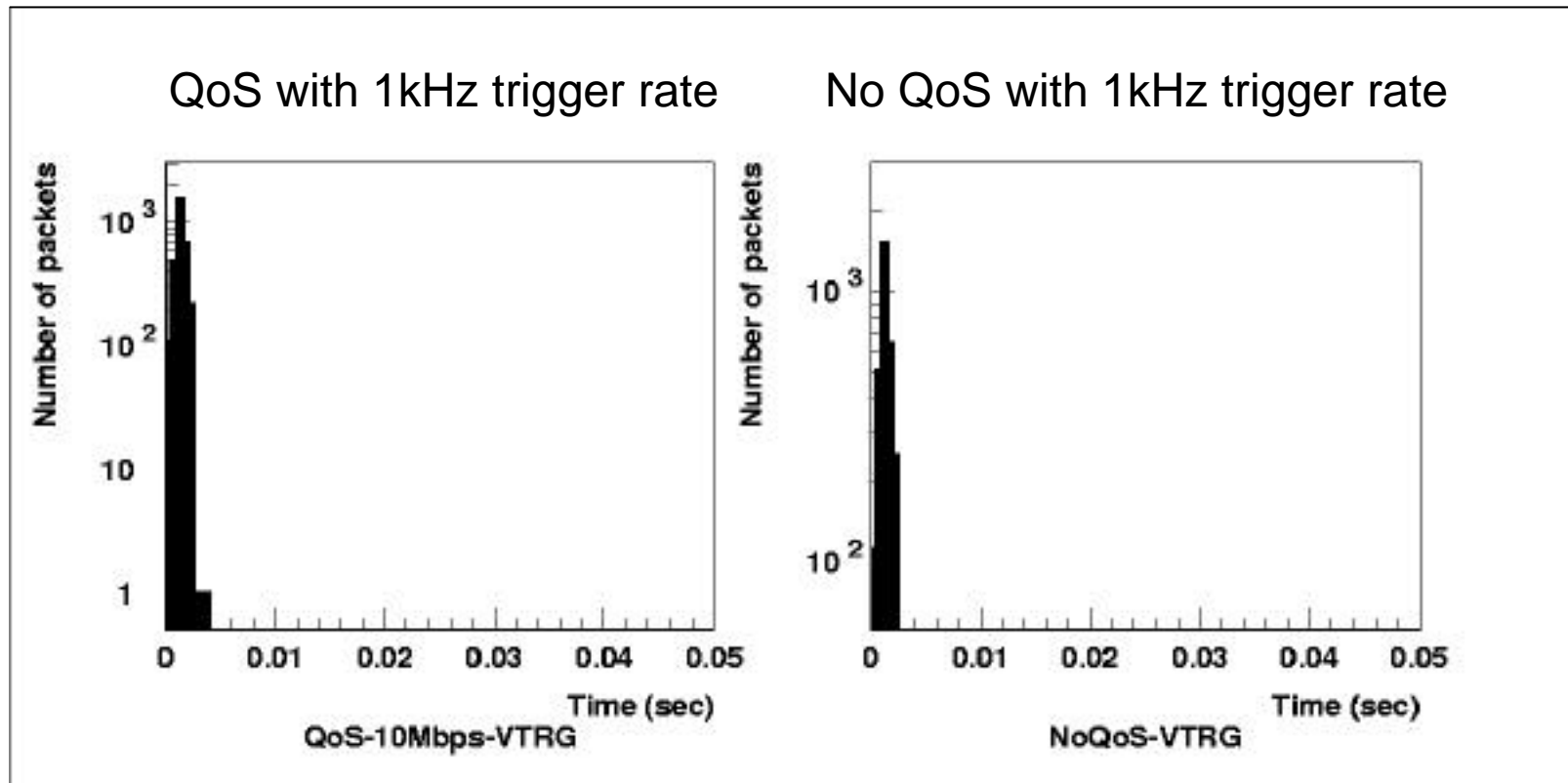


Horizontal axis : sending packet time interval

With QoS, some packets were sent at 10 msec scheduling time.
Without QoS, packets are mostly sent in several 10 usec.

# TCP/IP Packet Distribution(Cont.)

QoS with 1kHz trigger rate     No QoS with 1kHz trigger rate

With 1kHz trigger, packets are mostly sent around 1msec.
There are no difference between QoS and NoQoS.

# Conclusion

**We measured with/without QoS in Linux kernel:**
  **1) Packet loss on UDP/IP multicast transfer**
  **2) QoS software overhead on TCP/IP transfer**
  **3) Packet distribution on TCP/IP transfer**

**We concluded:**
  **1) QoS could eliminate packet loss on UDP/IP multicast transfer.**
     **This shows the feasibility to solve the scalability issue**
       **of ATLAS TDAQ Event Builder.**
  **2) QoS was efficiently performed on TCP/IP transfer.**
     **CPU usage of QoS on the transfer was small.**
  **3) With QoS, some packets were sent at 10 msec scheduling time.**