

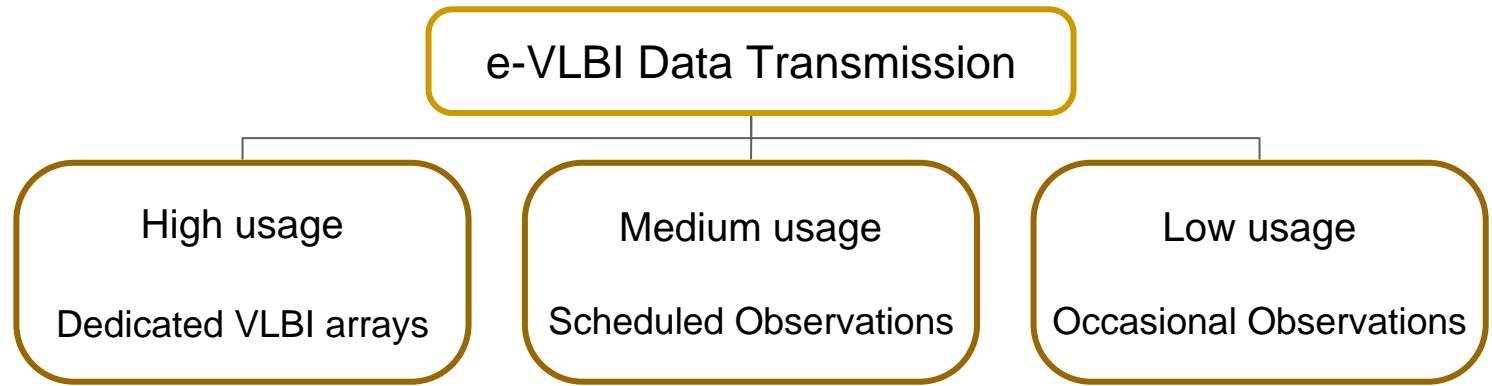
# VLBI Data Transmission System using Multiple IP Streams

---

Hisao Uose and Sotetsu Iwamura  
Service Integration Laboratories  
Nippon Telegraph and Telephone Corporation  
Tokyo, Japan

[uose.hisao@lab.ntt.co.jp](mailto:uose.hisao@lab.ntt.co.jp)

# Data Transmission Technologies for e-VLBI



Example: KSP, NMA?

GALAXY

Kashima-Haystack  
Observations

Suitable Technologies:

Dedicated Networks with dedicated BW

ATM, SONET or Lambda

- Reliable but rigid and expensive
- Direct mapping of data to reliable “pipes” with guaranteed bandwidth

Shared Networks

Internet

Multiple  
IP stream  
scheme

- Not reliable but flexible and economical
- Needs complicated processing (error correction, congestion control, etc.)

# IP Technologies for e-VLBI

## ■ Advantages

- Utilization of “off-the-shelf” equipment
  - PCs, routers, L2 switches, High-speed I/Os, etc...
- Existing resources (R&E networks with vast bandwidth)
  - Sharing resources with other BW eating applications
- Improved connectivity with other observation sites
  - With IP longer baseline becomes easier
  - Choice of antenna will be wider
- Compatibility to distributed processing schemes and flexible data retrieval / mining schemes
  - No data conversion required

## ■ Disadvantages

- QoS is hard to guarantee
- Complicated processing (error control, realignment of data packets etc.) necessary

# Idea: Reliable K4 cable over shared IP network

- Direct K4 replacement as the old ATM system does
  - Transparent ~256Mb/s data stream transmission
  - Can be upgraded to VSI compatible by changing the I/O units
- Inexpensive and scalable architecture
  - Use of inexpensive PCs (2 streams/PC)
  - Maximum 16 streams
  - Special hardware to interconnect K4 interface and PCs
  - Use of commercial interface standard (IEEE1394)
- Multiple operation modes
  - UDP mode (for networks with small congestion)
  - TCP batch mode (for congested network/short observation)
  - TCP stream mode (for congested network/longer observation)
- Ease of use
  - Centralized control application with GUI
  - Automatic delay adjustment (not implemented yet)

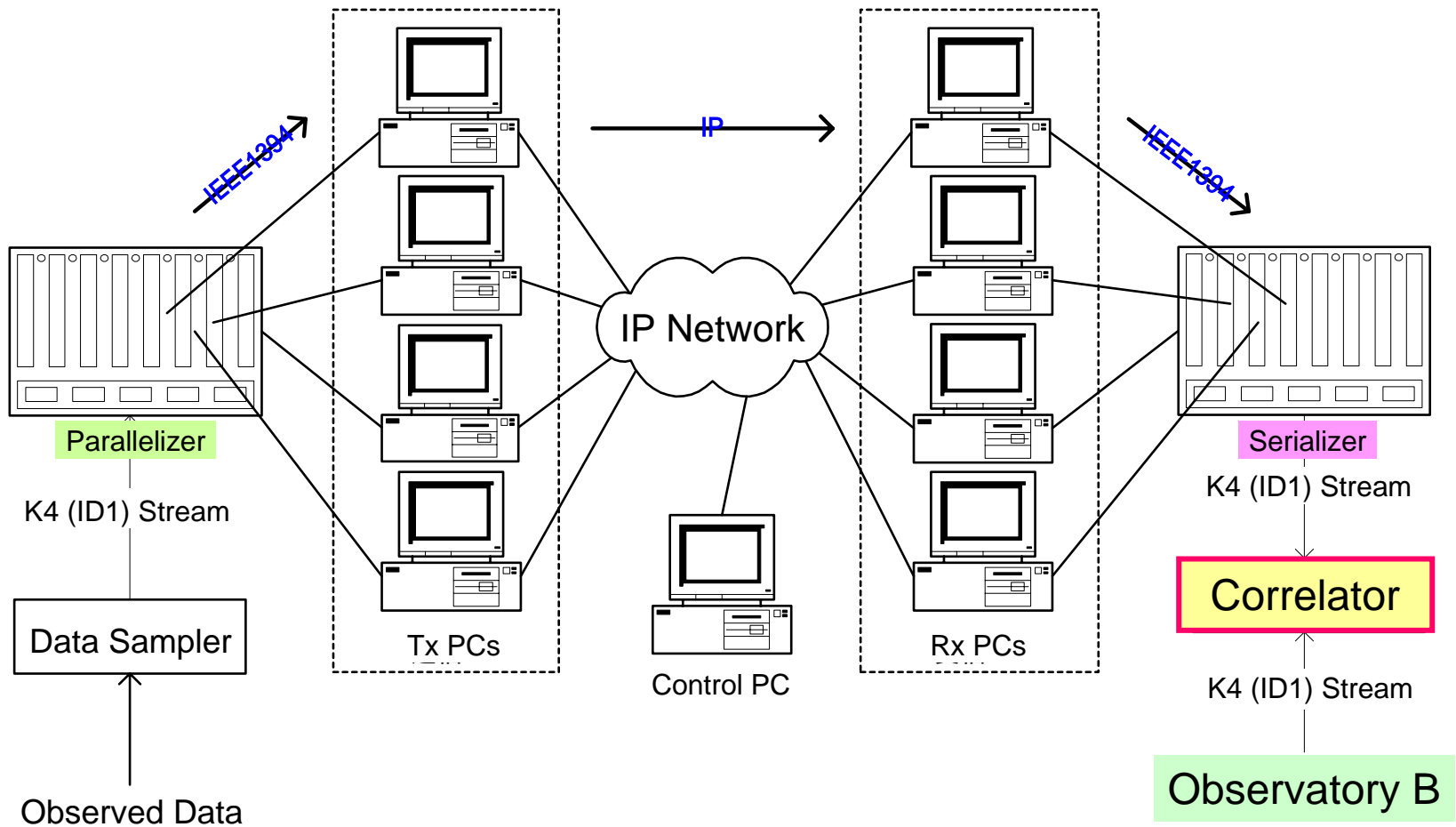
# Use of Multiple IP Streams

- Dealing with the latency problem
  - Accomplish high data rates with the use of multiple streams even under large latency environments
  - Suitable for international experiments
  - And have a nice share of the bandwidth at the same time..
- Dealing with the congestion problem
  - UDP mode assures time alignment with loss detection
  - TCP mode assures the reliability with built-in error control
  - Adjustable large buffer memories (disks) at both Tx/Rx sides
- Compatibility to software parallel correlation
  - Receiving PCs can store data in ordinary Unix file format
  - Can be used as a job dispatcher for distributed correlation
  - Why bother to send one very high-speed data stream if you have to chop up the stream into many fragments of data to be processed in parallel?

# Multiple Stream VLBI Data Transfer System

Observatory A

Correlator Station



# Transmitting and Receiving Data

- IP-transmitting PC: IEEE1394 --> IP
- IP-receiving PC: IP --> IEEE1394
- Software: coded in C language
- UDP mode
  - UDP/IP with sequence number inserted in payload
  - packet loss detection by counting sequence number
  - packet loss control: dummy (random) data insertion at receiving PC keeping the original time slots
- TCP mode
  - Error control with TCP/IP
  - Realignment of data stream at receiving PCs
  - Batch mode/stream mode
- Platform (for both IP-transmitting and receiving)
  - Pentium 2/400
  - Linux 2.4.6

# Serializer Unit



K4 Cable

Control Cable  
(RS232c)

IEEE1394  
Cable

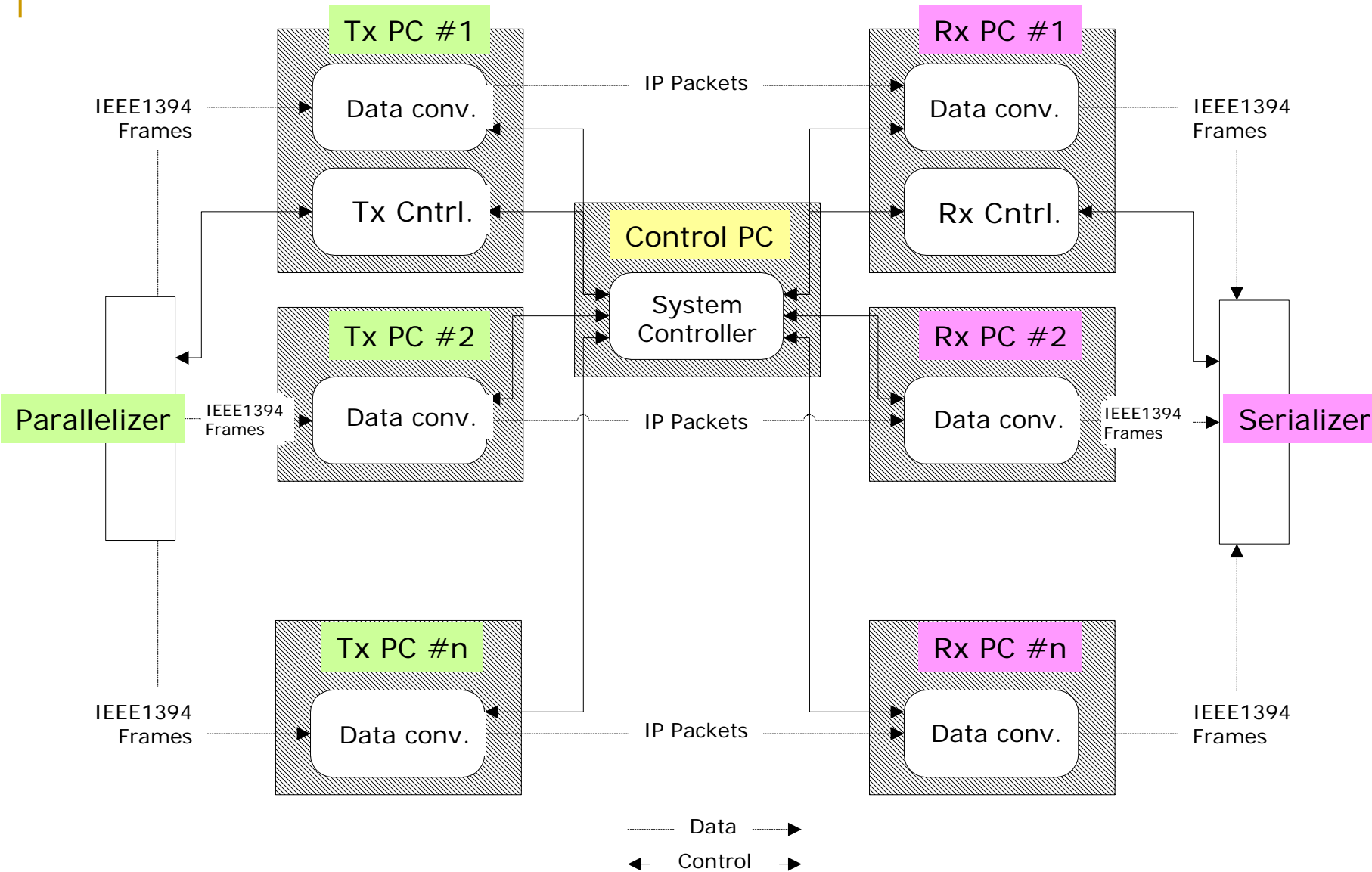
Display for  
operation mode



# Specifications of Parallelizer / Serializer

	ID1 parallelizer	ID1 serializer
Input I/F	ID1 x1	IEEE1394 x16
Output I/F	IEEE1394 x16	ID1 x1
Maximum PCs to connect	16	16
Internal data-block size (KB)	32, 64, 128, 256, 512, 1024 (manually selected)	
ID1 clock (MHz)	Automatic synchronization to input ID1 I/F	2, 4, 8, 16, 32 (manually selected)
Maximum throughput (Mbps)	256	256

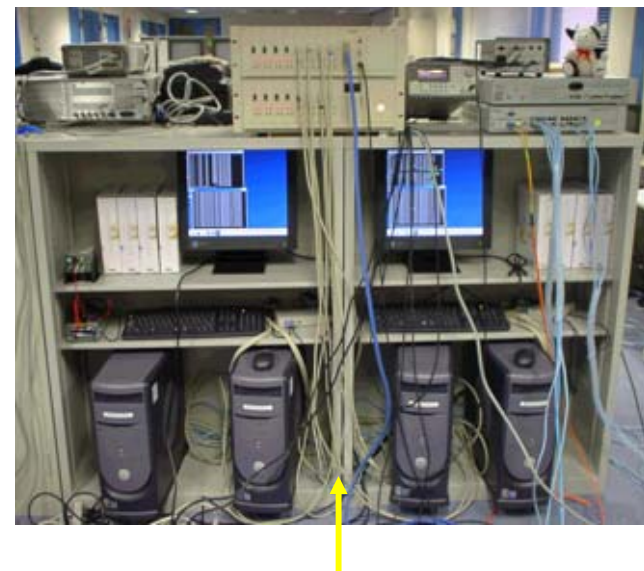
# Software Modules



# VLBI Data Transmission System



IP-transmitting  
PCs



IP-receiving PCs and ID1  
serializer



sampler and ID1  
parallelizer

# PC Setting Window

**PORT-PC LIST**

	Tx		Rx	
para equip	host name / IP	host name / IP	host name / IP	seri equip
PORT1	usuda1	musashino2		PORT1
PORT2	usuda2	musashino3		PORT2
PORT3	usuda3	musashino4		PORT3
PORT4	usuda4	musashino5		PORT4
PORT5	usuda1	musashino2		PORT5
PORT6	usuda2	musashino3		PORT6
PORT7	usuda3	musashino4		PORT7
PORT8				PORT8

**DATA STATUS**

**TxRx PC**

Trans Mode:  TCP-online  
 TCP-batch  
 UDP

Output Mode: (TCP-online / batch)  ISO trans  
 save DISK

Socket Buffer Size (MB):

Trigger Memory Size (MB): only TCP-online

Max Disk Size (MB):

IP Packet Size:  1KB  
 512B  
 256B

Com Time Out (sec):

Control Time Out (sec):

Clock Adjust (usec):

OK Cancel  
Quit

# Hardware Control Window



# System Control Window (TCP Batch Transfer Mode)

The screenshot shows a software interface titled "CONTROL SYSTEM" with several panels:

- TX PC Table:**

Parallel	PORT	IN	IP/HOST	Buffering Data Size	OUT
				0 100(MB)	
	port1	4	usuda1	4(MB)	0
	port2	4	usuda2	4(MB)	0
	port3	4	usuda3	4(MB)	0
	port4	3	usuda1	3(MB)	0
	port5	3	usuda2	3(MB)	0
	port6	3	usuda3	3(MB)	0
	port7	0			0
	port8	0			0
- RX PC Table:**

IN	IP/HOST	Buffering Data Size	OUT
		0 100(MB)	
0	musashino2	0(MB)	0
0	musashino3	0(MB)	0
0	musashino4	0(MB)	0
0	musashino2	0(MB)	0
0	musashino3	0(MB)	0
0	musashino4	0(MB)	0
0			0
0			0
- Serial PORT:** A column on the right side of the RX PC table listing ports from port1 to port8.
- SYSTEM:** Contains "START" and "STOP" buttons.
- EQUIP (parallel):** Contains a "STOP" button.
- SYSTEM STATUS:**

```

usuda2 : init SUCCESS
usuda3 : init SUCCESS
musashino2 : init SUCCESS
musashino3 : init SUCCESS
musashino4 : init SUCCESS
usuda1 : tx-rx connect SUCCESS
usuda2 : tx-rx connect SUCCESS
usuda3 : tx-rx connect SUCCESS
parallel equipment : start SUCCESS
*** Transmit Start ****
        
```
- PARAMETER INFO:**

```

Socket Buffer Size (B) :
    2048000
Trigger Buffer Size (MB) :
    Not define
Max Data Size (MB) :
    80
CLK :
    16M
        
```
- TIME INFO:**

```

start time :
    Fri Apr 25 19:09:12 2003

run time :
    0 h 0 m 4 s
        
```
- QUIT:** A button at the bottom right.

# System Control Window (TCP Stream Mode)

Parallel	TX PC			
	PORT	IN	IP/HOST	OUT
			0	62(MB)
port1	3	usuda1	0(MB)	3
port2	3	usuda2	0(MB)	3
port3	3	usuda3	0(MB)	3
port4	3	usuda1	0(MB)	3
port5	3	usuda2	0(MB)	3
port6	3	usuda3	0(MB)	3
port7	0			0
port8	0			0

RX PC				Serial
IN	IP/HOST	Buffering Data Size	OUT	
		0	62(MB)	
3	musashino2	3(MB)	0	port1
3	musashino3	3(MB)	0	port2
3	musashino4	3(MB)	0	port3
3	musashino2	3(MB)	0	port4
3	musashino3	3(MB)	0	port5
3	musashino4	3(MB)	0	port6
0			0	port7
0			0	port8

**SYSTEM**

START

STOP

**EQUIP (parallel)**

STOP

**SYSTEM STATUS**

usuda2 : init SUCCESS

usuda3 : init SUCCESS

musashino2 : init SUCCESS

musashino3 : init SUCCESS

musashino4 : init SUCCESS

usuda1 : tx-rx connect SUCCESS

usuda2 : tx-rx connect SUCCESS

usuda3 : tx-rx connect SUCCESS

parallel equipment : start SUCCESS

\*\*\* Transmit Start \*\*\*\*

**PARAMETER INFO**

Socket Buffer Size (B) : 2048000

Trigger Buffer Size (MB) : 50

Max Data Size (MB) : 130

CLK : 16M

**TIME INFO**

start time : Fri Apr 25 19:37:25 2003

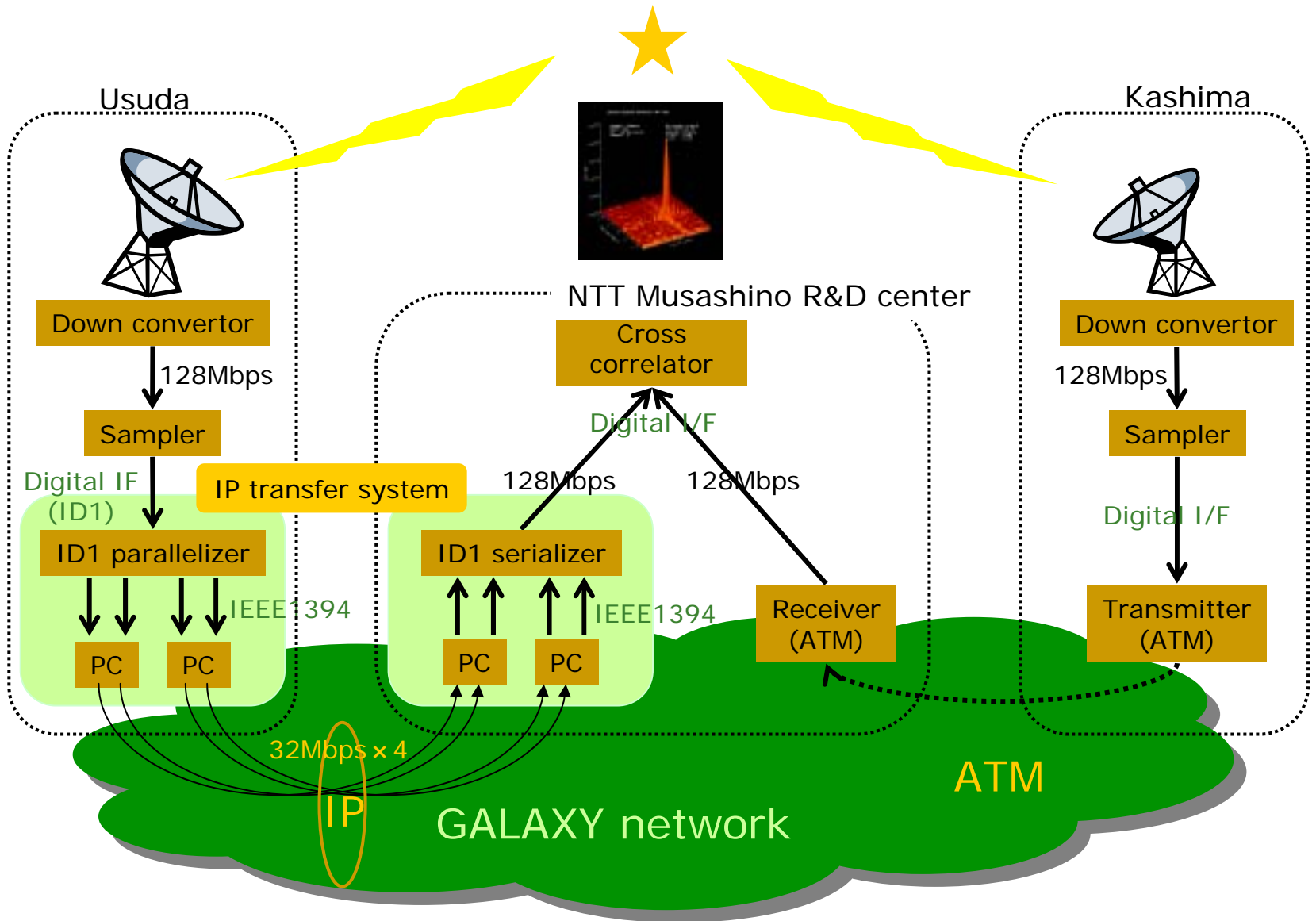
run time : 0 h 0 m 4 s

QUIT

May 16, 2003

Hisao Uose, NTT Laboratories

# Actual Observation using the system (Jan, 2002)





# Future Plan

- Test under the real environment
  - Performance
  - Reliability
- Application to real observation
  - Observations with universities connected with Super-SINET in Japan
  - International observations over GEMnet and Abilene
- Speed upgrade
  - Development of VSI interface
  - Use of newer PCs with GbE interface
- Delay management
  - Automatic delay adjustment among channels
- Integration with software correlation

## Acknowledgements

GALAXY members, CRL, Matsushita Electric Works, Ltd.

# TCP Port Assignments

