

第 XXVIII 部

非圧縮 HDTV 転送において
遅延制御装置を利用した
遠隔ジャズセッション

第 28 部 非圧縮 HDTV 転送において遅延制御装置を 利用した遠隔ジャズセッション

第 1 章 Introduction

In this report, we describe the overview of the experimental remote jazz jam session with uncompressed HDTV over the international LightPath at September 21st as a Grand Final event of the Aichi Exposition 2005. This is first International collaboration activities over LightPath operators for transmitting uncompressed HD streaming between Japan and Europe. In this report, we express the configuration of the LightPath and also describe the detail implementation of the Internet Metronome.

第 2 章 Lightpath Configuration

Recently the Internet research community is starting research on a next generation Internet architecture called as lambda networking. In the lambda networking, one or more lightpaths are provisioned between two locations, ideally the locations where the end systems reside. A light-path provides guaranteed bandwidth with fixed delay and very low packet loss rate. This property can hardly be implemented in a QoS enabled Internet. A number of national and international research networks including CANARIE, IEEAF, Abilene, SURFnet, and WIDE formed a group called GLIF (Global Lambda Integrated

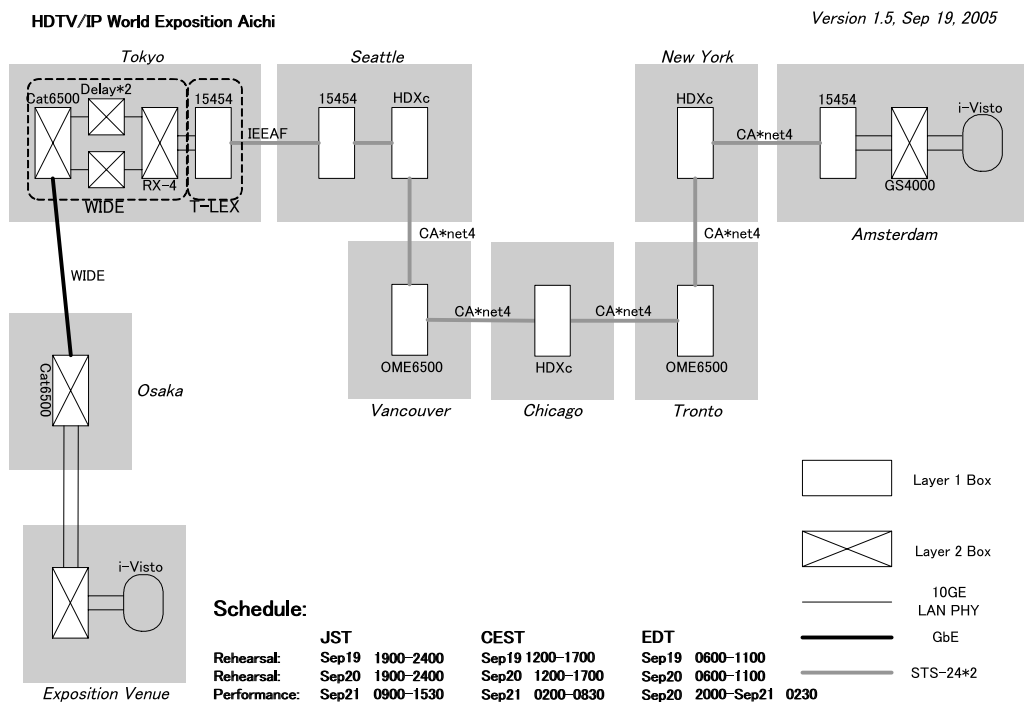


Fig. 2.1. World Exposition Aichi network diagram

第 28 部 非圧縮 HDTV 転送において遅延制御装置を利用した遠隔ジャズセッション

Facility)[197] in order to provide lightpaths which may spawn over multiple network infrastructures. Each lightpath may have large bandwidth such as GigabitEthernet (STS-24 equivalent), 2.4 Gbps SONET, or even 9.6 Gbps SONET or 10 Gigabit Ethernet WANPHY. A couple of STS-24 lightpaths were provided for the Internet Metronome demonstration.

2.1 Configured Lightpaths

A stream of HD-SDI consumes 1.485 Gbps. In order to transfer a bi-directional uncompressed HD stream between the venue and Amsterdam, a couple of Gigabit Ethernet links were provided, because the system we used was NTT’s i-Visto which used two Gigabit Ethernet ports for an uncompressed HD transfer. There were several fiber links to make the lightpaths between Japan and Netherlands. We prepared two gigabit ethernet over a STM-16 circuit between the venue and Osaka. Then packets were carried by the WIDE’s 10 Gbps backbone primary and JGNII 10 Gbps backbone for backup to Tokyo via two VLANs. In Tokyo, the VLANs on 10GE were mapped into two STS-24 lightpaths. The IEEAF Transpacific circuit provided the lightpaths to Seattle, WA and CA*net4 provided them their extensions to Amsterdam via Chicago, IL and New York, NY. All of the international links prepared through the international collaboration between among academic and research networks as a part of the GLIF activity.

Figure 2.1 illustrates the network diagram used for the demonstration.

第 3 章 Internet Metronome

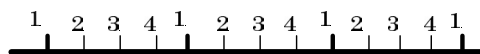
The beat is very important element of music. In remote music collaboration over the Internet, it is very hard to collaborate with beat, because, the communication delay can not be ignored in many

cases. In 1999, Ryuichi Sakamoto gave a performance called as “Internet Opera, LIFE” and we supported its communication path over the Internet. While we had been discussed about remote collaboration deeply before the performance, we failed to find a good solution at that time. In the “Internet Opera” performance, we made collaboration between the music that was played by Ryuichi Sakamoto at Tokyo and the dancers who were dancing with the music in New York and Frankfurt. The music was very slow without beat and the choreography of the dance was also flow without beat. Another remote music collaboration was that Seiji Ozawa gave a performance at opening ceremony of the Nagano Winter Olympic Games 1998 with Beethoven, symphony No. 9. “Chorus”. In this performance, the music of the orchestra conducted by Seiji Ozawa was transmitted to five cities in the world, and the chorus groups sang with this music and returned to the Nagano Olympic venue. Then mixing operation was performed with adjusting the timing of that music that played at each city. This performance was not collaboration with the orchestra and the chorus groups. It was just singing with the orchestra like a music minus one CD. To realize the music collaboration with remote musician, “Internet Metronome” was provided.

3.1 Concept of the “Internet Metronome”

The basic concept of the “Internet Metronome” is that making the beat based on the delay.

Figure 3.1 shows the basic beat with original metronome. Selecting 120 for the beat, the metronome ticks each 500 ms interval. The original metronome is placed at one side of distant locations and the ticks of the beat are transmitted



1/120: 1 beat is 500ms interval

Fig. 3.1. Beat with Original Metronome

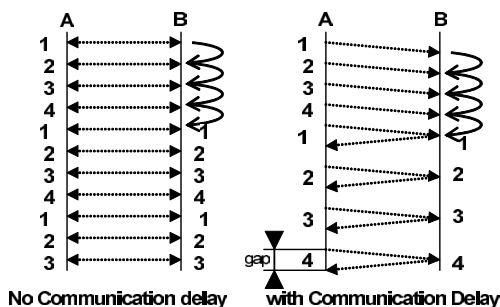


Fig. 3.2. Gap on the Communication delay

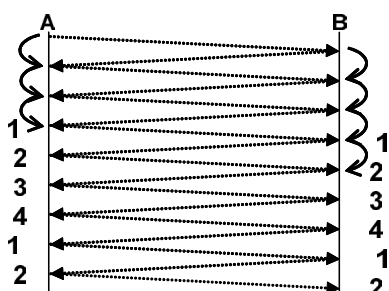


Fig. 3.3. Beat on the Internet Metronome

as shown in Figure 3.2. If there is no transmission delay, ticks make a beat at both sides. But with some transmission delay, ticks cannot make a beat at both sides. Suppose that a musician listens only the other side of music and play own music. (Like a normal remote conference setting just considers minus one.) A musician starts the play at side A. Then another musician plays at side B with listen the music transmitted from side A. The musician at side B can collaborate with the music played by side A. But the musician at side A cannot collaborate with the music played by side B. Because the musician at side A listens the music with gap of beat.

The “Internet Metronome” makes a beat based on transmission delay such as Figure 3.3. Using the “Internet Metronome”, the players can share the beat both side.

3.2 Implementation of the Internet Metronome

Figure 3.4 shows the overview of Internet Metronome. If a player in Amsterdam started performance at the same time of the players in the Aichi venue, the stream arrives at the venue with some delay.

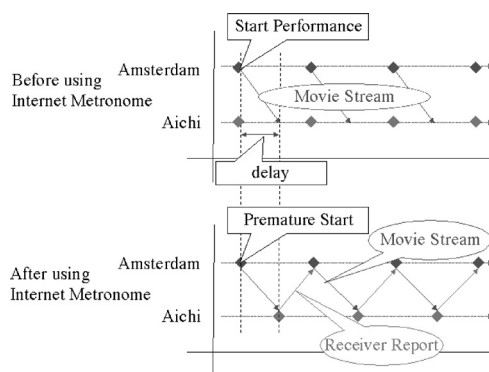


Fig. 3.4. System overview

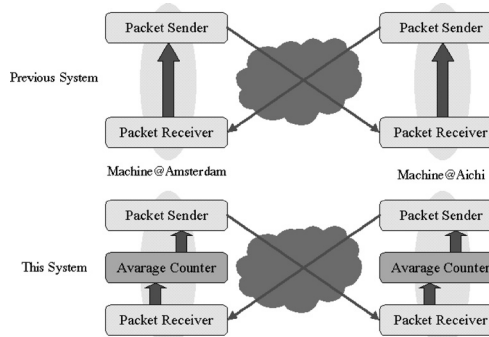


Fig. 3.5. Average Counter

A simple ICMP-like packet is used to measure delay time between Amsterdam and Japan. There are 3 steps of measuring the delay.

1. Send a packet from Amsterdam to Aichi and ring bell.
2. Upon reception of the packet in Aichi, send a response packet back to Amsterdam.
3. Upon reception of the packet in Amsterdam, calculate time since the last packet was sent. Calculated time is delay time between Amsterdam and Japan.

Consideration of packet-loss and jitter caused by a packet-loss is required for Internet Metronome. To solve this problem, we implemented a structure called “average counter”. Figure 3.5 shows the average counter. Previous counter is based on a packet retrieval base system, not effective when packet loss occurs. To solve packet loss problem, we implemented first version of “average counter”. This version calculates average time from previous sending time stamp and receiving time stamp. The calculating formula for average

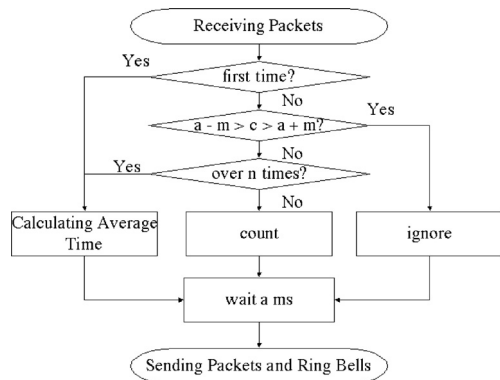


Fig. 3.6. Calculation method of average delay time

a (ms) is shown below. To calculate average time, current delay time c and previous average p are only used.

$$a \text{ (ms)} = \frac{c + p}{2}$$

We focused that network delay jitter can be divided into two types, accidental delay and consecutive delay. Figure 3.6 shows the calculation method of average time.

“Internet Metronome” is implemented on Windows XP Platform. In the Actual event, Fujitsu LOOX notebook PC was used for the metronome.

第 4 章 Delay-control

To evaluate the effectiveness of “Internet Metronome”, 2 types of delay should be considered: 1) Network delay, 2) Equipment codec latency. 1) as described in former section, a delay constructed by the continental location differences, from Netherlands to Japan. 2) is a delay caused by a codec’s of HD digital video streams. For applied evaluation, we constructed a delay-control unit within the network to artificially increase the delay of the network and prove the effectiveness of “Internet Metronome”. Network bandwidth consumed by i-Visto[187] is more than

1.5 Gbps per an HD stream. In this remote jam session event, a bi-directional streams is required for collaboration, consuming total of more than 3 Gbps bandwidth.

There are 3 ways to generate packet delays.

1. Using a bulk optical fiber to establish a physical delay.
2. Using a network memory delay box.
3. Using a network packet forwarding control application.

There are several fixed delay box with fiber connectors capable of generating precise delay using optical bulk fibers. However, it is hard to configure variable set of delays using the combination of boxes, together with the attenuation of laser signals caused by the distance of fiber itself. Solution using network memory boxes will be applicable to generate variety of delay timings, but assuming the bandwidth consumed by HD, network memory box will require a big memory buffer to allocate delay packets. In our events, we constructed a PC based delay-control unit using network packet forwarding control application. We enhanced DUMMYNET[227] which can be implemented in most of UNIX based operating systems. Using DUMMYNET will effectively generate delay within the packet streams.

DUMMYNET has been used in many applications, and researches for evaluation of bandwidth control. Somehow, most of the applications are using DUMMYNET as a packet flow control and bandwidth limitation. Generating a delay for a Gbps (Gigabit per seconds) class network bandwidth is not considered in current DUMMYNET design.

To establish a delay for a Gbps class traffic using DUMMYNET, optimization of hardware coupled with operating system is required. The operating system should collaborate with hardware for optimal specifications. The main goals for the design are: efficient code/minimum overhead.

We designed the delay-control unit with two types of network interfaces: 10 Gigabit Etherne and Gigabit Ethernet. 2 delay-control units are

required for Gigabit Ethernet to forward an HD stream.

In the traditional DUMMYNET implementation, due to network buffer availability as well as kernel memory map space limitation, construction of delay is limited. Our initial implementation with DUMMYNET results not more than 300 Mbps of bandwidth capability for 500 ms.

4.1 FreeBSD Kernel Optimization

Using DUMMYNET as a delay-control unit, kernel optimization is required. One of the most effective solutions to enhance the bandwidth and delay capability of DUMMYNET is to allocate as many memories as possible for delay buffers. Some of the kernel state can be configured through sysctl interface. The sysctl utility retrieves kernel variables and allows a process with appropriate privilege to set them.

kern.ipc.nmbclusters has been increased to enlarge the number of network mbufs in the system allocations. Each network cluster represents approximately 2 Kbyte of memory.

We increased *nmbclusters* to 819200, allocating 1.6 Gbyte of memory space to network memory buffers. Configuration of *vm.kmem_size* is also required for kernel memory space allocation.

Using 10 Gbit Ethernet effectively, modification of device driver was required. Intel PRO/10GbE LR Server Adapter has configurable parameters for optimal network throughput under conditions. Parameter *RxIntDelay* generates receive interrupts in units of 0.8192 microseconds. Receive interrupt reduction improves processor efficiency under burst UDP traffic streams. The default configuration of Hz in FreeBSD is 100, which is larger than single *RxIntDelay*. We tuned this parameter to 123 generating receive interrupt approximately each 100 ms.

We also considered device polling configuration for kernel interface. In conditions, which is called interrupt-based mode, network interface generates an interrupt whenever packet is received. Under heavy traffic load of an HD stream, the system

might persistently process the interrupts without being able to completing packet that needs to be sent. When using DUMMYNET in HD delay-control, configuring device based interrupt with interrupt delay was most effective.

Optimization of PCI card timing specification can results in expansion of bandwidth limitations. Configuration of MMRBC (Maximum memory Read Byte Count) field from 512 byte to 4 Kbyte in PCI-X configuration space increases transmit burst data length on the bus.

Since we used DUMMYNET in bridged mode, we had a difficulty in configuring DUMMYNET through tagged VLAN based network infrastructure. Whenever the delay configuration change occurs, tagged VLAN packet leak occurs within the bridge causing router to temporary shutting down the network interface. To solve the situation, in the actual event, we configured two delay-control unit for each HD network constructed by Gigabit Ethernet.

4.2 Implementation

2 types of delay-control box were implemented.

1. 10 Gigabit Ethernet Delay-control Box

Processor: AMD Opteron Model 270
(2 GHz) × 2

Motherboard: TYAN Thunder K8W

Memory: DDR400 4 GB

Chipset: AMD 8131 Chipset

Network Interface: Intel Pro/10GbE LR Server Adapter × 2, Broadcom Gigabit Ethernet

2. Gigabit Ethernet Delay-control Box

Processor: Intel Xeon 2.0 GHz L2 2 MB
800 MHz FSB

Motherboard: Dell Power Edge 1850

Memory: DDR400 3 GB

Chipset: Intel 7520 Chipset

Network Interface: Intel PRO/1000 Network Connection SC × 2, Intel 82541 Dual port Gigabit Ethernet

4.3 Packet Throughput

Table 4.1 show the status of packet throughput in delay-control box sending bi-directional HD stream. Delay-control box shows no packet losses during the event, with constant processor load average of under 0.3. In this status, we probed the bandwidth using systat. The period of probing of status was taken for 600 seconds. The size of input and output differ because of the packet dump occurred during the phase of changing the delay period.

Table 4.1. Packet Throughput of the Delay-control Unit

Interface	Traffic (Mbps)	Peak (Mbps)
Interface 1 in	191.290	191.611
Interface 1 out	191.260	191.931
Interface 2 in	191.123	191.886
Interface 2 out	191.203	191.668

第 5 章 Music Collaboration

We selected the jazz jam session for proving the Internet metronome from various kinds of music collaboration. Jazz jam session has beat and chord. But it is not so tight rather than classical music. As the timing of the translating the chord around the players is not so tight in jazz jam session, is suitable for the remote music collaboration with the Internet metronome.

In this experimental jazz jam session, four jazz musicians were staffed such as Piano/Vocal, Tenor Sax/clarinet, Violin and Drums/Percussion. The violin player played in Amsterdam and other musicians played in the Aichi Exposition venue. This staffing and locations of the players were defined through the previous excises. We had excises at local studio that has two soundproof rooms connected by DVTS/i-Visto streaming technologies twice. We can support any

combination of the players. The lead piano and rhythm section were located at same place and made main music stream. Then violin over came onto that music. This configuration was easy way to collaboration.

We demonstrated three types of the collaboration by controlling the speed of the Internet metronome on the stage as following.

1. Classical ensemble with one beat gap
2. Kanon with one subsection gap
3. Professional Jazz Collaboration (free jam session)

Classical ensemble with one beat gap was very hard to collaborate by remote players, because remote player (as violin player in this experimental session) should play the melody with one beat before.

Kanon is music that has originally one subsection gap. This matched the environment well. We controlled the Internet metronome as one subsection. Remote player just played with other side player, caused to kanon to the audience.

And finally professional jazz players played free jam session. They had several experiences to play with a delay environment and the Internet metronome. They were able to play and make good collaborated jazz jam session with one beat gap.

第 6 章 Conclusion

This experimental jazz jam session on the Aichi Exposition 2005 was the first music collaboration with long haul Internet with lightpath technology. Internet Metronome with delay-control units could provide the new music collaboration environment over the Internet with long delay. Previous music collaboration's trials tried to identify the allowance delay over the Internet. Our strategy was different from those previous music collaborations. We explicitly consider the delay over

the Internet, and did not ignore the delay. Internet Metronome could make a tempo using the delay. We could proof that the musicians were able to play with this new music collaboration environment. It was not normal music collaboration environment, but we believe that musician at different locations will be enjoy the collaboration over the Internet with Internet Metronome technology.

In this experiment, we deal with two new high-end technologies over the Internet. One was an uncompressed HD transmission system and the other one was lightpath technology. Using those technologies, we could proof that the availability and operation-ability of the lightpath technology, especially to provide provisioning services such as constant delay and guaranteed bandwidth for high data volume streaming services. Uncompressed HD streaming used 1.5 Gbps bandwidth. And i-Visto was very sensitive to the jitter. They did not enough buffers for absorbing the jitter. We could success this experiment by using lightpath technology.

Acknowledgements

We sincerely thank the people who have greatly contributed to this particular project from IEEAF, Pacific Northwest Gigapop, CANARIE, SURFnet and SARA, especially Dr. Egon Verharen who coordinated Amsterdam side for this project even David Verharen (his son)'s delivery expected at the same time, then he was born just before the performance. Also, we thank to the Japan Association for the 2005 World Exposition and National Institute of Information and Communications Technology who gave us the opportunity to have this demonstration as grand final event.

