

## Multipoint Streaming Technology for 4K Super-high-definition Motion Pictures

*Hirokazu Takahashi<sup>†</sup>, Daisuke Shirai, Takahiro Murooka, and Tatsuya Fujii*

### Abstract

This article introduces wide-area multipoint 4K-motion-picture streaming technology and reports the results of an experiment conducted at the 19th Tokyo International Film Festival.

### 1. Introduction

4K super-high-definition motion pictures (4K motion pictures for short) is a set of specifications for next-generation digital motion pictures that has recently attracted a lot of attention because of its high resolution. It has 2160 lines, each with 4096 pixels, which offers approximately four times the resolution of the most widely used high-definition television format (1080 × 1920 pixels). The 4K concept was first proposed by NTT Corporation [1]. In July 2005, it was applied to digital cinema specifications proposed by DCI (Digital Cinema Initiatives, LLC), which was organized by major studios in Hollywood. A cinema trial called “4K Pure Cinema” [2] is currently being conducted in movie theaters in Japan by NTT Corporation, NTT East Corporation, NTT West Corporation, NTT Communications Corporation, and some studios, distributors, and exhibitors.

High-resolution motion-picture applications other than cinema are also becoming a popular topic in video streaming communities. Such applications are called ODS (online digital source). One example of ODS is the highly realistic live streaming of music concerts or sports events using the large screens of movie theaters. To implement such applications, we are now studying wide-area multipoint 4K-motion-picture streaming. The way the streaming works is outlined in **Fig. 1**. A 4K motion picture taken by a

camera or retrieved from a storage device is streamed to several receiving clients through an IP (Internet protocol) network. Each receiving client displays the received motion picture using a 4K-motion-picture-capable projector that can project the images onto a screen or using a large-screen display such as a liquid crystal display (LCD).

In this article, we introduce a mechanism for implementing the streaming and discuss its problems and our solutions.

### 2. Problems of wide-area multipoint 4K streaming

A 4K-motion-picture stream, or 4K stream for short, occupies a bandwidth of around 400 Mbit/s. Multipoint streaming of such a large bandwidth is still difficult using a server based on a personal computer (PC) because of the limits of computation performance and the network access line bandwidth. The in-network stream replication mechanism called Flexcast [3] enables the streaming server to send only one stream to a stream-replicating node called a Flexcast splitter. One or more Flexcast splitters can be located in a network. A Flexcast splitter generates a number of replicated streams and sends them to corresponding Flexcast splitters or receiving clients. In this way, the stream replication load can be dispersed to several Flexcast splitters, so the server is not burdened with that job. In addition, Flexcast uses an IP unicast packet to transfer data through a network and can be operated on networks that are not capable of IP multicasting. In contrast, the conventional Flexcast

<sup>†</sup> NTT Network Innovation Laboratories  
Yokosuka-shi, 239-0847 Japan  
Contact: [http://www.ntt.co.jp/sclab/contact\\_e.html](http://www.ntt.co.jp/sclab/contact_e.html)

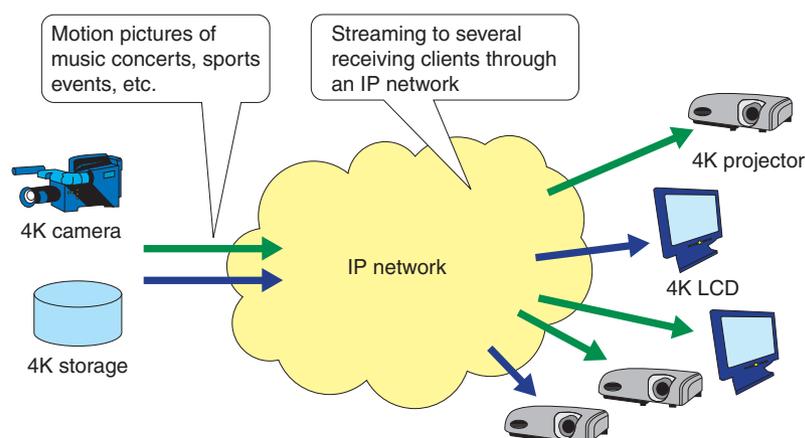


Fig. 1. Outline of wide-area multipoint 4K-motion-picture streaming.

splitter is implemented as software running on a conventional PC-based system and cannot support the bandwidth of a 4K stream.

Although 4K motion pictures provide high-quality video, the conventional IP network transfers data in a best-effort manner and does not support any mechanisms for recovering lost data, so accidental packet loss will be inevitable in a wide-area network. For the picture quality we desire, complete reception of 4K-motion-picture data is absolutely necessary. Therefore, we need a recovery mechanism that can cope with the bandwidth of a 4K stream. Some ODS applications, such as distance education, cannot tolerate a large delay for bidirectional communication. Therefore, delay is also an important aspect of the mechanism. From the above discussion, we can identify two key requirements.

#### (1) High-performance stream replication

Stable stream replication is essential for 4K-motion-picture multipoint streaming. Therefore, a high-performance stream replication mechanism must be developed.

#### (2) Lost data recovery mechanism

A lost data recovery mechanism is required to guarantee picture quality. It should support the bandwidth of the 4K stream with only a small processing delay.

### 3. Solutions

The framework of a wide-area multipoint 4K-motion-picture streaming system, which is our goal, is shown in **Fig. 2**. The J2K codec [4] is a codec for 4K motion pictures. It encodes 4K motion pictures using JPEG2000 and generates a 4K stream that occupies a bandwidth of around 400 Mbit/s in real

time. The 4K stream is replicated by Flexcast splitters and forwarded to several receiving clients. Each receiving client has a J2K codec that decodes the received 4K stream in real time. Two key technologies enable us to meet the requirements and implement the system.

#### (1) Hardware implementation of the replication process

We implemented the replication process of the Flexcast splitter as hardware to handle the large bandwidth of the 4K stream.

The Flexcast splitter consists mainly of a configuration process and a replication process. The configuration process assigns destinations to replicated streams according to requests from clients and registers the destinations in a memory area called a destination table. The replication process replicates a given stream and sends the replicated streams to destinations listed in the destination table. The configuration process is invoked by the arrival interval of the client requests, which is ten seconds per client. The replication process must handle around 30,000 incoming packets per second to replicate a 4K stream. Based on the characteristics of these processes, we implemented the configuration process as software and the replication process as hardware.

The Flexcast splitter is located in an IP network and connected to conventional IP routers, which have network interfaces that can handle data at up to 10 Gbit/s. Therefore, to make maximum use of the 10-Gbit/s bandwidth, we designed the replication process to be able to generate replicated streams with a total bandwidth of 10 Gbit/s. To achieve the design goal, we used pipeline-based parallel processing. This means that the sub-processes of the replication

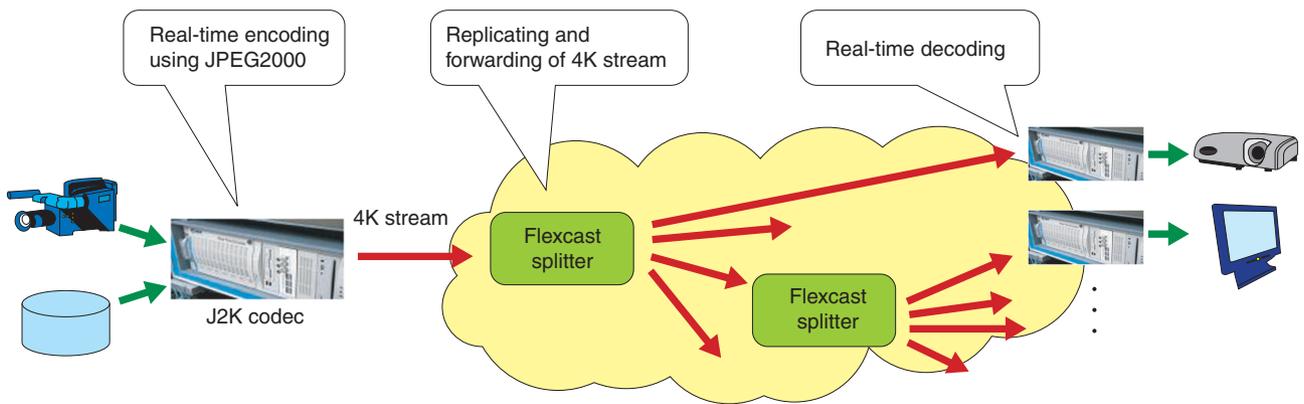


Fig. 2. Framework of a wide-area multipoint 4K-motion-picture streaming system.

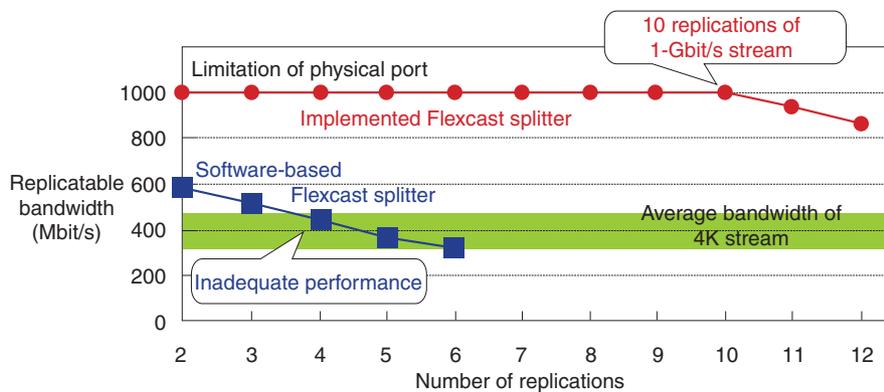


Fig. 3. Replication performance of Flexcast splitter.

process run simultaneously and can handle large-bandwidth streams at a lower clock rate. Therefore, the 4K-stream processing can be implemented on a field programmable gate array [5] that cannot support clock rates as high as those in an ASIC LSI (application specific integrated circuit large-scale integration).

Performance results for the Flexcast splitter are shown in Fig. 3. The figure shows the incoming stream bandwidth that the Flexcast splitter could replicate without discarding any packets. The conventional software-based Flexcast splitter running on a dual Xeon 3.6-GHz PC could not generate more than four replicated 4K streams. The Flexcast splitter that we implemented, on the other hand, could generate ten replicated 1-Gbit/s streams. Thus, the replication performance of the Flexcast splitter could be improved by implementing it as hardware and the design goal was achieved.

(2) Error correction code

We used an error correction code called LDGM (low-density generator matrix) code to recover lost packets.

The principle of a packet loss recovery mechanism that uses an error correction code is shown in Fig. 4. First, source packets that include data to be sent are encoded. The encoding process is done for every  $k$  packets. Thus,  $k$  is called the message size. Several parity packets are generated by the encoding and are used for packet loss recovery. Block size  $n$  is the sum of  $k$  and the number of parity packets, and the ratio of the number of parity packets to the message size,  $(n-k)/k$ , is called the redundancy. Here,  $k$  and  $n$  are coding parameters and can be configured by the operator. If packets are lost in the network, the decoding process attempts to recover the lost packets by using the parity packets.

Generally, the number of lost packets that can be recovered increases as  $n$  increases. However, using a

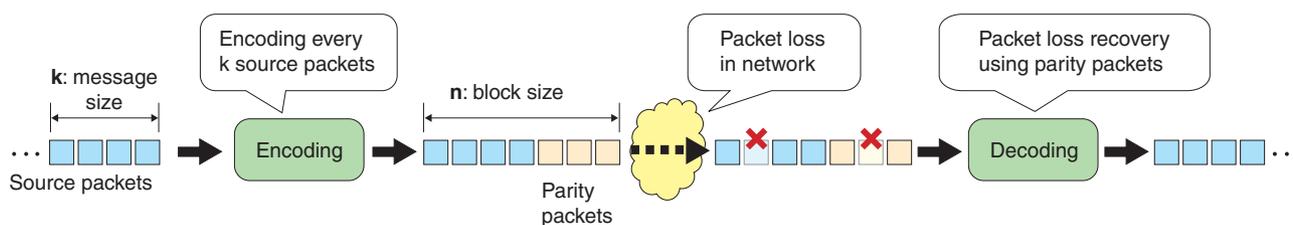


Fig. 4. Packet loss recovery using an error correction code.

Table 1. Recovery performance of error correction codes.

	LDGM	Horizontal parity	Reed-Solomon
(n, k)	(2928, 2700)	(13, 12)	(204, 188)
Redundancy	8.4%	8.3%	8.5%
Number of lost picture frames	0 in 11,560 (no loss)	11,540 in 11,560	212 in 11,560

larger  $n$  means that the coding process cannot be started until after a large number of source packets have been collected, which increases the processing delay. On the other hand, our target is to handle a large-bandwidth stream, so a large number of source packets are generated in a short period. This can suppress the increase in processing delay. Therefore, in this case, an error correction code that can handle a large  $n$  should be chosen. Unfortunately, because the amount of coding calculation increases exponentially as  $n$  increases, the widely used Reed-Solomon code has a practical limitation of  $n \leq 255$ .

We needed an error correction code that can handle a larger  $n$  and simultaneously minimize the increase in the amount of calculation required. Therefore, we chose an LDGM code that can handle a larger  $n$ , on the order of 1000 or 10,000, with a practical amount of calculation. We implemented it as software for the J2K codec. We also investigated and applied a coding parameter that can achieve stable recovery performance regardless of packet loss conditions.

Performance results for the LDGM, commonly used horizontal parity, and Reed-Solomon codes are shown in **Table 1**. In this evaluation, a 4K stream consisting of 11,560 picture frames was transported through a network with an average packet loss of 4%. The LDGM code was able to handle a larger  $n$  (2928) than the other codes (13 and 204, respectively). This  $n$  was determined by the delay and the amount of calculation. The results show that picture frame loss occurred when the horizontal parity and Reed-Solomon codes were used. The LDGM code, on the other hand, could recover all the lost packets, achiev-

ing higher recovery performance.

#### 4. Experiment at the 19th Tokyo International Film Festival

We conducted a public experiment at the 19th Tokyo International Film Festival to test the performance of the technologies described above and promote ODS. The experimental setup is shown in **Fig. 5**. The experiment involved five locations: Roppongi (festival site), Mita (DMC, Keio University), Akihabara (NICT: National Institute of Information and Communications Technology), Keihanna (NICT), and Yokosuka (NTT). Some photographs of these locations are shown in **Fig. 6**. They were connected by broadband access provided by NTT Communications, JGN II (Japan Gigabit Network II), and GEMnet 2 (global enhanced multifunctional network 2). In the experiment, 4K streams of a lecture, a music concert, and a video game generated by a J2K codec in Mita were replicated by a Flexcast splitter in Akihabara. The streams were received by a total of ten receiving clients, which consisted of J2K codecs and dummy receiving clients, and displayed at the locations. During the experiment there was no picture frame loss or degradation of picture quality, and high-quality 4K images were successfully streamed to the ten receiving clients.

#### 5. Future work

In the future, we plan to investigate and implement security functions such as client authentication and

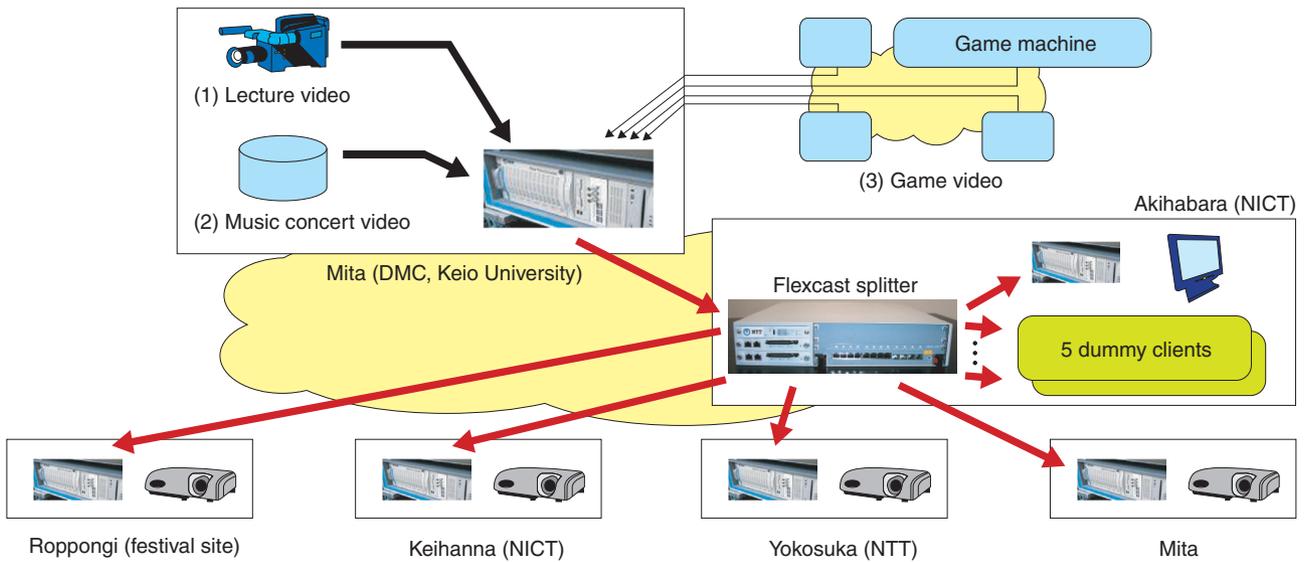


Fig. 5. Experimental setup at 19th Tokyo International Film Festival.

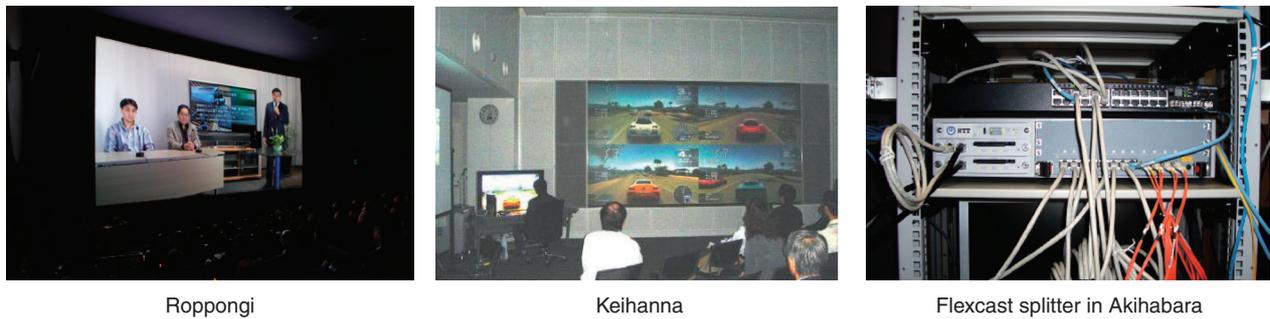


Fig. 6. Photographs of experimental locations.  
 RIDGE RACER™7 ©2006 NAMCO BANDAI Games Inc.

content encryption to prevent theft of the streams in the network.

### References

[1] N. Ohta and S. Ono, “Digital Signal Processing for Super High Definition Image Transmission,” Proc., Telecom’91, 1991.  
 [2] H. Sakamoto, K. Minami, K. Shirakawa, T. Fujii, Y. Saito, and H. Yamane, “The “4K Pure Cinema” Joint Digital Cinema Trial,” NTT Technical Review, Vol. 4, No. 7, pp. 18–22, 2006.  
 [3] T. Inoue, S. Tani, H. Takahashi, S. Minato, T. Miyazaki, and K. Toyoshima, “Design and Implementation of Advanced Multicast Router Based on Cluster Computing,” Proc., ICPADS2005, 2005.

[4] D. Shirai, T. Yamaguchi, T. Shimizu, T. Murooka, and T. Fujii, “4K SHD Real-time Video Streaming System with JPEG 2000 Parallel Codec,” Proc., APCCAS2006, 2006.  
 [5] S. D. Brown, R. J. Francis, J. Rose, and Z. G. Vranesic, “Field-Programmable Gate Arrays,” Kluwer Academic Publishers, 1992.

This research is part of the project for “Research and development of next generation video content distribution and production technologies” supported by the Ministry of Internal Affairs and Communications.

**Hirokazu Takahashi**

Media Innovation Laboratory, NTT Network Innovation Laboratories.

He received the B.E. and M.E. degrees in electrical engineering from Nagaoka University of Technology, Niigata, in 2000 and 2002, respectively. Since joining NTT Network Innovation Laboratories in 2002, he has been researching multicasting technologies for realtime applications. His current interests include reliable multicasting techniques for large-bandwidth streams.

**Takahiro Murooka**

Senior Research Engineer, Media Innovation Laboratory, NTT Network Innovation Laboratories.

He received the B.S. degree in physics from Saga University, Saga, in 1989 and the Ph.D. degree from Kyoto University, Kyoto, in 2006. Since joining NTT in 1989, he has been engaged in research on system design methodology and an FPGA CAD system. His research interests are in the architecture of programmable telecommunication nodes and high-level system design methods. He is a member of IEEE, the Association for Computing Machinery, the Information Processing Society of Japan, and the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan.

**Daisuke Shirai**

Media Innovation Laboratory, NTT Network Innovation Laboratories.

He received the B.E. degree in electrical engineering and the M.E. degree in computer science from Keio University, Tokyo, in 1999 and 2001, respectively. He joined NTT Network Innovation Laboratories in 2001. Since then, he has been researching super-high-definition imaging systems and their transmission technology. His current interests are in terabit-level image transmission and robust image transmission.

**Tatsuya Fujii**

Senior Research Engineer, Supervisor, Media Innovation Laboratory, NTT Network Innovation Laboratories.

He received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Tokyo, Tokyo, in 1986, 1988, and 1991, respectively. Since joining NTT in 1991, he has been researching image processing and image communication networks. In 1996, he was a visiting researcher at Washington University in St. Louis, USA. He is a member of IEEE, IEICE, and the Institute of Image Information and Television Engineers.

---