A Program Production System utilizing ID and File-data over IP Networks

Shinya Takeuchi*
Yutaka Kaneko*
Makoto Yamamoto*
Masahiro Shibata*
Atsushi Narumi*
Shunji Sunasaki**

NHK (Japan Broadcasting Corporation)*
NHK Engineering Service Inc.**

Abstract

This paper proposes a program production system over an IP network which uses and integrates UMID, MXF and non-destructive editing based on editing description. In this system, materials and their metadata on distributed servers can be accessed from anywhere on the network by attaching a UMID to all materials and by wrapping the UMID, metadata, and materials in MXF format. The result of editing is output in the form of an editing description and all materials are referred to by the UMID in the editing description. In broadcasting or completing the program, the video reproduction equipment searches the original source material by the UMID, and processes materials with effects according to the editing description. Images to be superimposed can be created in parallel while editing the program by notifying the program producer only of the UMID. We have implemented the system experimentally, verified its operation, and confirmed the effectiveness.

1. Introduction

Computer and network technologies are rapidly permeating almost every aspect of life. Many devices, not only computers but also household electric appliances, will be connected to each other via the network. The trend applies also to broadcasting stations, with research progressing on the system in a distributed environment for broadcasting stations, exploiting the advantages of IP networks to produce broadcast programs more promptly and efficiently.

There are two possible ways of editing during program production: the traditional method based on VCR, or non-linear editing using hard disks. With the former, as is well known, the main work involves copying material on video tape. To make a program, scenes are copied from many source tapes to a master tape according to EDL, which is decided by the program producer. To change even just one superimposing image or scene, it is necessary to repeat the same operation using almost all the original materials. With the latter method, a simple non-linear editing system is possible. However, it is necessary to convert all materials to the same format to suit the equipment, and compatibility with other equipment is low. With both methods, editing cannot be started until all video and image materials to be used are ready, because editing involves processing the materials themselves.

To improve the efficiency of program production, non-destructive editing is proposed, by which editing is based on an editing description language and the final program is made from original materials according to the description [1]. With the progress of computers, some technologies for processing file-based audio-visual materials have already been proposed. The Unique Material Identifier (UMID) has been proposed as an ID which specifies materials uniquely, and the Material eXchange Format (MXF) has been proposed as a file format for exchanging materials on a network. However, little has been done toward implementing and integrating these technologies effectively.

On an IP network, every type of data can be transferred in the same way and interactively. By connecting equipment on a network, it is possible to share not only materials but also equipment. This will change the work flow of program production, and promote the expansion of non-destructive editing techniques on networks. A unique ID and a unified format will be necessary to share materials on the network.

This paper reports a program production system which we have newly developed by extending the technique of non-destructive editing to program production over an IP network and by applying UMID and MXF.

2. System Overview

The experimental system is based on distributed processing on a network and the editing description, as shown in Figure 1. The system consists of editing equipment, equipment for creating images for superimposing, video reproduction equipment, and material servers. These are connected via IP network.

The system allows materials to be accessed from anywhere and every file on the network is wrapped by the unified format. The system allows non-destructive editing

based on the editing description; actual image processing is performed in real time only at the last step. Modifications are immediately reflected in the final video output by changing the editing description. Because original materials are not processed during editing, the work of program editing and image creation can also be done in parallel and it is easy to reuse materials. We have adopted UMID as the material ID and MXF as the unified format.

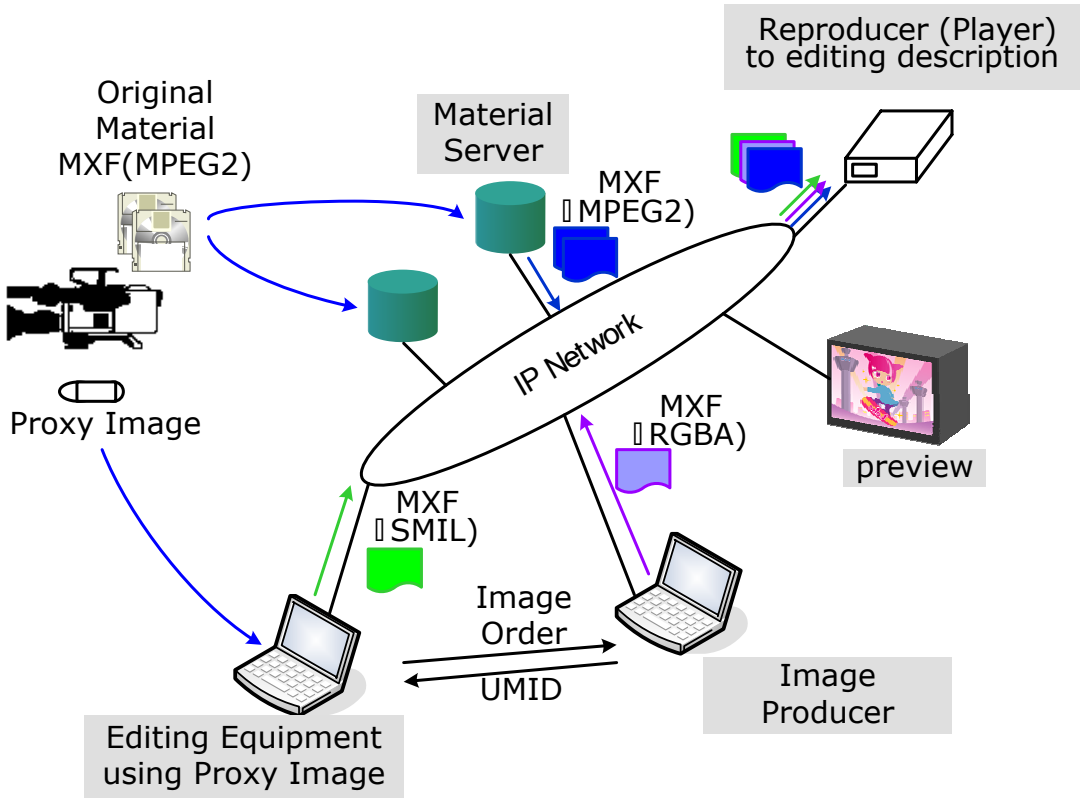

Figure 1: Experimental System of Program Production on a Network

Original materials recorded and used for program production are stored on servers distributed on the network. All materials are given a UMID and are wrapped by MXF. These material images are reduced in size and compressed to form a proxy image which is referred to during editing in a stand-alone environment. In this system, the same UMID should be given to an original material and its proxy image.

By assigning a unique ID to a material, the material can be specified even if it is moved to another server, and it can be used from anywhere. Furthermore, by wrapping materials with a UMID and its metadata in a unified format, any equipment can read metadata of materials and learn about the material. This system has a resolver which resolves a UMID to a network address. When accessing an original material specified by a UMID, it is necessary to ask the resolver to identify the network address of the material.

Each part of the system is explained below.

2.1 Program Editing

Figure 2(a): Example of editing (main display)

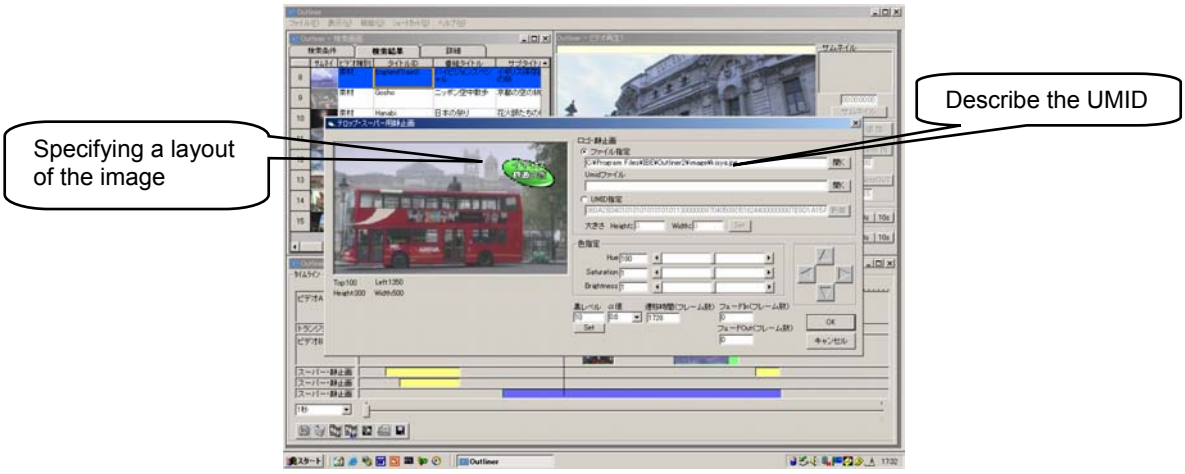

Figure 2(b): Example of editing (configuring a superimposing image)

A program producer performs editing using proxy images with editing equipment in a stand-alone environment such as a laptop computer. Editing is done using a GUI, and proxy images allow the editing to be done with previewing videos. Figure 2 shows an operation screen of the editing software which we have developed. Transition effects and color correction controls can be specified in addition to specification of video sources with a time-line.

After editing is finished, the result is output in the form of an editing description. All materials are referred to by UMIDs in the editing description. The output editing description is assigned the UMID prepared beforehand for the program as a planned program code, and is wrapped by MXF to be used on the network. It is not necessary to store these description files wrapped by MXF on a particular server; they have only to be saved in a shared folder of each appliance.

When an image to be superimposed has not been completed, the producer acquires only its UMID and works with a dummy image. Because the image itself is not

processed here, the image is not needed; only the UMID of the image needs to be written in the editing description. Simple text captions with decorations can be created directly from the description, rather than being prepared in the form of an image.

2.2 Image Creation

An image for superimposing can be created in parallel while editing the program by notifying the program producer only of its UMID. When the image is completed, it is wrapped by MXF with the UMID notified to the program producer, and is stored in a shared folder to be used on the network.

We have developed a plug-in module for commercial painting software to save the images as MXF files.

2.3 Program Playing

When a player plays back an editing description, it switches and mixes the original materials with transition effects, and reproduces the program in real time by following the editing description made using the proxy images. During this reproduction, the original materials on the network are searched automatically form the UMID.

On a terminal connected to the equipment, all MXF files and their metadata can be searched and accessed on the network (Figure 3). From the metadata, the operator can determine which program is to be played (an MXF file wrapping an editing description is a program).

Some programs are almost the same except for captions or a few scenes, such as a preliminary program announcement. With the traditional work flow, as many tapes as the number of programs must be prepared, which is done by repeating almost the same process. We have extended the editing description to describe a particular part by conditions, and to treat an editing description file as a single material. These functions make it possible to describe multiple versions of a program, and so by making one main material without captions, captions and some differences can then be automatically added according to conditions such as the broadcast day or broadcast media.

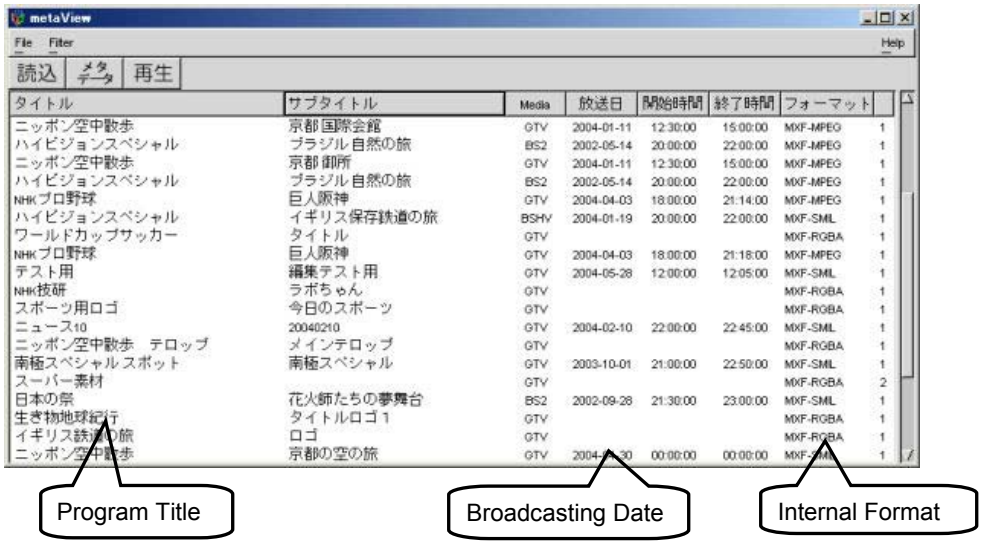

Figure 3(a): GUI for displaying MXF files on the network

Figure 3(b): GUI for displaying detailed metadata of a MXF file

3. Technical Details

We examined and implemented the following technologies in producing the prototype of this system, as explained in detail below.

3.1. UMID

UMID is a unique identifier for audio-visual material. It is standardized by SMPTE [2], and can be generated automatically without reference to a central database. It enables a material to be identified uniquely from anywhere, even if the material has moved.

There are two versions, basic UMID and extended UMID. We use the basic UMID which has a length of 32 bytes. An MXF file has some UMID in its File Header, and we use Material Package UMID as the UMID for identifying the material.

3.2 MXF

MXF is a file format for exchanging audio-visual material on the network, standardized by SMPTE [3]. An MXF file consists of sequences of KLV-coded packets [4] and the data of a packet is identified by a unique 16-byte key. Hence, a simple MXF decoder can analyze the key and identify what the file is, even if the contents of the file can not be decoded.

An MXF file has a File Header and a File Body (and a File Footer). A File Header includes information about the main body, such as type of contents, coding method, UMID, and structure of tracks and metadata of the file. Information about the context of the contents can be recorded in the File Header. For this description, DMS-1 (Descriptive Metadata Scheme-1) is being standardized as SMPTE Standard 380M ("Material Exchange Format (MXF) Descriptive Metadata Scheme -1"). We use DMS-1 in order to describe information containing the title of the material, the broadcasting time, and production time.

Optional patterns can be defined as the complexity of the file structure in the MXF standard. We use only MXF operational pattern 1a (op1a), in which a single item is wrapped in a single MXF file package.

MXF can contain a variable format of signals as the unified format in the File Body, which is called a Generic Container. In this system, we have wrapped two types of data in the Generic Container of the File Body. One is MPEG2@HL (1440x1080, 50Mbps) as an original material. MPEG2 (video elementary stream) is wrapped according to SMPTE Standard 381M (standardization is now progressing). We chose frame wrapping in the system, and have defined local tags in the Primer Pack of MPEG2@HL, because it is not standardized yet. The other type of data is an image with an alpha channel for superimposing. The image with an alpha channel is wrapped by MXF as one picture frame of uncompressed RGBA video (we referred to SMPTE Standard 384M “MXF Mapping of Uncompressed Pictures into the Generic Container”). We have developed and implemented an original library for processing MXF files in this system.

One more type of file exists on the system, which is the editing description. The editing description formed by SMIL (explained in the following section) is also wrapped by MXF for exchange and identification by UMID on the network in the same way as video and images. SMIL is in the style of XML; XML text can be contained as dark metadata of the Header Metadata in MXF, so we defined a proprietary format to wrap SMIL as MXF in the XML text part of the Header Metadata of the File Header, in the same manner as dark metadata. The file has no Generic Container data.

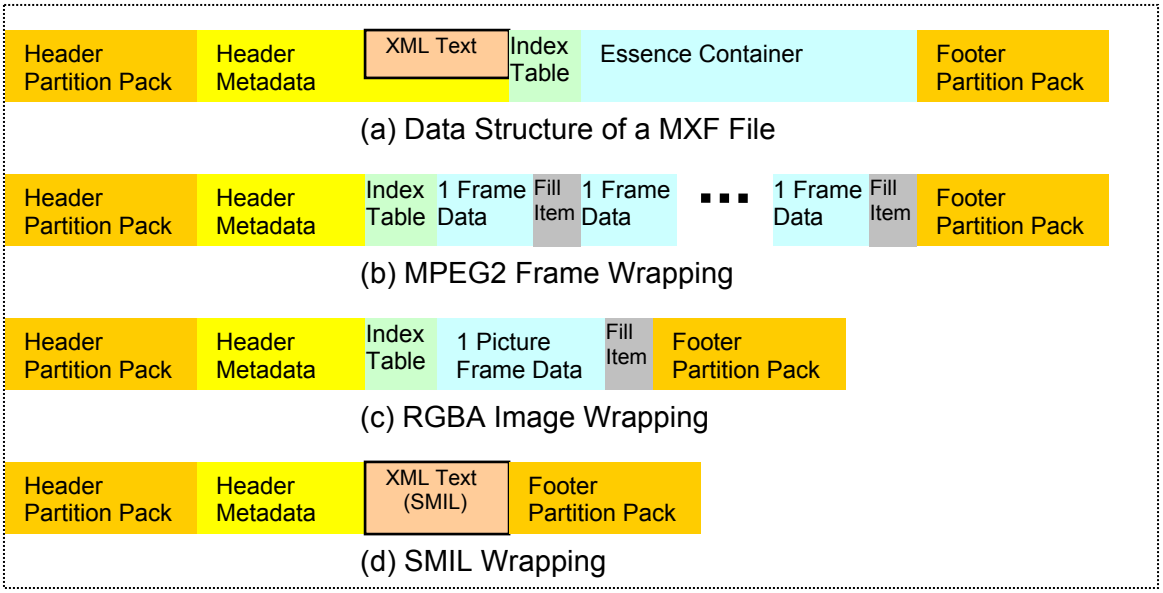

Figure 4: Data Structure of MXF Files in the System

We have had a problem in working with the Index Table array of MPEG2 frame wrapping. We describe index table material by a frame. According to the standard, however, by using one index table array and 4-byte local tag, only 5957 frame data can be contained, and it can wrap an image for only 4 minutes. One solution may be to lengthen the local tag to 6 bytes, but there is no definition about lengthening the local tag in standards. Another possible solution is to partition some Index Tables.

3.3 Editing Description

EDL (Edit Decision Lists) is used conventionally to describe edit information, but it is inadequate for program production because it can describe only cut points and cannot describe various effects, and also because it cannot specify the original materials. To overcome these shortcomings for professional program production, for example, SMIL and MIKSS (Media object InterlocK and SynthesiS) has been proposed as an edit information description [6].

In the system, we have implemented SMIL2.0 Extension [7]. SMIL is an authoring language for interactive multimedia presentations for the Web, standardized by W3C [8]. It realizes synchronized playing of multiple media and positioning of layout, but it is still insufficient for describing the editing of program production, so SMIL 2.0 Extension has been proposed, which has the following functions: independent time code, partial playing of material from the middle, transition effects, color correction control, mixing images with alpha channel, specifying contents by UMID, and creating captions with font decoration and fringes.

SMIL Executor (which was developed by SONY Corp.) is used as the equipment which mixes and changes materials in real time, and reproduces the program according to an editing description written by SMIL2.0 Extension [9]. There is no delay in rendering images. The features of SMIL Executor are the following.

Table 1: Specification of SMIL Executor

Video Format	1440x1080, 60i
Compression	MPEG2 MP@HL, 50 Mbps, GOP: M=3, N=15
Effect	Dissolve, Wipe (bar, box, diagonal, iris, ellipse, barnDoor, clock, split)
Color Correction	Brightness, Saturation: 0.0-2.0, Hue: 0-360deg
Image Overlay Format	4:2:2:4, 1920x240 pel x 3 planes

3.4 Address Resolution

In a description language like SMIL and HTML, content is normally specified by a static address based on its location, such as URL (Uniform Resource Locator). But when content moves, there is the problem that contents cannot be specified. In this system, materials are always specified not by location but by UMID, and the system has a resolver UMID for network addresses. We have developed the resolver, which is applied the broadcasting system based on distributed object architecture and communication [5].

When an MXF file is created or appears, it registers its UMID and its address on the resolver. When searching a material referred to by UMID, the resolver is asked to resolve the UMID. We use CORBA (Common Object Request Broker Architecture) as a platform for object communication in a distributed environment and IOR (Interoperable Object Reference) as an address.

4. Summary

We have developed a program production system over an IP network by applying and integrating the editing description, UMID, and MXF. The system enables materials to be used from anywhere; original materials are not processed and are easy to reuse; program editing and image creation can also be done in parallel; and modifications are immediately reflected in the final video output by changing the editing description.

The concept of the system is subtly different from the conventional program production, but we have demonstrated the effectiveness of program production using this system. We also verified the practical effectiveness of UMID and MXF. Even though the system is one example of implementing UMID and MXF on a network, it will serve as a springboard for discussion and implementation.

We intend to transmit contents via streaming and to apply the system to uncompressed HDTV videos. We will examine an MXF application to store multiple media and tracks.

Acknowledgements

The authors gratefully acknowledge the help of Sony Corporation.

References

[1] P. Laven, “Computer-based production systems for TV”, EBU Technical Review Editorial, No. 299, July 2004

[2] SMPTE Standard 330M, "Unique Material Identifier (UMID)”

[3] SMPTE Standard 377M, "Material Exchange Format (MXF) File Format Specification (Standard)"

[4] SMPTE Standard 336M, "Data Encoding Protocol using Key-Length-Value”

[5] Y. Kaneko, M. Yamamoto, Y. Izumi, "Plug & Play System for Advanced Networked Broadcasting Station”, 144th SMPTE Technical Conference

[6] K. Iguchi, Y. Kaneko, O. Mizuno, Y. Itoh and Y. Shishikui, “Scene Description Language for Video Program Production in Broadcasting Station”, IEICEJ Tech. Rep., IE2000-06, 2001 (in Japanese)

[7] W3C, “SMIL2.0 Extension for Professional Multimedia Authoring-Preliminary Investigation”, http://www.w3.org/TR/SMIL2-AuthExt/

[8] W3C, “SMIL2.0 Recommendation”, http://www.w3.org/TR/smil20/

[9] K. Sayama, Y. Shibata, S. Kawa, M. Hatakeyama, T. Suzuki, "HD non-destructive editing system based on extend SMIL", ITE Annual Convention 2004, 21-6, 2004 (in Japanese)