

The MXF mastering format

The initiative aims to streamline file-based operations.

BY DAVID AUSTERBERRY

As broadcasters move from videotape to file-based operations, it is apparent that some form of standardized format is needed to exchange content between platforms and systems. Analog PAL and NTSC over 75Ω coax is just one example. This philosophy was carried forward

with the 270Mb/s SDI format. The latter had a couple of glitches at the start with field flags and data rate changing from 243Mb/s to 270Mb/s, but it soon settled down into a ubiquitous standard.

Not all interchange is over a cable, and videotape has provided an alternative. That has been where propri-

etary formats emerge. Although most broadcast tape formats have been standardized by SMPTE, they tend to be linked to a single manufacturer. The main exceptions are consumer formats such as miniDV.

Many ways to exchange video and audio as files have been developed, but they are predominately proprietary to a single manufacturer as well. Broadcasters' requirements for file exchange are similar to a tape; it needs a video track, several audio tracks and time-code wrapped in synchronism. Additionally, it requires an assortment of meta-

data, including tape number and program ID. Such wrappers include QuickTime and OME. For real-time transfer, there are MPEG wrappers such as the MPEG-TS or transport stream.

In the groundbreaking joint EBU/SMPTE report "Harmonized Standards for the Exchange of Program Material as Bitstreams" (1998), the groups identified a requirement for wrapping program content into suitable containers that would ensure complete interoperability in a future networked production environment. The Pro-MPEG Forum

developed this idea into Material eXchange Format (MXF), a streaming wrapper that is now defined in a set of SMPTE standards.

What is MXF?

MXF is an open object-based file format for the interchange of audio-visual content as data and associated

metadata. It is now a set of SMPTE standards. MXF does not define the audio and video (essence) formats; instead it wraps existing formats, both compressed and uncompressed.

However, the noble aim of interoperability floundered once "plug-fests" were carried out to exchange data between equipment from different vendors. It soon became apparent that the 700 pages of standards were too complex, giving too many options. Most vendors fell back to basic Operational Patterns (OPs), merely a replacement for a tape cassette. For more complex OPs, if a suitable subset could be found, then the different vendors could concentrate on the simpler task of achieving interoperability with that subset.

OP1a interleaved	MXF mastering format
Lengthy tape rewrites	Simple to repurpose files
Bandwidth hogging file transfers	Built-in metadata storage
Incompatible proprietary formats	Highly interoperable

Table 1. The early applications of MXF used interleaved essence, the mastering format has several advantages.

metadata. It is now a set of SMPTE standards. MXF does not define the audio and video (essence) formats; instead it wraps existing formats, both compressed and uncompressed.

However, the noble aim of interoperability floundered once "plug-fests" were carried out to exchange data between equipment from different vendors. It soon became apparent that the 700 pages of standards were too complex, giving too many options. Most vendors fell back to basic Operational Patterns (OPs), merely a replacement for a tape cassette. For more complex OPs, if a suitable subset could be found, then the different vendors could concentrate on the simpler task of achieving interoperability with that subset.

Driven by Turner Broadcasting, a group of interested parties, mainly vendors, created an MXF mastering format. (See "Principal develop-

ers of the MXF mastering format" on page 14.) The subset addressed the requirements of publisher-broadcasters. The format is now sponsored by Advanced Media Workflow Association (AMWA), formerly the AAF. The group continues to drive it forward to a working proposition. Without

Broadcasters' requirements

The large multichannel publisher-broadcasters have to perform many processes between receiving a program master tape and airing a pro-

gram. It is becoming normal to create eight or more language versions. Programs may need alternate segmenting to suit the advertising break structures in different countries. Broadcasters

may want to create an edit suitable for minors or religious sensibilities. All this could add as many as 20 or 30 versions of the original master. Formerly, these were edited and recorded as new transmission master tape for ingest to the playout servers. This led to the classic problem of a serious tape handling operation, plus expensive time in an online edit suite.

Apart from traffic, the promotions department has to make trailers for programs, again in many versions. Engineers at Turner felt that advantages of moving from videotape to server operations would not be fully realized if all the different versions had

to be stored as new broadcast master files. Tape handling problems become storage and network costs.

The NLE provided a the model for cost-effective operation. Source files are referenced from the timeline. Several different final cuts can be made, all referencing the same source files. The timeline is a compact file of metadata. The large media files do not need to be copied to make the versions. This leads to big savings in expensive online storage space.

Driven by Turner Broadcasting, a group of interested parties, mainly vendors, created an MXF mastering format. The subset addressed the requirements of publisher-broadcasters.

gram. It is becoming normal to create eight or more language versions. Programs may need alternate segmenting to suit the advertising break structures in different countries. Broadcasters

The requirement for versioning programs and making trailers is much the same as editing. If a timeline could be created for each version, the final cut could be published to the playout

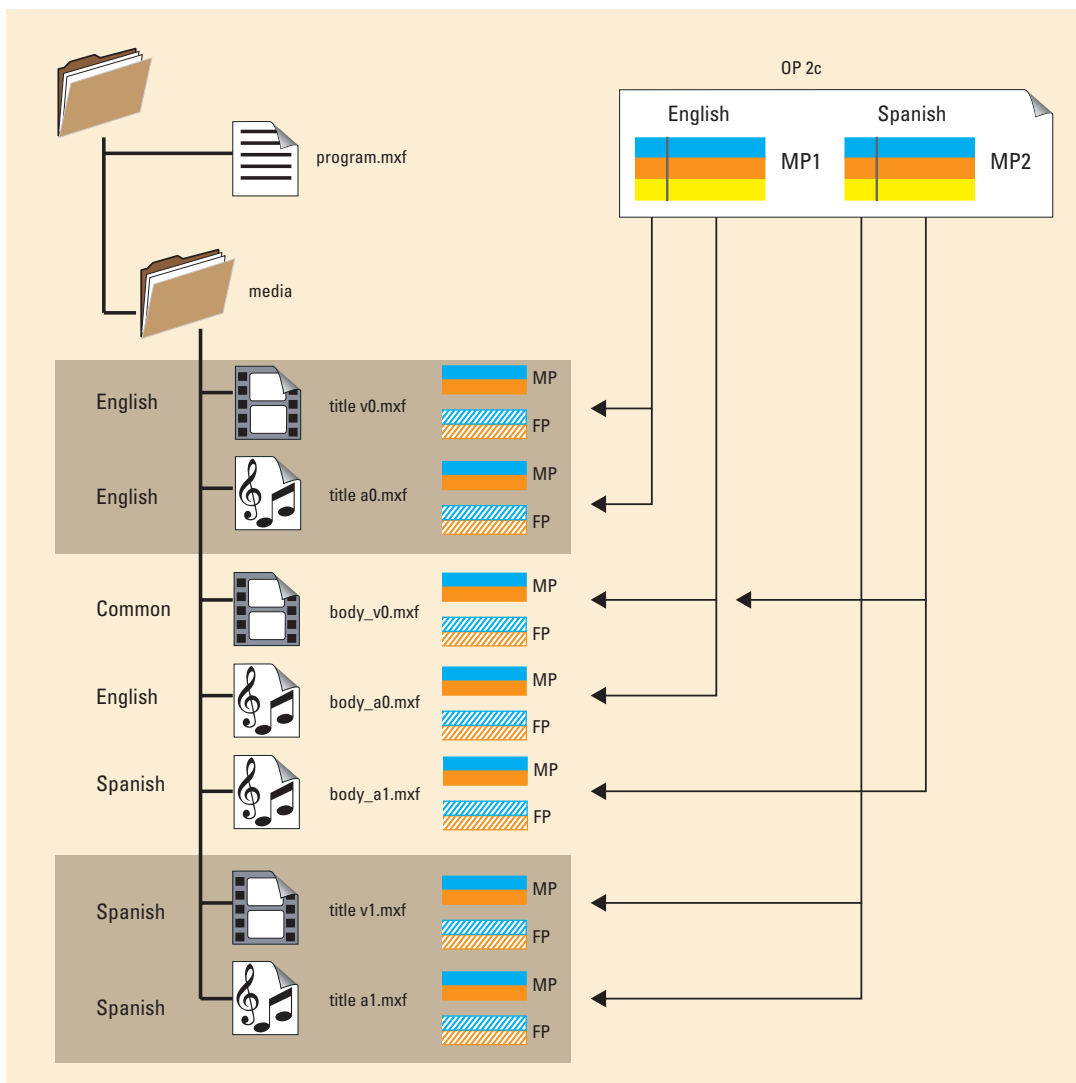


Figure 1. The MXF mastering format simplifies the handling of media files with several language tracks.

BroadcastEngineering
THE JOURNAL OF DIGITAL TELEVISION

Reprinted from July 2007 • www.broadcastengineering.com

server and deleted after use. Only the master program files need to be stored indefinitely.

The NLE is a closed system, although some vendors allow source files and timelines to be stored as AAF data. The versioning requirement was for something similar but simpler, easing the problems of interoperability between vendors. MXF already had many of the structures in the OPs, so it represented a good starting point.

Problems with MXF

MXF offers too many choices; interoperability is too difficult. There are about 25 standards. Attempts to exchange files between vendors revealed many issues stemming from differing interpretations of the standards.

The MXF standards support several essence formats, especially for video. The Mastering Format limits the choice to MPEG-2 encoding: I-frame at 50Mb/s, and long-GOP at 25MB/s or 15MB/s. The I-frame format can be repurposed, and the long-GOP format is suitable to store on playout servers. Audio is handled in the BWAV format.

The MXF mastering format is an Application Specification (AS) that will detail a profile of MXF. Interested

vendors can produce equipment that conforms to the specification, with the intention that this should deliver the interoperability that the broadcasters demand.

MXF mastering format

The first goal was to simplify the specification. The second was to match the AS to the processes in the key areas of ingest, archive, repurposing and playout. A further require-

ment was inventory distribution. It soon became evident that multiple OPs would be needed to cover these applications: ingest, repurposing, archive, partial restore and playout.

essary assets can be pulled from the inventory in the future to reconstruct a specific version. One of the advantages of storing an asset inventory is that a copy can be sent to a remote site. As an MXF container, versions can be created from the copy inventory by a completely different set of products and systems, as long as they are compliant with the MXF mastering format AS. This provides freedom from vendor lock-

The MXF mastering format will remove many of the tape handling and QC operations from the reversioning process.

in and allows program distribution to independent sites that may have completely different equipment. It also abstracts the asset from specific storage platforms, so as equipment becomes obsolete, the asset can migrate in a standard format.

Ingest

The requirements at ingest are minimal, save the need to preview work during ingest, before the process is complete and the file is closed (partitioned).

Repurposing

This is the key area for the introduction of cost savings. The mastering format must support different commercial break structure (segmentation), different languages (including subtitles and closed captions), different opening titles and closing credits, and different output formats, such as 4:3 and 16:9, and HD and SD.

Archive

The archive should be able to store everything related to an asset, that is a single episode of a show. All the new files created during repurposing should be stored as the inventory for that asset. That means that the nec-

essary assets can be pulled from the inventory in the future to reconstruct a specific version. One of the advantages of storing an asset inventory is that a copy can be sent to a remote site. As an MXF container, versions can be created from the copy inventory by a completely different set of products and systems, as long as they are compliant with the MXF mastering format AS. This provides freedom from vendor lock-

in and allows program distribution to independent sites that may have completely different equipment. It also abstracts the asset from specific storage platforms, so as equipment becomes obsolete, the asset can migrate in a standard format.

Partial restore

To meet the goals, the mastering format must support partial restores. In most broadcast systems, the archive is a robotic data tape library. Normally if any operations are to be performed on a file, the entire file is restored to online disks.

For large media files, this can be inefficient. Suppose the promotions department is making a 30-second trailer for a movie, and then the two-hour file is restored. This occupies a tape drive and network bandwidth, when perhaps two minutes is needed for editing. Using time-code markers from a browse resolution editor, it must be possible to restore a time slice of the entire file using the index table.

The developers realized that the most efficient way to store inventories was to keep the essence (video and audio) as external atomic files, not stored in the main MXF container. So far, most implementations of MXF

have interleaved video and audio in a single container. This lacks the flexibility needed for multilingual operation. Instead, the main MXF container stores metadata, including references or pointers to the essence files.

One goal for managing the archive is to minimize the number of essence files. Another goal is to limit transfers between the archive and the online storage to only those that are essential. This saves storage network resources. Both of these goals lower cost.

Playout

In the analysis of broadcast workflows, files are referred to by function rather than OP, such as program components, simple program versions, complex program versions and program inventories. Program

The essence is separate from the program version, so a video server storing essence need only support OP1a.

components include the essence and metadata and could exist in several resolutions (master I-frame, TX long-GOP and proxy). Complex program versions include multilingual soundtracks and different program sequences (titles, body and credits).

Program components can be represented by OP1a containers, program versions by OP2b or OP3b, and the inventories by OP3b and OP3c.

The essence is separate from the program version, so a video server

storing essence need only support OP1a. This example demonstrates how interoperability can be simplified by storing components in an atomic form. Figure 1 on page 10 shows a complex program version.

A proxy browse editor could be used to create a version of a program, an OP2b file. Another application could take this file and conform an I-frame source component to a long-GOP transmission master version. The editing applications only need to exchange containers of metadata, which will be small files, rather than transferring large media files.

Another example is the addition of a language track. The additional language can be added by manipulation of the complex program version OP2b files, adding a reference to the

new BWAV audio file container. The audio file is copied to the media server. Contrast this with an interleaved media file, which would have to be opened, the additional track added, and then resaved.

Asset inventory

Several broadcast networks have become global operators. A program could be aired from London, Los Angeles, Singapore and Tokyo. Each transmission center will need to cre-

ate versions in the local languages. This has always been done by flying tape dubs around the world and starting from scratch at each center. The MXF mastering format will remove many of the tape handling and QC operations from the reversioning process.

With the mastering format, broadcasters can now start with the program inventory and add language tracks at will. If necessary, these files could be returned to the originator to form a more complete inventory of the program asset. All that is returned is the additional program component and the complex version metadata.

Goals

The MXF mastering format aims to meet the needs of major broadcast networks that distribute programs in multiple formats and languages. It promises to lower costs by providing a standard for interoperability between equipment suppliers and simplifying the processes of reversioning. Table 1 on page 12 contrasts the initial implementation of MXF as an interleaved file wrapper with the mastering format.

It also provides a simpler way to archive material in a form that can be repurposed and copied to independent transmission centers. It simplifies a complex set of standards to a manageable subset that can be used to build cost-effective video storage and distribution networks.

For more information on the AMW Association, visit www.aafassociation.org, and for more information on MXF standards, visit www.smpte.org. **BE**

Principal developers of the MXF mastering format:

- EMC²
- Marquis
- MetaGlue
- Omneon
- Opencube
- Pro-Bel
- Quantum
- Snell & Wilcox
- Softel
- TMD
- Turner Broadcast Technology

Reprinted from July 2007 • www.broadcastengineering.com



Reprinted with permission from the July 2007 issue of *Broadcast Engineering* magazine. (www.broadcastengineering.com) Copyright 2007, Penton Media, Inc. All rights reserved.