

Technik: MXF-File-Format



Ein File für alle?

MXF steht für ein neues File-Austauschformat, das viele Probleme im Postproduction- und Broadcast-Bereich lösen könnte. Derzeit wird noch mit Hochdruck an der Standardisierung gearbeitet, erste Applikationen sollen während der IBC2001 gezeigt werden. Dr. Nick Wells von BBC Research & Development ist als maßgebliches Mitglied des ProMPEG-Forums am Standardisierungsprozess beteiligt. Er gab www.film-tv-video.de Auskunft über Inhalte, Anwendungen, Vor- und Nachteile des Formats.

TEXT: NONKONFORM • GRAFIK: DR. NICK WELLS, BILD: NONKONFORM

? Mit MXF soll ein neues File-Austauschformat, geschaffen werden, das Video-, Audio- und Metadaten enthält. Es wurde gemeinsam vom ProMPEG-Forum und der Advanced Authoring Association (AAF) entwickelt. Können Sie die Zielsetzung von MXF einfach und in kurzen Worten erklären?

Dr. Nick Wells: Um diese Frage zu beantworten, muss ich etwas weiter ausholen und von der bisherigen Arbeit des ProMPEG-Forums berichten.

Das ProMPEG-Forum nahm seine Arbeit ursprünglich auf, um Interoperabilität von MPEG-Datenströmen in einer professionellen Umgebung zu testen. Diese Formate sind:

- a) MPEG über SDTI/CP, das unkomprimierte Studio- SDI-Digital-Infrastrukturen nutzt, um komprimierte MPEG-Signale, in erster Linie 50-Mbps-Signale, zu transportieren.
- b) MPEG-Transport-Ströme über ATM-Netzwerke



Dr. Nick Wells

Sehr schnell wurde in dieser Diskussion klar, dass es künftig zunehmend nötig sein wird, Programm-Material unterschiedlichster File-Formate zwischen verschiedenen Servern austauschen zu können. Dieser Bedarf wird sicher wachsen – im Gegensatz zu dem traditionellen Austausch von

»Streaming«-Formaten.

(Anmerkung der Redaktion: Dr. Nick Wells versteht im gesamten Artikel unter Streaming-Formaten die bislang üblichen Videostandards für die isosynchrone Signalübertragung, wie etwa SDI-Signale).

Allerdings gab es zum Zeitpunkt dieser Diskussionen noch kein gemeinsames File-Format, das sich für den

Programmaustausch zwischen Servern und Editing-Stationen unterschiedlicher Hersteller geeignet hätte. Die Mitglieder des ProMPEG-Forums einigten sich deshalb darauf, dass sie gemeinsam ein File-Austausch-Format definieren wollten – wohl wissend, dass es auch mit diesem neuen Format nach wie vor notwendig wäre, das jeweilige interne, native File-Format des Servers und Editing-Systems

des jeweiligen Herstellers in das neue, gemeinsame File-Austausch-Format zu konvertieren.

Selbstverständlich sollte das File-Austausch-Format Video-, Audio- und Metadaten enthalten und auch andere Informationen wie etwa Index-Tables, die die Position innerhalb eines Files in Bezug zum Timecode setzen. Die wichtigsten Anforderungen an das neue Format sind:

- Das Format solle in erster Linie dazu dienen, vollständige, fertige Programme oder Programmsegmente zwischen Geräten und Equipment unterschiedlicher Hersteller und

Organisationen austauschen zu können.

- Das Format sollte klar strukturiert und einfach zu implementieren sein.

- Es sollte möglich sein,

mit einem File schon vor Ende eines File-Transfers arbeiten zu können, im Prinzip so, wie man es auch von Streaming-Formaten her kennt. Diese Anforderung ist dann wichtig, wenn große Files über langsame Netzwerke übertragen werden und es bedeutet, dass die unterschiedlichen Komponenten innerhalb des Files bis zu einem bestimmten Grad verzahnt sein sollten. Im Klartext: Es würde nicht möglich sein, zuerst alle Video-, dann alle Audio- und schließlich alle Metadaten zu übertragen.

- Die Struktur des Formats sollte kompressionsunabhängig sein. Insbesondere sollten sich die Metadaten möglichst unabhängig vom jeweiligen Video- oder Audio-Kompressionsschema übertragen lassen.

Nun waren bei verschiedenen ProMPEG-Meetings auch Avid-Vertreter anwesend, und sie verfügen über einen sehr großen Erfahrungsschatz auch mit älteren File-Formaten. Zudem war Avid sehr aktiv bei

der Etablierung von AAF, was nunmehr ein Standard-File-Format für Authoring- und Editing-Plattformen im Fernsbereich ist. Das AAF-Format bietet zwar sehr viel Funktionalität innerhalb einer Editing-Umgebung, es erfüllt allerdings zwei der oben beschriebenen Grundanforderungen nicht: Es ist nicht sehr einfach zu verstehen und es kann nicht bei laufender Übertragung mit der Wiedergabe begonnen werden. Aus diesem Grund entschied sich das ProMPEG-Forum, ein neues Format, das Media eXchange Format (MXF) zu entwickeln, um alle Anforderungen zu erfüllen, die man für ein einfaches File-Austausch-Format definiert hatte.

Ein Großteil der Arbeit an den neuen File-Austausch-Format bestand darin, File- und Datenstrukturen zu entwickeln, mit denen sich komplexe

Metadaten übertragen lassen, wie sie bei modernen Produktionsverfahren entstehen.

? Wie kam die Zusammenarbeit des AAF und des ProMPEG-Forums zustande und weshalb hat sich die BBC in der Entwicklungsarbeit des MXF-Formats so stark engagiert?

Dr. Nick Wells: Als die Arbeit an MXF begann, gab es die AAF-Association schon eine geraume Zeit, etliche Hersteller und auch wichtige Programmanbieter wie die BBC, unterstützen AAF. Die allgemeine Sichtweise ist jedoch, dass sich AAF eher für den Editing- und Postproduktionsprozess eignet und weniger für den Austausch von fertigen Programmen zwischen unterschiedlichen Organisationen über möglicherweise sehr langsame Netzwerke.

Unter Berücksichtigung der weiten Akzeptanz von AAF war es wichtig, dass das neue MXF-Format möglichst zu AAF kompatibel sein und die Funktionalität von

Letztlich kann man ein MXF-File als die »Streaming«-Variante eines AAF-Files sehen.

Dr. Nick Wells, BBC Research & Development

AAF eher ergänzen und nicht damit konkurrieren sollte.

Letztlich kann man ein MXF-File als die »Streaming«-Variante eines AAF-Files sehen. Dabei sind die Datenmodelle, die für die Metadaten verwendet werden, in beiden Formaten dieselben. Deshalb sollte es auch Metadaten-Transparenz zwischen Systemen geben, die beide Formate unterstützen.

Es war sehr wichtig, die Kompatibilität zwischen MXF und AAF zu garantieren und deshalb hat die AAF Association auch zugestimmt, am Entwicklungsprozess des MXF-Formats aktiv teilzunehmen. Diese Aktivitäten werden von einem AAF- und einem ProMPEG-Vertreter gemeinsam koordiniert.

Auch die BBC entwickelt derzeit zukünftige Produktions- und Speicher-Infrastrukturen, bei denen vernetzte Server sowie der File-Austausch zwischen diesen

Servern eine zentrale Rolle einnehmen. Standards für den File-Austausch zwischen Systemen unterschiedlicher Hersteller müssen bei solchen zukünftigen Architekturen unbedingt geschaffen werden und vor allem effektiv sein. Zudem ist auch ein konsistente und optimierte Herangehensweise notwendig, denn nur so können zukünftige Produktionsprozesse wirklich effektiv gestaltet werden. In jedem neuen Standard müssen deshalb große Kapazitäten für Metadaten vorgesehen werden, und ebenso wichtig ist es, dass diese innerhalb eines Netzwerks auf transparente und konsistente Weise weitergegeben werden. Die BBC unterstützt die MXF-Initiative daher aus voller Überzeugung und natürlich auch in der Erwartung, dass der Standard rechtzeitig für die Pläne der BBC

verfügbar und kommerziell einsetzbar sein wird.

Teilweise bemängeln Anwender, dass es sehr komplex sei, mit AAF-Formaten zu arbeiten. Nun hängt ja auch das MXF-Format damit zusammen. Wie soll es gelingen, die teilweise existierenden Vorbehalte gegenüber AAF bei potenziellen MXF-Anwendern abzubauen?

Dr. Nick Wells: MXF ist einfacher als AAF und unterstützt auch nicht die ganze Funktionalität, die in AAF vorgesehen ist. Beispielsweise ist MXF nicht dazu in der Lage, komplexe Transitions zwischen zwei Clips in einem File zu beschreiben. Außerdem war es eine der Grundanforderungen ans MXF-Format, die Strukturen so einfach wie möglich zu halten und dabei die

Anforderungen ans neue Format auf den Punkt zu bringen. Unterschiedliche Anwender haben aber auch unterschiedliche Anforderungen in Bezug auf die Komplexität des Formats, und deshalb gibt es bei MXF

MXF ist einfacher als AAF und unterstützt auch nicht die ganze Funktionalität, die in AAF vorgesehen ist.

Dr. Nick Wells, BBC Research & Development

unterschiedliche »Operational Patterns«, die unterschiedliche Komplexitäts-Ebenen unterstützen. Beispielsweise enthält der »Operational Level I« ein einziges Programm oder Programm-Segment. Komplexere Patterns können dagegen mehrere Programme und die Möglichkeit zu Edits zwischen den Programmen enthalten. Diese Pattern sind nützlich, wenn ein Programm in mehrere Länder distribuiert wird und einzelne Szenen entfernt werden müssen. Außerdem können komplexere Patterns auch mehrere Versionen eines Programms enthalten, etwa einmal in voller Bildqualität und einmal in Browse-Qualität.

Eine wichtige Aufgabe bei der Spezifikation von MXF bestand darin, Strukturen zu definieren, die eine möglichst breite

Palette von Metadaten unterstützen. Dabei können natürlich die Details und auch die Terminologie solcher Spezifikationen auf den ersten Blick kompliziert wirken. Dennoch bietet die Struktur auch Funktionen, die aus den praktischen Erfahrungen der Implementierung des AAF-Formats resultieren und die zwar kompliziert wirken, die sich aber bei der praktischen Implementierung von MXF deutlich vereinfachen.

Das ProMPEG-Forum versucht mit viel Aufwand, die MXF-Spezifikationen zu beschreiben und zu erläutern und auch für Ingenieure zugänglich zu machen, die eher aus dem Audio-/Videobereich und weniger aus dem Computerbereich kommen. Diese Referenz-Software-Initiative wird gemeinsam von der EBU, dem ProMPEG-Forum und der AAF Association vorangetrieben. Außerdem sollen diese Referenzen für Firmen zur Verfügung gestellt werden, die ihre eigenen Implementierungen testen wollen und sowohl ProMPEG wie auch AAF organisieren Interoperabilitätstests und –Demonstrationen, etwa während der IBC und der NAB. Das gibt vielen Firmen die Möglichkeit, zusammen zu kommen und Interoperabilitätsfragen gemeinsam zu klären.

In welcher Phase befindet sich die Entwicklung des MXF-Formats aktuell? Nach unserem Kenntnisstand wurde MXF bei der SMPTE schon zur Standardisierung eingereicht.

Dr. Nick Wells: Das Dokument wurde bei der SMPTE erstmals im Mai 2001 eingereicht. Es gab viele Kommentare zu dieser ersten Version, deren Inhalte nun eingearbeitet wurden. Eine verbesserte Version wurde im August neu eingereicht – rechtzeitig zur nächsten Meeting-Runde im September.

Als einer der Hauptvorteile des MXF-Formats wird dessen Format- und Plattform-Unabhängigkeit genannt. Können Sie dies näher erläutern?

Dr. Nick Wells: MXF bietet eine Struktur, die AV-Content zusammen mit den dazugehörigen Metadaten und den Details

der File-Informationen umhüllt und die zudem beschreibt, in welcher Weise die unterschiedlichen Komponenten synchronisiert werden müssen. Diese Struktur ist unabhängig vom Kompressionsformat von Video und Audio. Komprimiertes Audio und Video oder ein Multiplex-Signal aus diesen Bestandteilen, sitzen in dieser Struktur in einer Art »Body Container«, und das in einfacher und klar definierter Weise. Für unterschiedliche Kompressionsformate sind unterschiedliche »Container« spezifiziert. Außerdem ist MXF auch ein »lineares« Format mit einer klar definierten Struktur. Daher ist es unabhängig von jeglicher Speicherstruktur und dem jeweils betreffende Operating- oder Speichersystem. Es gibt bei MXF aber dennoch eine Möglichkeit, Informationen in Paketen zu einzelnen Sektoren zusammenzufassen (etwa 4 kb), um damit eine nachfolgende Verarbeitung sowie Speicher-Operationen zu vereinfachen.

Was sind aus Ihrer Sicht die Hauptvorteile des MXF-Formats?

Dr. Nick Wells: Der Hauptvorteil besteht darin, dass es mit MXF ein Standard-Format für den Austausch von vollständigen Programmen zwischen unterschiedlichen Systemen und Firmen gibt. Außerdem liefert MXF eine Standardmethode, nach der Metadaten innerhalb dieser Files ausgetauscht werden können, unabhängig vom verwendeten Audio- und Videokompressionsformat. Man kann sich kaum vorstellen, wie Server-, Netzwerk- und File-Austausch-Architekturen künftig ohne einen solchen Standard funktionieren sollten. Zudem kann MXF auch ein ideales Archiv-Format sein, wenn gefordert ist, dass auch die Metadaten eines Programms zusammen mit dem eigentlichen Programm archiviert werden sollen. Ein weiterer Vorteil von MXF besteht darin, dass es als »Quasi-Streaming-Format« eine sehr bequeme Brücke zwischen den Bereichen schlägt, in denen mit konventionellen »Streaming«-Formaten gearbeitet wird (etwa SDI/SDTI im Bereich Ingest und Playout) und zwischen dem Editing- und Produktionsbereich, wo

eher mit File-Transfers und dem AAF-Format gearbeitet wird.

Da MXF auf KLV-Kodierung (Anmerkung der Redaktion: Key-Length-Value-Kodierung) basiert, ergeben sich weitere Vorteile, denn bei dieser Art der Kodierung wird jedem einzelnen Bestandteil innerhalb der linearen Struktur des Files ein Identifizierungscode vorausgeschickt, der genau angibt, welcher File-Inhalt auf welcher Länge gleich folgen wird. Das macht es für Hard- und Software-Systeme sehr einfach, die Files zu analysieren und jene Passagen auszulassen, die für die jeweilige Anwendung uninteressant sind.

MXF verfügt auch über Mechanismen, mit denen sich harte Schnitte zwischen einzelnen Videoclips und Überblendungen zwischen Audioclips definieren lassen. Damit können sich von einem einzigen File unterschiedliche Programm-Varianten erstellen lassen, etwa, wenn bestimmte Szenen in bestimmten Ländern herausgeschnitten werden müssen.

Wie viele Hersteller unterstützen derzeit das MXF-Format und wer sind neben den beiden Foren die treibenden Kräfte an dem Projekt?

Dr. Nick Wells: Das ProMPEG-Forum hat derzeit über 40 Hersteller-Mitglieder, die alle mehr oder weniger intensiv die Entwicklung von MXF unterstützen. Etwa 15 dieser Unternehmen sind aktiv an der Entwicklung der Spezifikationen beteiligt und etliche dieser Firmen werden während der IBC im September erste MXF-Implementierungen zeigen. Andere treibende Kräfte des Projekts kommen von der EBU, die eine Referenz-Software für den Einsatz der MXF-Spezifikationen entwickelt. Außerdem fungiert die SMPTE als »Due process« Standardisierungs-Body für öffentliche Diskussionen und Dissemination des MXF-Standards und der MXF-Spezifikation.

Wie soll die weitere Entwicklung des MXF-Formats verlaufen?

Dr. Nick Wells: Der Hauptteil des Standardisierungsprozesses sollte bis zum Frühjahr 2002 abgeschlossen sein. Da etliche der beteiligten Firmen Codes als Teil des Standardisierungsprozesses entwickeln und auch weil sich die Kooperationen mit dem ProMPEG-Forum, der AAF Association und der EBU gut entwickeln, gehe ich davon aus, dass vermutlich schon kurz vor Standardisierungsfertigstellung gegen Ende des Jahres 2002 auch ein fertiges Produkt verfügbar ist.

Weiterführende Arbeiten an MXF werden im geringeren Umfang noch notwendig sein, um das Format zu erweitern, etwa um neue Bodies für weitere Essentypen zu definieren, die im ersten Standardisierungsdokument noch nicht enthalten sind (siehe Kasten hierzu).



Was sind die wichtigsten technischen Eckdaten des MXF-Formats?

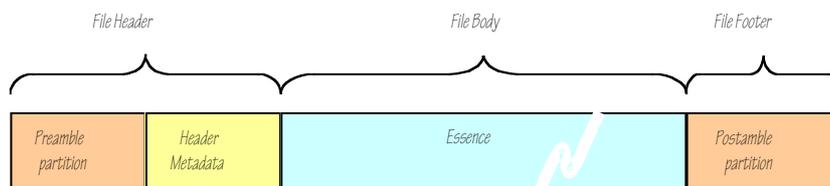
Diese Frage lässt sich nicht sehr kurz beantworten. Aber die grundlegende Struktur eines MXF-Files kann man wie folgt zusammenfassen:

BASIS-STRUKTUR

Das MXF Format orientiert sich am grundlegenden typischen Aufbau, den viele Files haben. Das sind in Überblick:

- Ein File-Header, der die Information über das File als Ganzes enthält. Darin sind auch Informationen über das Operational Pattern des Files enthalten, um rasch Detektieren zu können, ob das ankommende Signal decodiert werden kann.
- Ein File-Body, der sich aus der Essenz des Containers zusammensetzt
- Ein File-Footer, der das Ende des Files festlegt.

Das einfachste MXF-File ist in folgender Abbildung zusammengefasst:



Zusätzlich können MXF Files optional eine Index-Tabelle enthalten, die eine schnelle Konvertierung etwa von Timecodes sorgt. Der Index-Table kann segmentiert sein, er kann vor, nach oder auch mit den Essenzdaten-Segmenten gespeichert werden.

MXF-Files können optional außerdem auch »File-Continuers« enthalten, die in Intervallen innerhalb des Bodies auftauchen und folgende Funktion haben:

- erhöhte Robustheit der Metadaten-Informationen durch die Wiederholung der Header-Metadaten.
- Multiplexing unterschiedlicher Body Container.
- Distribution von Index-Tabellen in kleinen Stücken, sinnvoll etwa für eingeschränkte Speicher-Devices.
- Bereitstellung von Index-Tabellen »per stream«, die positionsunabhängig innerhalb des Files sind.
- Einfachere Lokalisierung der Body-Container-Daten wenn auf Band gespeichert wird.
- Optimierung der Datendistribution in einem File für Speicherung oder Übertragung.

Ein Ziel der Header-Metadaten-Wiederholung ist es, die Rekonstruktion kritischer Metadaten zu unterstützen, etwa bei Applikationen, bei denen Files unterbrochen werden oder bei denen die Decoder während des Transfers weitere Daten empfangen.

Die MXF-Struktur bietet sehr viel Flexibilität bei der Positionierung von Partitionierungs-Informationen und der Einsatz von Filler-Bytes ist ebenfalls möglich, etwa um die Files für unterschiedliche Geräte und Applikationen wie Speicherung oder Übertragung zu optimieren.

MXF-Files arbeiten durchgängig mit Key-Length-Value (KLV)-Kodierung, um dadurch flexibel und ausbaufähig zu sein. KLV-Codierung ist in SMPTE 336M definiert, eine vollständige Be-

schreibung davon wurde im Juli 2000 im SMPTE-Journal (Vol. 109, No7, Engineering Report) veröffentlicht.

Bei speziellen Operational Patterns kann die erste Sequenz (Präambel) auch mit Teilen beginnen, die nicht gemäß KLV-kodiert sind. Das ist erlaubt, um Synchronisierungs-Bytes oder sogenannte »Camouflage«-Bytes hinzufügen zu können, die bei bestimmten Applikationen am Beginn des Files stehen müssen. Das wird etwa verwendet, wenn ein Broadcast-WAV-Header vor den Anfang eines MXF-Files gesetzt wird, damit dieses für ein WAV-Gerät so wirkt wie ein WAV-File, ohne dass sich dabei die Byte-Struktur ändern müsste.

Die Metadaten-Bestandteile des Headers können in zwei Gruppen unterteilt werden: strukturierende und beschreibende Metadaten. Strukturierende Metadaten sind dazu da, unterschiedliche Elemente eines Files zu verbinden und um die grundlegende Struktur eines Files festzulegen. Die beschreibenden Metadaten enthalten dagegen zusätzliche Informationen über das File, etwa den Namen des Programms oder Szenenbeschreibungen. Im SMPTE-Dictionary und in den MXF-Spezifikationen ist eine Vielzahl von Metadaten-Elementen beschrieben.

Ein »Body Container« enthält die wichtigen Informationen über ein spezielles Kompressions-Schema oder über den jeweiligen Essenz-Typ. Zweck des Body-Containers ist es, die KLV-Kodierung der Essenz zu ermöglichen und dabei die Verbindung zu einer optionalen Index-Tabelle sicherzustellen. Dadurch ist schnellerer Zugang zu einem vorhandenen Time-Offset innerhalb eines Files möglich. Der Body-Container ist so strukturiert, dass er per Multiplexing leicht mit anderen Body-Containern verbunden werden kann. Zudem sorgt er für eine Identifizierungsmöglichkeit, anhand deren ermittelt werden kann, welche Decodier-Anforderungen erfüllt werden müssen, damit der Content angesehen, abgehört oder wiedergegeben werden kann.

Der Body Container beinhaltet ein Identifikationslabel sowie eine Methode, um die Essenz in eine KLV-Struktur zu bringen. Unterschiedliche Body-Container können allerdings zu Einschränkungen führen was die Kompatibilität der Essenzdaten mit bestehenden Applikationen betrifft.. Ein MXF-File kann aus mehrere Body-Containern bestehen. Die genaue Anzahl der Body-Container und ihrer Beziehungen zueinander ist vorgegeben durch das Operational Pattern auf dem das File basiert.