3. Sony: Long-GOP MPEG-2 Mapping for MXF File Storage Applications **Jim Wilkinson** The full specification of the Sony MXF mapping for long-GOP MPEG2 has been published by SMPTE as the Registered Document Disclosure, RDD-9. This section provides a summary of	**3. Sony: Công nghệ ánh xạ Long-GOP MPEG-2 phục vụ cho các ứng dụng lưu trữ tập tin** Mô tả đầy đủ về kỹ thuật ánh xạ Sony MXF cho long-GOP MPEG2 đã được SMPTE xuất bản dưới dạng Tài Liệu Công Bố Đã Đăng Ký, RDD-9. Phần này trình bày vắn tắt một số nội dung

that document. Those parties implementing files conforming to this application specification should refer to the published version of RDD-9 that contains far more detail than can be presented in this section of this chapter.

File Structure

The file structure conforms to MXF operational pattern OP1a (SMPTE 378M).

Figure 15.7 below shows the outline of the MXF file structure. The file consists of one header partition, one or more body partition(s), one footer partition, and is completed with a random index pack.

The audio-visual essence is present only in a sequence of one or more body partitions. The system, picture, sound, and optional data items aremapped into the generic container and placed in each body partition using frame wrapping. A body partition consists of a sequence of edit units (frames) that comprise the MPEG2 GOPs as shown in Figure 15.7. The number of GOPs in a partition can be one or more up to a maximum partition duration of ten seconds.

Because of the long-GOP MPEG structure of the picture item, index tables are segmented and each segment indexes the essence of the previous partition. Thus, the first index table segment is present at the beginning of the next partition and the last index table segment is present in the footer partition. The partition pack data

trong tài liệu đó. Các bên thực thi các tập tin theo đặc tả ứng dụng này có thể tham khảo phiên bản đã được xuất bản của RDD-9 có những thông tin chi tiết hơn chương này.

structure provides a property to define the size of any index table segment in its partition; thus the presence (or not) and the offset to the essence edit unit is easily decoded.

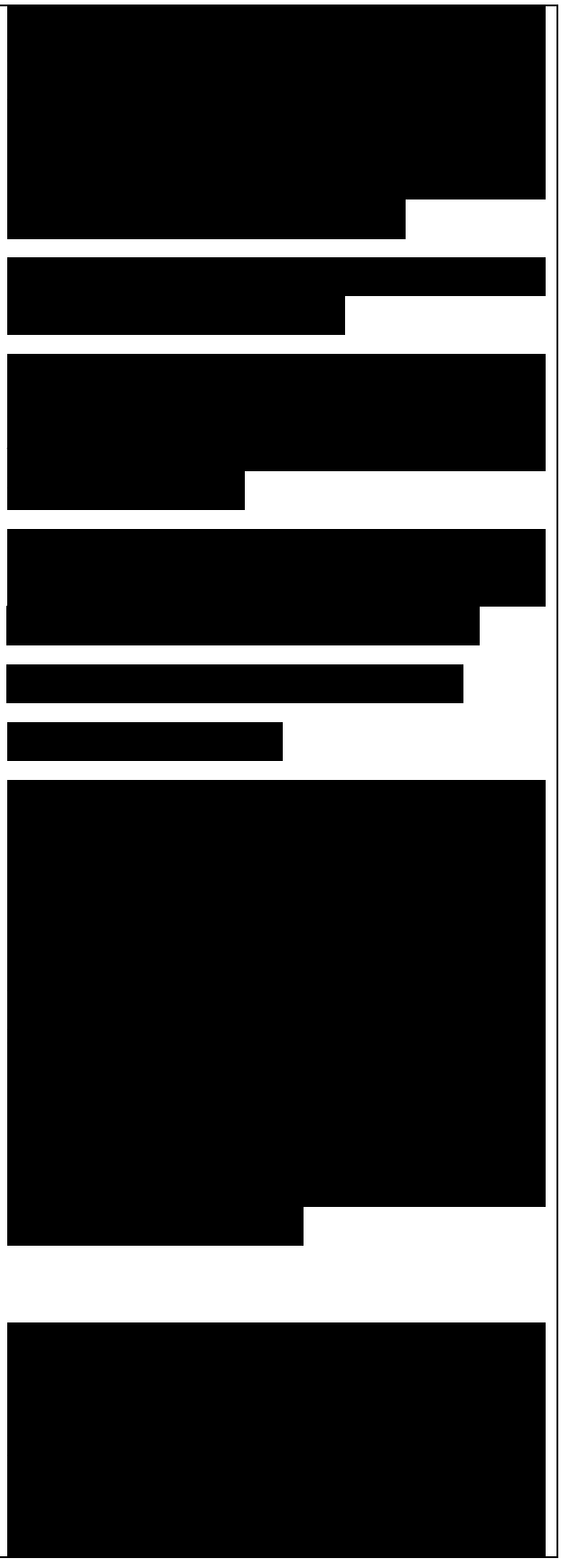
Some aspects ofthis file structure are indicatedbelow.

- It is only necessary for an encoder to build each index table segment from the previous body partitionand body partitions are limited to 10 seconds duration.
- It is easy for a decoder to play out while building indexing information as the file is received.
- It is easy to create a "partial file."

Index Table Usage

The long-GOP essence requires the use of index table segments. Index entry and delta entry arrays are both required. One index table segment isplaced at the beginning of each body partition, except for the first body partition, and it indexes the essence of the previous partition. The last index table segment is placed in the footer partition. The relationship between index tables and the essence that it indexes is illustrated in Figure 15.8.

Having each index table segment index the essence of the previous body partition permitsrealtime creation of MXF files tobe performed with minimal buffering and avoids unnecessary delay. In this mapping specification, the



picture essence is VBE (Variable Bytes per Element) and the sound essence is CBE (Constant Bytes per Element).

Random Index Pack (RIP)

The random index pack is provided to rapidly locate partitions scattered throughout an MXF file as illustrated in Figure 15.9. This fixed length pack defines the BodySID and Byte Offset to the start of each partition (i.e., the first byte of the partition pack key). This pack can be used by decoders to rapidly access index tables and to find partitions to which an index table points. The random index pack is not required but is highly recommended as a decoder performance enhancement tool. If present in the file, it followsthe footer partitionand it isalways the last KLV item in a file.

Frame-Wrapped Structure

This application specification defines frame-based wrapping only.

An arrangement of system, picture, and sound items in a frame-based wrapping is shown in Figure 15.10 illustrating theuse of 4 channels AES3 audio.

Generic Container Specification

This application specification defines each frame to contain a system item; a picture item containing a single KLV wrapped MPEG2 video elementary stream; a sound item containing, typically, 2, 4, or 8 separate AES3 audio

channel elements; and an optional data item as illustrated respectively in Figures 15.11, 15.12, 15.13, and 15.14.

Use of the KLV Fill Item

Within any MXF partition containing an essence container with this mapping specification, the KAG value defined in the partition pack hasthe value of 512 (02.00h) and the first byte of the key of the first element of each item isaligned to the KLV alignment grid of that partition. For each item in a content package, the length of the KLV fill item should be the minimum required to align to a KAG boundary.

System Item Mapping

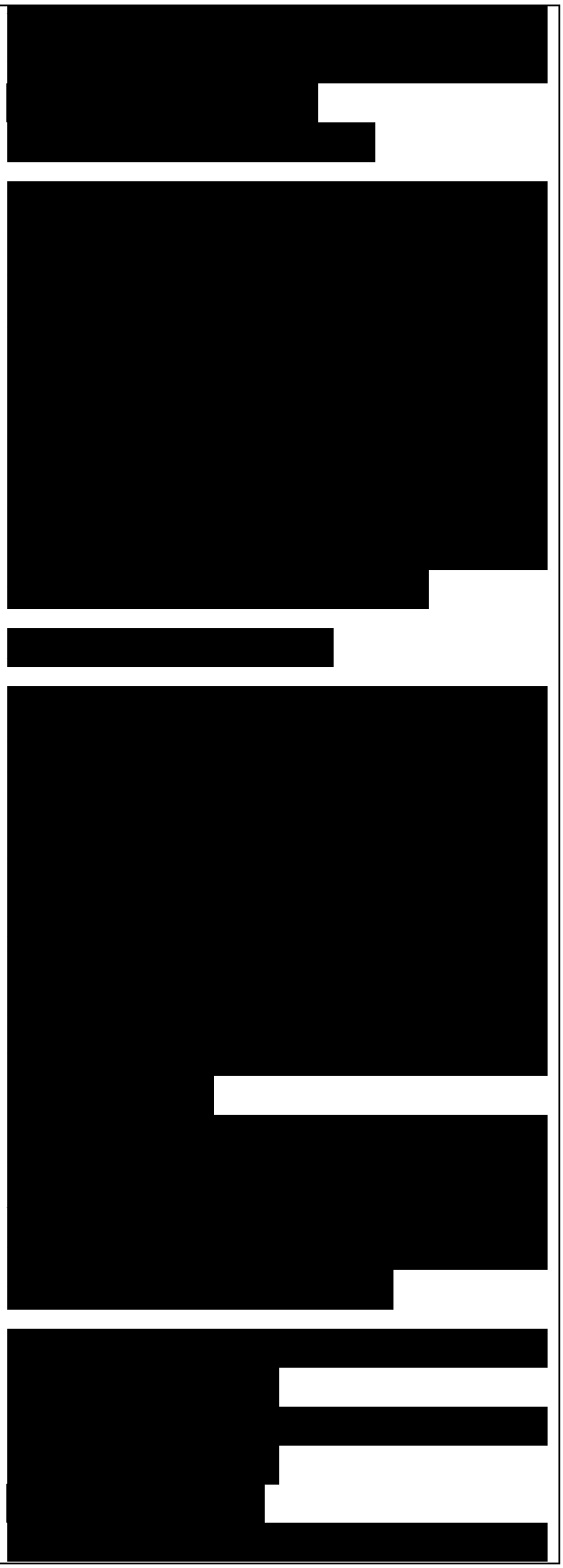
The system item in each edit unit consists of a system metadata pack and a package metadata set. The data structure is illustrated in Figure 15.11. The mapping of the system item data complies with that defined in SMPTE 385M. The structure of the package metadata set is defined in SMPTE 385M and items in the set are defined in SMPTE 331M.

Linear timecode (LTC) data is carried in the User Date/Time Stamp, and the UMID and the KLV metadata are carried in Package Metadata set as indicated in Figure 15.11.

UMID (basic or extended) Tag = 83h(SMPTE 331M)

KLV metadata Tag = 88h(SMPTE 331M)

Picture Item Mapping



The picture item contains a single frame of long-GOP MPEG-2 elementary stream data, thus the data length of each picture element in an edit unit is a variable value. The location of the picture item in each content package is illustrated in Figure 15.12.

MPEG2 Temporal Reordering

The compressed video pictures are reordered from their display order according to the MPEG specification. This reordering is applied only to the MPEG elementary stream data. All other items in the content package retain their natural temporal order.

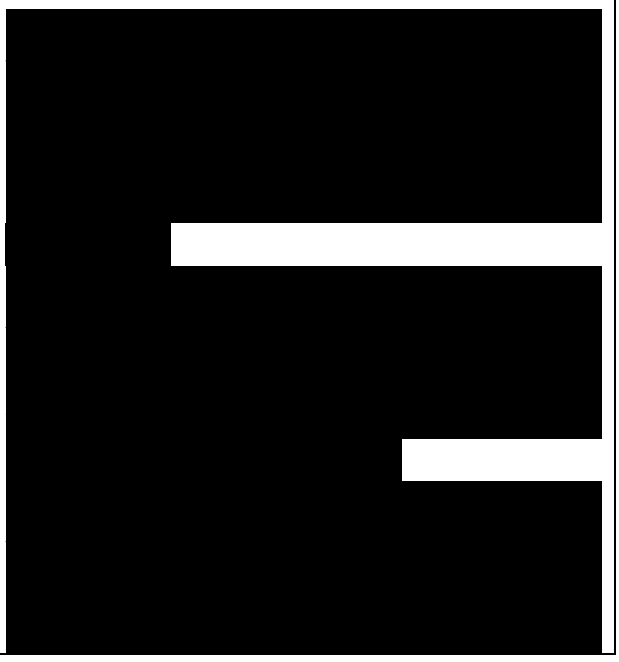
Sound Item Mapping

Each AES3 sound element complies with 382M using the AES3 mapping specification. The mapping of the sound item is as illustrated in Figure 15.13.

The number of audio data channel numbers is limited by the MXF generic container to a theoretical maximum of 128, but current implementations use 2, 4, or 8 channels and a realistic upper limit is 16 channels.

In the case of audio locked to video at 25 content packages per second or multiples, each sound element will contain the same number of samples, for example 1920.

In the case of audio locked to video at 30*1000/1001 content packages per second, the number of samples in each



sound element will contain the same number of samples, but the value for each frame will vary to maintain a correct aggregate rate. Typically, the number of samples varies according to a 5-frame sequence, 1602, 1601, 1602, 1601, and 1602. The number of samples in each content package is calculated from the Length field of the surrounding KLV packet, divided by the value of the BlockAlign property of the AES3 Audio Essence Descriptor.

In the case of audio locked to video at 24*1000/1001 content packages per second, each element will contain the same number of samples, for example 2002.

Data Item Mapping

Any data item in each edit unit may include two kinds of data element: a VBI line element and an ancillary data packet element. These are both illustrated in Figure 15.14. Both use data elements as specified in SMPTE 331M that are mapped according to SMPTE 385M for use in MXF. Use of either of the data elements in the application specification is optional.