AVC-~~596~~ 396

International Organisation for Standardization

Organisation Internationale de Normalisation

ISO-IEC/JTC1/SC29/WG11

Coded Representation of Picture and Audio Information

CCITT SGXV
Working Party XV/1
Experts Group for ATM Video Coding

ISO-IEC/JTC1/SC29/WG11
MPEG92/549
23 October, 1992

Source: A. G. (Sandy) MacInnis, IBM, USA
Geoff Morrison, BT Labs, UK

Title: The MPEG Systems Specification and Multiplex

Background

The MPEG Systems specification, which includes a multiplex, is specified in ISO DIS 11172 part/section 1 (Systems). That document should be consulted to resolve detailed points as the following is only intended to be a broad overview concentrating on salient general features.

Strictly speaking the Systems specification in DIS 11172 is applicable only to the first MPEG standard ("MPEG-1") which is commonly, though wrongly, associated with a bit rate of 1.5 Mbit/s. MPEG is currently working towards MPEG-2 and that includes activity on multiplexing and other system issues. However the considerable effort which went into MPEG-1 Systems has been recognized along with the merits of its outcome. Therefore MPEG-2 is building upon MPEG-1 Systems rather than beginning again with a clean slate. The new work is aimed at extending the features etc., while maintaining not just the basic approach but the original syntax also. The consequence of this is that the future MPEG-2 Systems specification may be seen as a superset of the current MPEG-1 Systems, with some very slight modifications to MPEG-1. The points made in this document will therefore also apply to MPEG-2 Systems.

Requirement

The main requirements of the multiplex and system coding layer are:

1. Define a coding specification for the necessary and sufficient information, in addition to the coded audio, video and other data, that enables decoding and precisely synchronized presentation. This must be possible whether or not the streams are multiplexed together.
2. Define a multiplex for 0 to many audio and video streams; these multiple streams should themselves be relatively unconstrained by the systems specification.
3. Provide a means by which coded data buffers in decoders are guaranteed not to overflow or underflow both at start up and during continuous operation, through a coding specification, semantic constraints, or both.
4. Design the coding specifications such that decoding after a random access into the data is possible after a reasonably short delay.
5. Provide absolute time identification information for the overall stream or streams that are coded for synchronized playback.
6. Require very low systems data rate overhead in typical applications.

Systems coding philosophy

MPEG Systems coding is directed towards a variety of storage and transmission means, some of which have specific constraints concerning e.g. packet or cell sizes, data rates, and higher level multiplexing or coding. Therefore it addresses all of the requirements in a wide variety of applications, with flexibility as to the source of timing, packet sizes, and whether or not the constituent data streams are in fact multiplexed at all.

The MPEG Systems specification contains a System Target Decoder (STD) as an integral part of the coding and multiplexing semantics. This is a theoretical model and the system coder and multiplexer must ensure that the multiplexed result it produces can be handled by the STD. Real demultiplexers may differ from the STD but must be able to handle multiplexed data which the STD could.

Multiplex philosophy

As MPEG concerns digital coding the multiplex is a time division multiplex. The approach is that of variable length packets, each with a header followed by data from one of the tributaries to be multiplexed. These are known as "streams" in MPEG parlance. The multiplex is byte structured, but bit serial transmission media are supported by facilities to recover byte timing.

Start codes

MPEG makes heavy use of 4 byte 'start codes' and the packets of the system layer always have one at their beginning. The first 3 bytes of start codes are always the same being two of all '0's followed by one with a '1' in the LSB (23 '0's and a '1'). The fourth byte is used to identify what comes next. The 256 codes in the fourth byte are partitioned between the System coding (where they identify the separate streams the mux is carrying) and the video algorithm (where they are used for slice addressing). The syntax of the MPEG-1 compressed video data and of MPEG systems ensure that this 24 bit pattern is not generated unintentionally by concatenation of other codewords. It can therefore be used for byte timing recovery from serial links and for mux framing recovery. Unfortunately the pattern is not unique within the total multiplexed data as it can be emulated by MPEG encoded audio and by unrestricted data streams in the multiplex. To combat this, packet length fields are present to indicate where the next start code should be found. Thus the mechanism is very similar to that used in conventional telecoms links where the length fields are unnecessary because of the known regularity of the framing bits. MPEG does not standardize the receiver strategies for detecting loss of alignment or for reacquiring it etc.

Requests have been made for MPEG-2 Systems to support explicit fixed length packets, to aid in robust parsing in error-prone environments. MPEG-2 Systems may include a flag to indicate that packets are of fixed length, probably in addition to the length field.

Packs

The MPEG mux has a two stage hierarchy of packs and packets. A pack contains a number of packets. The structure is:

SC	PACK-ID	SCR	MUX_RATE	PACKET	....	PACKET	END
3	1	5	3				

(Numbers indicate the length in bytes of the fixed length fields)

SC is the start code prefix '00 00 01' (hex notation)

Pack-id 'BA' distinguishes the pack header

SCR is System Clock Reference and is a 33 bit counter incrementing at 90 kHz. (More than 26 hours between repeats). The 33 bits are spread over 5 bytes with the extra 7 bits containing fixed values to avoid emulating '00 00 01' with certain counter values. The SCR is a time stamp on the bit stream itself, and effectively has the meaning that "when you get this, the time is the value of the data field". The SCR supports synchronizing the decoder system with the data source, regardless of which device is the timing master, while simultaneously supporting playout synchronization and buffer data flow management. The SCR is an integral part of the STD model.

Mux_rate is a 22 bit field giving the total multiplexed bit rate in units of 50 byte/s. Many receiver implementations will not need to use the data in this field. The field is 3 byte long with two '1's inserted to prevent generation of '00 00 01'. The Mux_rate can vary from pack to pack, explicitly supporting variable rate coding while simultaneously guaranteeing proper decoder buffer behavior without overflows or underflows.

Packets then follow. It is important to note that:

Each packet contains data from only one input stream.

The total number of packets per pack is not defined and can also vary from pack to pack.

The number of packets per pack for each stream is not defined and can vary from pack to pack. A pack can have no packet for a stream.

The sequence of packets is not defined and can vary from pack to pack. However, packets for any individual stream will be in chronological order, ie oldest first.

The primary constraint on the packet sizes and ordering is that the System Target Decoder (STD) model must be satisfied, i.e. it could be used as a bit stream verifier. In practice this means that essentially anything goes as long as the encoder (including the system level encoder or multiplexer) ensures that the STD buffers are not violated. This allows wide freedom in packet sizes and placement, while providing the one essential constraint that is of relevance to decoders.

Packets

The structure of packets is:

SC	STREAM_ID	LENGTH	BUF_TS	STREAM_DATA
3	1	2		

SC is '00 00 01' as always

Stream_id is 8 bits indicating the stream as follows:

```
    binary
MSB      LSB
 110x xxxx  MPEG Audio stream number xxxxx (32 available)
 1110 xxxx  MPEG Video stream number xxxx (16 available)
 1111 xxxx  Data stream xxxx (16 available)
```

A few other codes are allocated for reserved streams, stuffing streams etc.

The length field is 16 bits indicating the distance in bytes to the next SC. In addition to implementing the 'framing flywheel' as discussed above this enables a parser to jump over and discard data for streams that are present in the multiplex but are not required by the application.

The combination of the byte oriented data stream and the inclusion of packet length fields facilitates the construction of simple micro-processor based demultiplexers. The micro-processor need only look at the header and length field of each packet.

The 16 bit length field limits the size of a packet but multiple packets with the same stream_id can be placed in a pack, if required, to overcome this limit. If this is done the packets need not be contiguous.

The BUF_TS field can contain STD buffer size information. Time stamps are included in some packets to control decoder buffers and synchronize presentation and decoding times of individual streams. These time stamps refer to the same 90 kHz clock as the SCR and therefore allow timing to 11 microseconds.

STREAM_DATA is the input from that particular input to the multiplex.

Timing

The 11.11 microsecond time stamp timing accuracy is approximately half an audio sample time. In practice a decoder would use a control loop which averages errors over multiple time stamps, producing an even finer synchronization accuracy if required. MPEG audio normally assembles L and R audio into its own multiplex before the MPEG mux described here and therefore achieves the same phase accuracy as a conventional Compact Disc. It is expected that MPEG-2 multichannel audio will use the same approach.

The 11 microseconds is more than adequate for phasing video sources and synchronizing them with audio and with the data source.

In some applications such as replay from a local storage medium the audio or video D to A clock may be the master and others including the data source and multiplex clock will then be slaved to it.

In cases such as broadcasting where the transmission clock is the master at the receiver, phase locked loops will be required to recover audio and video D to A clocks. The jitter performance requirements may set some practical limits on the minimum rates at which system clock references and time stamps appear in the multiplex. Experimental results are still sparse in this area. There is a requirement that SCRs and Presentation Time Stamps (PTSs) occur with intervals no greater than 700ms; this maximum interval ensures that stable control loops can be constructed. Analysis shows that it is feasible to build timing control loops with simple filters that attenuate any timing jitter from the source by at least a few tens of decibels. More frequent SCRs support control loops with faster response times.

Flexibility

It will be appreciated from the above that the mux is extremely flexible and can be configured in many ways either permanently for a given application or dynamically within an application. Applications with a fixed number of fixed rate input streams can be handled with a fixed version of the multiplex, but need not be as dynamic variation is also easily accomplished. In essence the MPEG mux follows the same philosophy as the MPEG-1 video algorithm. The standard describes how the receiver interprets the compressed data. This places certain constraints on the encoder but does not define a unique method thus leaving scope for optimization for the particular circumstances. This flexibility allows for 'intelligent' multiplexers while the demultiplexer is 'dumb'. 'Dumb' multiplexers are also permitted.

Further, the MPEG Systems coding is applicable in cases where there is no multiplexing at all, or where the multiplex is done outside of the MPEG data stream. The only requirement is that if the various streams are to be synchronized they use a common time base for all time stamps. This facility is useful for example on storage devices with separate files or devices for each stream, or perhaps for communications where distinct data streams are necessary for such reasons as multiple transmission priorities or tariffing which is dependent on data type.

Bit rate

It will have been noted that nothing in the systems specification is bit rate sensitive and the use of a defined mechanism for stuffing bytes can match any combination of input streams to any channel whose rate is higher than their total plus the extra for the fields introduced by the system coding itself. While this overhead is not a constant, depending on pack and packet frequencies which can be selected over very wide ranges, it is readily bounded. Enough information is given above to estimate the overhead in particular circumstances.

Buffers

Any multiplex where a single channel is time shared between sources requires buffers. At the multiplexer these continue to receive data from a stream while the medium is devoted to another stream. At the demultiplexer they have the inverse role of supplying data for other streams while the medium is servicing one. In the STD model these buffers are integrated with the coding algorithm buffers, allowing optimal usage of the buffers by the encoder system. Encoders can readily be constructed which have separate coding and multiplex buffers, which may simplify the encoder. The amount of delay introduced by the multiplex depends on the multiplex structure and timing implemented by the multiplexer. An intelligent mux is able to minimize this delay by dynamically changing the sequence and proportion of time it allocates to each stream, or by e.g. simply constructing the multiplex to rotate between the streams at a fast enough rate; faster multiplex rotation comes at the expense of increased system overhead.

Since the STD model incorporates models for audio and video decoders and coded data buffers, the STD model supersedes the Video Buffer Verifier (VBV) when a multiplex is used. Multiplexing affects the timing of data delivery and so the timing assumed by the VBV is no longer valid when the data streams are multiplexed. This is not a problem for an encoder, however, as video data encoded to satisfy the VBV may be readily multiplexed with an STD buffer slightly larger than the VBV buffer. This fact is also useful for variable rate coding, which is not supported by the VBV.

One of the few constraints which MPEG specifies is that the buffering delay at the demux shall not be greater than 1 second. This should have no practical effect as normal, usable multiplexes are likely to use buffers with maximum delays on the order of 250 ms.

The BUF_TS field contains information giving the minimum STD buffer size which will be needed to decode any particular stream. This information must be included in the very first packet of a stream (i.e. when a stream is first used) and when the value changes. It may be also be included at any other time.

SCR and Timestamp frequency

It is a requirement that the SCR and timestamps occur at least every 0.7 seconds. These limits acknowledge the possible practical problems with phase locked loops etc. More frequent updates of these are of course possible by having shorter packs and packets though at the penalty of extra overhead. Since the standard only requires these time stamps to have this interval as a maximum, decoders should be constructed to perform properly with this time stamp interval. If specific applications truly require more frequent time stamps they can be included, however if a decoder is to take advantage of them it may need to be assured that shorter intervals will always be adhered to.

Capacity

Though MPEG has stream_ids for 32 audio and 16 video bit streams, all of these do not have to be used. It is not necessary to have empty packs for unused streams, nor for streams which temporarily have no data (e.g. bursty data sources such as subtitles.) The data rates supported explicitly in MPEG-1 Systems range from a 400 bits per second to approximately 1.7 gigabits per second.

MPEG-2 has announced its intention to design and standardize a Super-Mux to handle multiplexing of individual program muxes, following the MPEG-1 or similar MPEG-2 multiplex. This will permit many more than 16 individual programs, with independent time bases, in applications such as satellite and CATV broadcasting. Requirements for this development are being sought. The general philosophy is that the basic MPEG mux will combine streams which are related and the Super-Mux will combine unrelated combinations. However, this will not be an imposed rule, other than the requirement that streams within a single program multiplex use a common time base for all time stamps, meaning in practice that they are meant to be synchronized. The Super-Mux is likely to have several distinct differences from MPEG-1 systems, such as fixed packet lengths (although not necessarily of a standardized length) and very low overhead per packet.

A noteworthy feature of the MPEG mux is that it supports variable bit rate streams. The mux was expressly designed to support both the situations where the total rate of the varying streams is constant and where it varies on a pack by pack basis, even though the audio and video portions of the standard do not specify how to manage buffers and start-up delays with variable rate operation; this facility is provide by using the STD model. Though perhaps not so useful in broadcasting applications, variable rate data streams are of interest with many storage devices and also to ATM networks.

Use without multiplexing

MPEG Systems coding is equally applicable with and without an overall multiplex, although this point is perhaps not clearly made in the specification. Historically the technical solution addressed non-multiplexed environments first. If there is a multiplex performed at a higher layer that is not visible to MPEG processing timing jitter will inevitably be introduced by the fact of multiplexing. This can be handled in decoders that are designed accordingly. The problem to be solved in a decoder is very similar in nature to the problem of decoding multiplexed data which is carried over a transmission medium such as a LAN which introduces jitter, although it can be made to be less severe by limiting the multiplex rotation time. In any case practical solutions are possible.

Error correction

MPEG-1 Systems does not incorporate error correction as it is a generic multiplex scheme whereas error handling really needs to be matched to the error characteristics of the individual transmission or storage medium. Thus error correction would be applied to the final multiplexed bit/byte stream. This means that the performance of the link and corrector is determined by the most demanding constituent in the multiplex.

It is currently recognized in the MPEG-2 Systems committee that coding to support a reasonable degree of post-correction channel errors is highly desirable, and steps are being taken to make it less sensitive to errors.

In circumstances where there may be advantages in disassembling the multiplex or not multiplexing in the first place and putting parts over different links the MPEG timestamps aid subsequent resynchronization.

Encryption and scrambling

MPEG has not and does not intend to specify encryption and scrambling methods. Conventional schemes can be applied to the total multiplex or to individual streams within the multiplex. The latter offers the possibility of layered services with tiered charging as well as logically related programs with a single subscription. Under the MPEG-2 work MPEG has already accepted the need for 'service' streams within the multiplex to carry data for encryption and scrambling, e.g. encrypted scrambling seed values and information needed for decrypting the seeds etc. Stream-ids will be allocated for these purposes. Requirements input from broadcasting organizations so far indicate that scrambling should be performed on an individual stream basis.

Conclusion

The MPEG Systems coding specification has been designed with a very high degree of flexibility and adaptability in an attempt to meet the requirements of all known and future applications, without being overly optimized to any one of the widely divergent specific applications. It should be practical and desirable to use this specification in conjunction with ATM transports with or without multiplexing at the MPEG layers. MPEG-2 Systems is at this date still in the process of being defined, and if there are specific requirements which are neither met nor listed in the MPEG-2 requirements document they can be given due consideration, to the aim of reaching a single standard that is applicable across a variety of inter-operable application, communication and storage systems.