CCITT SGXV

WORKING PARTY XV/1

EXPERTS GROUP FOR ATM VIDEO CODING

DOCUMENT AVC-386

October 1992

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION

ORGANISATION INTERNATIONAL DE NORMALISATION

ISO/IEC JTC1/SC2/WG11

CODED REPRESENTATION OF PICTURE AND AUDIO INFORMATION

SO-IEC/JTC1/SC2/WG11 N
MPEG 92/

TITLE: Proposal for General One/Two-Tier MPEG Transport Syntax

SOURCE: D. Raychaudhuri / R. Siracusa / J. Zdepski (David Sarnoff Research Center)
B. Beyers / R. St. Girons (Thomson Consumer Electronics)

This document presents a specific MPEG-2 transport syntax proposal for consideration and discussion by the system group. The transport protocol described here was designed to meet the needs of various one- or two-tier delivery media alternatives including terrestrial broadcast, Cable, DBS, and ATM. The syntax presented enables various service features (such as flexible multiplexing, variable bit-rate operation, conditional access, etc.), while also providing mechanisms for robust decoding in the presence of bit-error/cell loss experienced on practical transmission media. It is noted that earlier versions of the proposed protocol syntax have been validated through both software simulation and use in operational prototypes for TV and HDTV delivery.

This specification is based on a "structured packing" framework with fixed length packets (cells) of specifiable size. The protocol consists of 3 sublayers: 1) network, 2) adaptation, 3) video service, each with a clearly defined set of functions. The network layer provides generic service multiplexing, scrambling and priority support for video delivery systems, and can be substituted with an available alternative such as ATM. The video adaptation layer is designed for efficient packing (segmentation) of variable length MPEG data into fixed length cells, as well as to provide a number of error management facilities for robust decoding. The video service level is an extension of the MPEG slice header concept, providing decoder resynchronization boundaries in the video bitstream.

The transport protocol format defines cells (or packets) of data where each cell includes a Prefix, an Adaptation Header, and a transport Data Block. The Prefix consists of four bits of control information and twelve bits for service channel identification. Service multiplexing capabilities provide support of a mix of video, audio, and data services. This prefix is followed by a payload of n*8 bytes (e.g. 48 bytes used in ATM, 128 bytes used in a specific TV broadcasting implementation, etc.). The payload consists of a 2 byte (or 1 byte) adaptation header followed by a Transport Data Block for MPEG video, redundant MPEG headers or auxiliary data. Support for redundant headers has been provided to avoid serious decoding problems caused by occasional loss of critical MPEG video headers under moderate to high bit-error/cell loss conditions. Auxiliary data service is provided for decoder mode indicators, time codes, scrambling information, etc.

Provided within this protocol are mechanisms to facilitate rapid decoder recovery after detecting the loss of one or more cells on the channel. By identifying specific bundles of information and redundantly transmitting key MPEG data, the decoder can control the region of the image affected by errors, and then apply appropriate concealment algorithms such as temporal replacement, spatial interpolation, etc. When prioritized transmission is used for graceful degradation on difficult media, the two-tier adaptation layer supports recovery of partially received video data and helps to resynchronize high- and low-priority streams.

The following specification conforms to MPEG-style format; attached is a set of figures presenting the same syntax in pictorial form.

1 Specification of the Transport System Bitstream Syntax

1.1 Network Layer

The Network Layer is defined for systems without an existing network protocol (eg. AD-HDTV). An alternate network protocol could be substituted for existing networks (eg. ATM).

packet() {		
prefix()	16	prefix
adaptation_header()	8 or 16	ah
data_block	*	data_block
}		

* The number of bytes for the Adaptation Header plus Data Block should be a multiple of 8 bytes to simplify encryption hardware implementation. The maximum size of the Adaptation Header plus Data Block is 256 bytes.

1.1.1 Prefix

The service channel identity defined in the Prefix is mapped to a particular service type (eg. audio, video, data) using information obtained in the "SCID Mapping Table" (presented in SCID channel number 1). A sample of this mapping function is described in supporting documention.

prefix() {		
priority	1	p
auxiliary_data_group_flag	1	aux
control_flag_for_scramble_key	1	cf
control_sync_for_scramble_key	1	cs
service_channel_identity	12	scid
}		

1.2 Adaptation Layer

The selection of Adaptation Header Type is done using auxiliary data groups. Two Adaptation Header Types are defined in this document. Type 1 is a header extension to type 0, and type 0 is default.

1.2.1 Adaptation Header Type 0

adaptation_header_0() {		
continunity_counter	4	cc
identity()	4	
}		

1.2.1.1 Identity

identity() {		
identification()	2	ah_id
frame_flag	1	f
priority_flag	1	p
}		

1.2.1.2 AH Identification

1.2.1.2.1 Identity - minimal definition for all SCID services

```
identification() {
    switch (id) {
        case 0:
            (reserved for future definition)
            break
        case 1:
            basic_service()                    (service specific)
            break
        default:
            other cases are system specific
}
```

1.2.1.2.2 Identity - for MPEG video services

```
identification() {
    if (No entry-point operation) {
        switch (ah_id) {
            case 0:                                          (one-tier systems)
                (reserved for future definition)
                break
            case 1:                                          (one-tier systems)
                basic_service()
                break
            case 2:                                          (one-tier systems)
                redundant_mpeg_header()
                basic_service()
                break
            case 3:                                          (one-tier systems)
                non_mpeg_header()
                basic_service()
                break
        }
    } else {
        switch (ah_id) {
            case 0:                                          (one/two-tier systems)
                (reserved for future definition)
                break
            case 1:                                          (one/two-tier systems)
                entry_pt()
                basic_service()
                break
            case 2:                                          (one/two-tier systems)
                entry_pt()
                redundant_mpeg_header()
                basic_service()
                break
            case 3:                                          (one/two-tier systems)
                entry_pt()
                non_mpeg_header()
                basic_service()
                break
        }
}
```

If the Redundant MPEG Header in AH-ID 2 or Non-MPEG Header field in AH-ID 3 extends into two or more packets (a Data Group), the Data Group must complete before changing the Adaptation Header ID. The first completely filled packet of a Data Group exceeding one packet will have its "number of bytes" field set to 0xff. Successive packets filled with the data from the same group data will have its"number of bytes" field set to 0xfe. The final packet of the Data Group will have its "number of bytes" field set to the number of bytes which following that complete the Data Group (i.e. 1...253). The "number of bytes" value of zero is reserved for future definition.

When Adaptation Header Type 1 is available, and a Data Group extends into two or more packets, the B bit is set to one in the packet beginning the Data Group, and the E bit is set to one in the packet ending the Data Group. If the Data Group fits within one packet, the B and E bits are both set to one.

basic_service()

basic_service() {		
MPEG_data*		
}		

redundant_mpeg_header

redundant_mpeg_header() {		
number_of_bytes	8	n_bytes
redundant_mpeg_data	8 * n_bytes	
}		

non_mpeg_header

non_mpeg_header() {		
number_of_bytes	8	n_bytes
non_mpeg_data	8 * n_bytes	
}		

entry_pt()

```
entry_pt() {
    if (8-bit entry-point indicated in Aux Data or AH-ID Extension bits I3,I2) {
        8b_entry_pt_data()
    } else {
        32b_entry_pt_data()
    }
}
```

* Some systems might choose to packet align (start new packets) a MPEG Start Code at the start of, for example, a Picture. These packet-aligned Picture Start Codes would be coincident with the state change of Adaptation Header Control bit F.

8b_entry_pt_data

8b_entry_pt_data() {		
8b_entry_pt	8	entry_pt
}		

32b_entry_pt_data

32b_entry_pt_data() {		
8b_entry_pt	8	entry_pt
bit_entry_pt	3	bit_entry_pt
frame_type	2	frame_type
frame_number	5	frame_number
unassigned (reserved for future definition)	1	
slice_mblk_id	13	s_mblk_id
}		

1.2.2 Adaptation Header Type 1

adaptation_header_1() {		
adaptation_extension_size	2	a1, a0
set_boundary	1	s
tiers	1	t
beginning_of_data_group	1	b
ending_of_data_group	1	e
i3_i2	2	i3,i2
}		

1.2.3 Auxiliary Data Groups

Any number of Auxiliary Groups may appear in a packet designated as carrying auxiliary data groups (refer to prefix aux bit), but the sum of all Auxiliary Groups must be less than or equal to the size of one Data Block. Each Auxiliary Group defined within a data block has its associated CFF bit set to one. Unused bits in the data block are padded with zero.

```
auxiliary_data_group() {
    while (cff == 1) {
        auxiliary_group()
    }
}
```

auxiliary_group

auxiliary_group() {		
auxiliary_data_prefix()		
auxiliary_data_block	N	N defined in afs
}		

auxiliary_data_prefix

auxiliary_data_prefix() {		
modifible_flag	1	mf
current_field_flag	1	cff
aux_field_id	6	afid
aux_field_size	8	afs
}		

2 Semantics for the Transport System Bitstream Syntax

2.1 Network Layer

data_block -- The data_block is the group of bytes that immediately follow the adaptation_header. The format of the data_block is defined by the adaptation_header. The data_block plus adaptation header is a multiple of 8 bytes in length, and has a maximum length of 256 bytes -- limited by the size of the entry-point pointers.

priority -- Network transmission link priority; one bit flag set to 1 for HP or 0 for SP , or set to 1 for a single tier network.

auxiliary_data_group_flag -- A one-bit flag set to one when the contents of the Data Block is a Auxiliary Data Group. (See section 2.3 for Auxiliary Data Group syntax.)

control_flag_for_scramble_key and **control_sync_for_scramble_key** -- Two bit integer indicating a dual-key packet scrambling operation, as indicated below:

control_flag_ for_scramble_key	control_sync_ for_scramble_key	meaning
0	0	Use the "Even" key to decode this packet
0	1	Use the "Odd" key to decode this packet
1	0	Packet is unscrambled; If it is an Aux packet containing a scramble key, this key definition is for the "Even" key.
1	1	Packet is unscrambled; If it is an Aux packet containing a scramble key, this key definition is for the "Odd" key.

service_channel_identity -- Identifies the service number of this packet, and is a twelve bit integer defined below:

service_channel_identity	meaning
0	reserved for null packet
1	reserved for "SCID Mapping Table"
other	Implementation specific identifiers for applications such as video channels, audio channels , data channels, etc.
4095	reserved for future definition

A null packet is used to when no data is available to transmit from available SCID channels. A "SCID Mapping Table" (available on SCID channel 1), will map the type of packet (video, audio, data, etc) to each applicable SCID channel number.

2.2 Adaptation Layer

2.2.1 Adaptation Header (AH Type 0)

continunity_counter -- The continunity counter (CC) is a four bit integer that sequences by one from 0 through 15. A separate CC exists for each SCID and AH-ID where defined. The CC is used to detect a packet loss due to transmission or other errors. The CC counter is not reset to 0 to offer better protection between small groups of AH-ID data blocks. If the packet has the Set-Boundary bit set to one, then the CC of this packet may not be related to the CC of the preceding packet.

frame_flag -- A one bit flag that changes state when a frame boundary is crossed. The first packet of the new frame can only be a Basic Service packet, AH-ID = 1, to aid error recovery.
Some systems might choose to packet align (start new packets) a MPEG Start Code at the start of, for example, a Picture. These packet-aligned Picture Start Codes would be coincident with the state change of frame_flag.

priority_flag -- In two-tier systems, this one bit flag defines this packet as HP (0) or SP (1); single tier systems set this bit to 0. The adaptation header priority bit can be used for reconstructing two tier application data.

2.2.2 Adaptation Header (AH Type 1)

adaptation_extension_size -- This two bit integer defines the number of bytes in the AH-Type 1 Header Extension as follows:

adaptation_extension_size	meaning*
0	One byte extension (defined in this document)
1	Two byte extension
2	Three byte extension
3	Four byte extension

*The two, three, and four byte extension syntax is reserved for future definition.

set_boundary -- Set to one, this one bit flag indicates this packet is the beginning of a set of information and can be used for applications requiring splicing or editing. For MPEG video applications, this the Set-Boundary would be set to one in the first packet starting each new GOP.

tiers -- Set to one, this one bit flag indicates a one-tier application, set to zero indicates a two tier application.

beginning_of_data_group -- If a Data Group extends into two or more packets (eg. Redundant MPEG Headers), the beginning_of_data_group bit (B) is set to one in the packet beginning the Data Group.

ending_of_data_group -- If a Data Group extends into two or more packets (eg. Redundant MPEG Headers), the ending_of_data_group bit (E)is set to one in the packet ending the Data Group. If the Data Group fits within one packet, the B bit and the E bit are both set to one.

i3_i2 -- This is a two bit integer extension to i1 and i0 in the AH Type 0 ID field, and has the following definition:

i3_i2	meaning
0	No entry-point prefix
1	8-bit entry-point prefix format
2	32-bit entry-point prefix format
3	Reserved for future definition

2.3 Auxiliary Data Group Syntax

auxiliary_data_block -- Integer number of data bytes defined by application specific needs, with size identified by field "afs" and identity determined by field "afid". (eg. Time Code, scramble ID/KEY, etc);

modifible_flag -- If the contents of the "auxiliary_data_block" can be modified, this one bit flag is set to one. (eg. A time code that should (or should not) be modified by a remux.)

current_field_flag -- This one bit flag is set to one if the auxiliary group is present, and set to zero if the auxiliary group is not present.

aux_field_id -- This six bit integer provides the identity of the auxiliary_data_block. Initial definitions are shown in the table below:

aux_field_id	meaning
0	Time Code (see table below)
1	Scramble ID/KEY *(format syntax to be defined)*
2	Branding/Copyright information *(format syntax to be defined)*
3	Tape Recorder Playback-Mode Signaling *(format syntax to be defined)*
4	MPEG Transport Mode (identifies: Number of tiers, Entry-point mode, Adaptation header format, Set-Boundary, etc). *(format syntax to be defined)*
5...63	Reserved for future use.

The aux_field_id = 0 (time code) is defined in the five byte format shown below:

```
time_code {
    zero_fill_most_significant_bits        3
    time_code                              37          time_code
}
```

aux_field_size -- This eight bit integer determines the size (in integer bytes) of the auxiliary_data_block.

2.4 Entry-point data

8b_entry_pt --The Entry-Pointer points to first byte of an entry-point in the Data Block. The entry-point for one-tier systems and for the HP tier of two-tier systems is called a Record Header. The first byte after the "Entry-Point Data" field is byte number zero. If the pointer value is 0xff, then this Transport Cell has no entry-point.

bit_entry_pt -- This 3 bit integer is an bit entry into the byte entry-point. A value of 0 indicates the most-significant bit of the byte entry-point. The bit entry-point is only used in the 32b_entry_pt_data.

MPEG data -- Standardized MPEG bit-stream.

frame_type -- Two bit flag defining the frame type of the entry-point, as indicated below:

frame_type	meaning
00	First Packet (packet aligned) I-Frame
01	I-Frame other than above case 00
10	P-Frame
11	B-Frame

frame_number -- Five bit integer counter that sequences from 0 through 31 by one for each new video frame.

unassigned -- One bit reserved for future definition.

slice_mblk_id -- If packet is HP priority, this thirteen bit integer contains the Slice number of the entry-point, if packet is SP priority, this thirteen bit integer contains the Macroblock number of the entry-point.

macroblocks_per_slice -- The number of macroblocks contained in the current Slice.

Note: frame_type, frame_number, and slice_mblk_id are used to align the two HP and SP streams after a loss due to transmission or other error, when packet buffers are used at the receiver.

2.5 Adaptation Headers Data Block

number_of_bytes -- Eight bit integer to indicate the number of bytes used in this packet for the redundant_headers field. This number of bytes would be discarded if the receiver did not need to make use of the redundant data.

redundant_mpeg_data -- A redundant copy of in-stream MPEG Sequence Header and/or MPEG GOP Header and/or MPEG Picture header(s).

non-MPEG data -- System implementation defined bit-stream.

2.6 Record Header

If entry-points are used in one-tier systems, then within the data block containing MPEG data exists an entry-point of the following Record Header format:

record_header_one_tier() {	
MPEG Slice Header (without 0x000001 prefix)	8
number_of_macroblocks_per_slice()	
}	

If entry-points are used in two-tier systems, then within the data block containing MPEG data exists an entry-point of the following Record Header format in the HP tier (the SP tier has no record header):

record_header_two_tier() {	
priority_breakpoint	7
MPEG Slice Header (without 0x000001 prefix)	8
number_of_macroblocks_per_slice()	9 or 17
}	

number_of_macroblocks_per_slice()

number_of_macroblocks_per_slice() {		
length_of_size	1	lofs
macroblocks_per_slice	S*	
}		

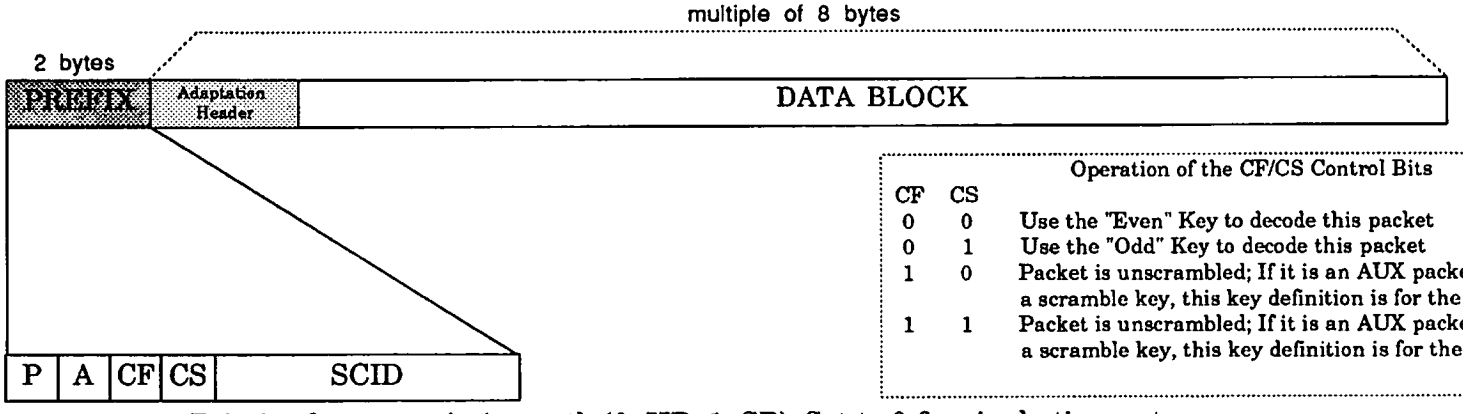
* if (lofs)==0, S=8 else if if (lofs)==1, S=16

priority_breakpoint -- Provided by a priority processor, this 7 bit integer indicates the computed threshold of what data is placed on HP and on SP for a two-tier system. This breakpoint remains constant over a MPEG slice.

macroblocks_per_slice -- The number of macroblocks contained in the current Slice.

NETWORK LAYER

Fixed packet size = Prefix + Adaptation Header + Data Block
(Scrambling applies to both the Adaptation Header and Data Block)



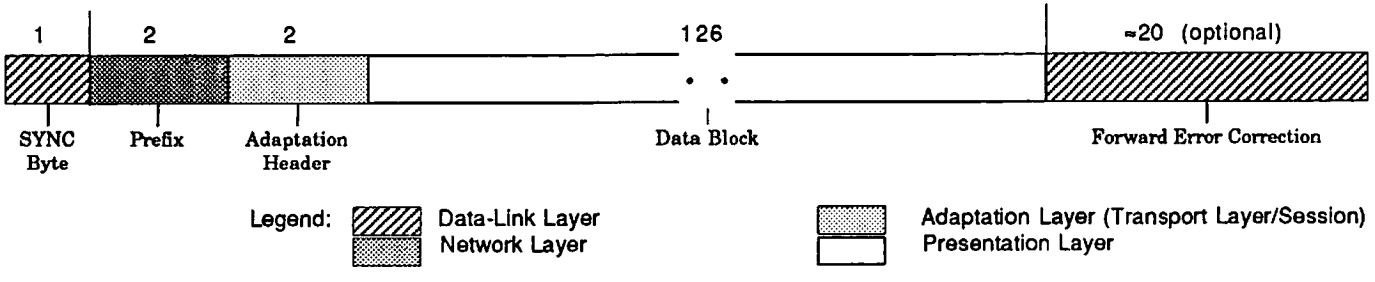
Operation of the CF/CS Control Bits

CF	CS	
0	0	Use the "Even" Key to decode this packet
0	1	Use the "Odd" Key to decode this packet
1	0	Packet is unscrambled; If it is an AUX packet containing a scramble key, this key definition is for the "Even" Key
1	1	Packet is unscrambled; If it is an AUX packet containing a scramble key, this key definition is for the "Odd" Key

P (1 bit) : Priority for transmission path (0=HP, 1=SP); Set to 0 for single tier systems
A (1 bit) : Auxiliary Data Block Format (0=False, 1=True,)
CF, CS (2 bits) : Scramble control flags; see accompanying table
SCID (12 bits) : Service Channel Identity: -
SCID 0 = Reserved for Null packet, SCID 1 = Reserved for "SCID Mapping Table"
SCID 4095 = Reserved for future definition
Other SCID assignments reserved for specific applications, (eg. Audio, Video, Data, etc)

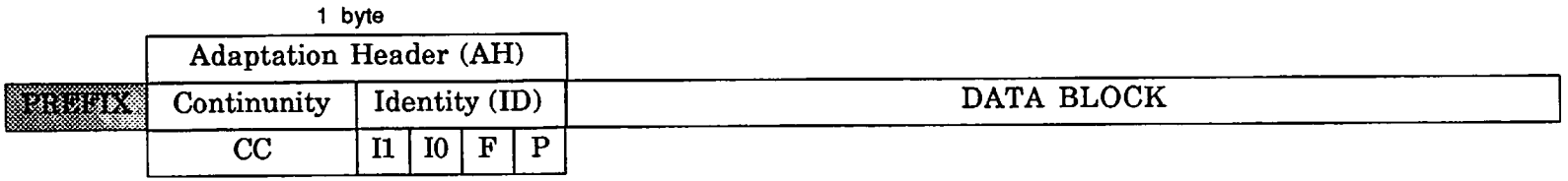
This network protocol is defined for systems without an existing network protocol (eg. AD-HDTV).
An alternate network protocol could be substituted for existing networks (eg. ATM).

Logical Structure of a Sample Transport Cell



rjs 10/22/92

ADAPTATION LAYER (AH TYPE 0)



CONTINUNITY (Continunity of Data Blocks) 4 bits

CC: The Continunity Count sequences by one from 0 through 15. A separate CC exists for each SCID and AH-ID where defined. (See definition of Adaptation Header Identity)
The CC is used to detect a packet loss due to transmission or other errors.
The CC counter is not reset to 0 to offer protection for small groups of AH-ID data blocks.
If a Set-Boundary has been indicated in Auxiliary Data,
then the CC of this packet may not be related to the CC of the preceding packet.

IDENTITY (Identity of Data Block: AH-ID) 4 bits

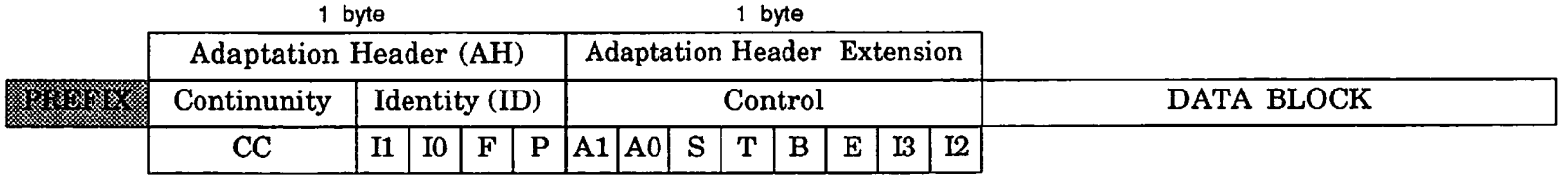
I1, I0: (See definition of Adaptation Header Identity)

F: Framing flag toggles state at presentation frame boundary (eg. MPEG video Picture Header)

P: Application Priority flag (for two-tier video, 0=HP, 1=SP; set to zero for single-tier video)

Selection of AH Type formats is done using Auxiliary Data Groups. Two Adaptation Header Types are defined in this document. Type 1 is a header extension to type 0, and type 0 is default. Adaptation Layer Type 0 requires a supporting Auxiliary Data Group with Aux Field ID 4 to define: Number of tiers, Entry-point mode, and Set-Boundary flag for application splicing or editing. In the absence of Auxiliary data, one-tier, no entry-point data, and no Set-Boundary indications are assumed.

ADAPTATION LAYER (AH TYPE 1)



CONTINUNITY (Continunity of Data Blocks) 4 bits

CC: The Continunity Count sequences by one from 0 through 15. A separate CC exists for each SCID and AH-ID where defined. (See definition of Adaptation Header Identity)
The CC is used to detect a packet loss due to transmission or other errors.
The CC counter is not reset to 0 to offer protection for small groups of AH-ID data blocks.
If this packet has the AH Extension Set-Boundary flag set to one,
then the CC of this packet may not be related to the CC of the preceding packet.

IDENTITY (Identity of Data Block: AH-ID) 4 bits

I1, I0: (See definition of Adaptation Header Identity)

F: Framing flag toggles state at presentation frame boundary (eg. MPEG video Picture Header)

P: Application Priority flag (for two-tier video, 0=HP, 1=SP; set to zero for single-tier video)

CONTROL(This Adaptation Layer Header Extension is available when AH Type 1 indicated in Auxiliary Data) 8 bits

A1, A0: Adaptation Extension Size defines the number of bytes in the AH TYPE 1 header extention --
0,0 = One byte header extension (as defined here); 0,1 = Two byte; 1,0 = Three byte; 1,1 = Four byte extension
(The 2, 3, and 4 byte extension syntax is reserved for future definition.)

S: Set-Boundary flag for application splicing/editing (eg. video: set to one in first packet of GOP)

T: Number of tiers used for application (0 = two-tier, 1 = one-tier)

B: Set to one indicates the packet beginning a data group (eg. starting multiple packets of Redundant Data).

E: Set to one indiactes the packet ending a data group.

I3, I2: A two bit extension to the AH-ID field of the basic Adaptation Header. (For MPEG video applications, these bits are set as follows: 0,0 = no entry-point prefix; 0,1 = 8 bit entry-point prefix; 1,0 = 32 bit entry-point prefix; 1,1 = reserved for future definition.)

Adaptation Header - Minimal Header for all Services

I1	I0	ID	Tiers~	CC Counter	Contents of Data Block
0	0	0	-	-	(reserved for future definition)
0	1	1	-	#	Basic Service

Other states of the bit combinations for I1 and I0 are application dependent.

Adaptation Header - Identity Definitions for MPEG Video Services

If Auxilary data indicates NO Entry-Point operation, the Adaptation Header has the following definition:

I1	I0	ID	Tiers~	CC Counter	Contents of Data Block
0	0	0	-	-	(reserved for future definition)
0	1	1	1	#	Basic Service
1	0	2	1	#	Redundant MPEG Headers & Basic Service
1	1	3	1	#	Non-MPEG Headers & Basic Service

If Auxilary data indicates Entry-Point operation, the Adaptation Header has the following definition:

I1	I0	ID	Tiers~	CC Counter	Contents of Data Block
0	0	0	-	-	(reserved for future definition)
0	1	1	1,2	#	Entry-Point Prefix* & Basic Service
1	0	2	1,2	#	Entry-Point Prefix* & Redundant MPEG Headers & Basic Service
1	1	3	1,2	#	Entry-Point Prefix* & Non-MPEG Headers & Basic Service

~ Tiers indicates applicability to one or two tier systems. The number of tiers is defined in Auxilary Data.

* The size of Entry-Point data is either 8 bits for one-tier systems or 32 bits for two-tier systems.

Only one CC counter is needed to support MPEG Video Adaptation Header ID 1, 2, and 3.

Note: If the Redundant MPEG Header in AH-ID 2 or Non-MPEG Header field in AH-ID 3 extends into two or more packets (a data group), the data group must complete before changing the Adaptation Header ID. The first completely filled packet of a data group exceeding one packet will have its "number of bytes" field set to 0xff. Successive packets filled with the data from the same group data will have its "number of bytes" field set to 0xfe. The final packet of the data group will have its "number of bytes" field set to the number of bytes which following that complete the data group (ie. 1...253). The "number of bytes" value of zero is reserved for future definition.

ADAPTATION HEADER / DATA BLOCK FORMATS

Auxiliary Data Group Format

Auxiliary Group					Additional Auxiliary Groups
Auxiliary Data Prefix				Auxiliary Data Block	
MF	CFF	AFID	AFS	Data	
1 bit	1 bit	6 bits	8 bits	N bytes	

Any number of Auxiliary Groups may appear in this packet, but the sum of all Auxiliary Group bytes must be ≤ size of one Data Block.

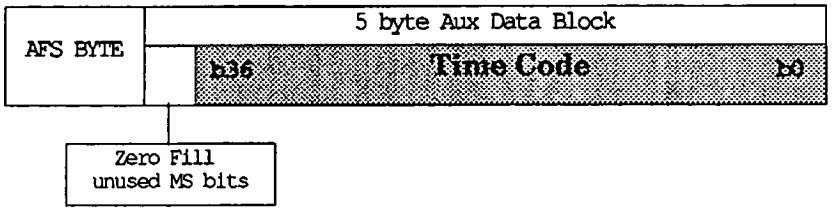
MF Modifiable Flag : If Auxiliary Data field can be modified, this bit is set to 1. (eg. retimestamping at REMUX)

CFF Current Field Flag : If an Aux Group is defined in this field, this bit is set to 1. (Pad unused bits in packet with zero), If no Auxiliary Group is defined in this field, this bit is set to 0.

AFID Aux Field ID : Identity of the Auxiliary Data. (eg. 0 = Time Code, 1 = Scramble ID/KEY,
2 = Branding/Copyright, 3 = Tape Recorder Playback-Mode Signaling (to be defined))
4 = MPEG Transport Mode (Identifies: Number of tiers, Entry-point mode, Adaptation Header format type, Set-Boundary flag for application splicing/editing, etc)

AFS Aux Field Size : Size in bytes of the Auxiliary Data Block.

Data Aux Data : Auxiliary Data for this Auxiliary Group.
A data element contained in an Aux Data Block is LS bit justified.
Example: 37 bit Time Code: Bit 36 ... bit 0, bit 36 is MS, bit 0 is LS.



ADAPTATION HEADER / DATA BLOCK FORMATS (con't)

NO ENTRY POINT DATA

Adaptation Header ID = 1

Packet aligned MPEG Sequence, GOP, or Picture Header.
(system implementation dependent)

Basic Service

Adaptation Header ID = 2

Number of bytes used in this packet for Redundant Data;
value 0 is reserved for future definition.

Redundant MPEG Headers & Basic Service

Examples of "Redundant MPEG Data" are:
Sequence or GOP or Picture Headers.

Byte alignment zero-fill stuffing,
if not already byte aligned.

Adaptation Header ID = 3

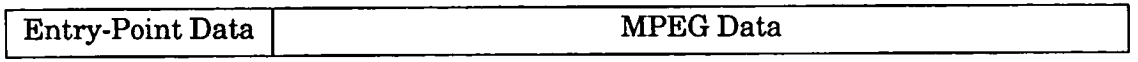
Non-MPEG Headers & Basic Service

An example of "Non-MPEG Data" is Presentation Time

ENTRY POINT DATA

Adaptation Header ID = 1

Basic Service with Entry-Point Prefix



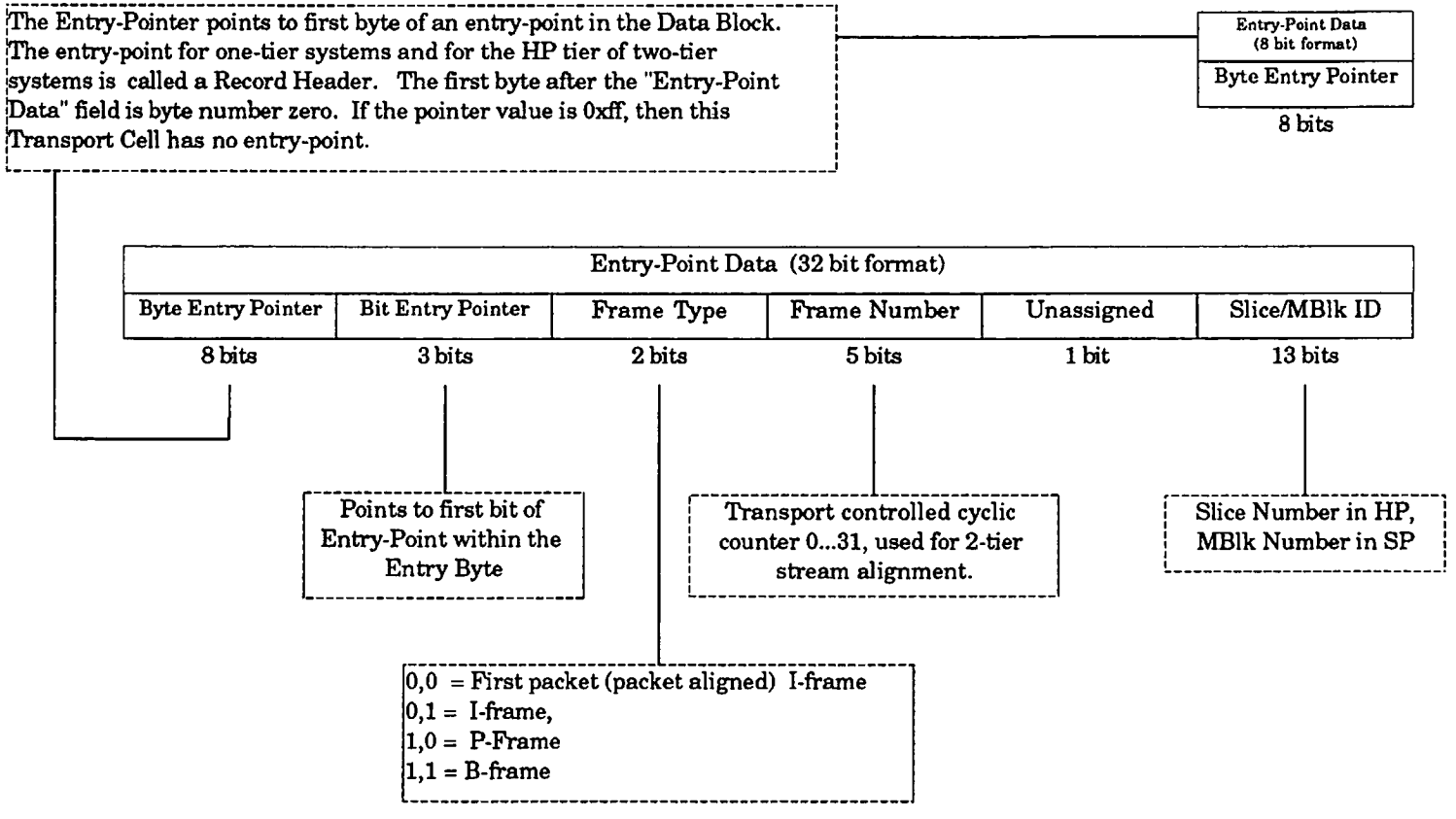
Adaptation Header ID = 2

Entry-Point Prefix with Redundant MPEG Headers & Basic Service

Adaptation Header ID = 3

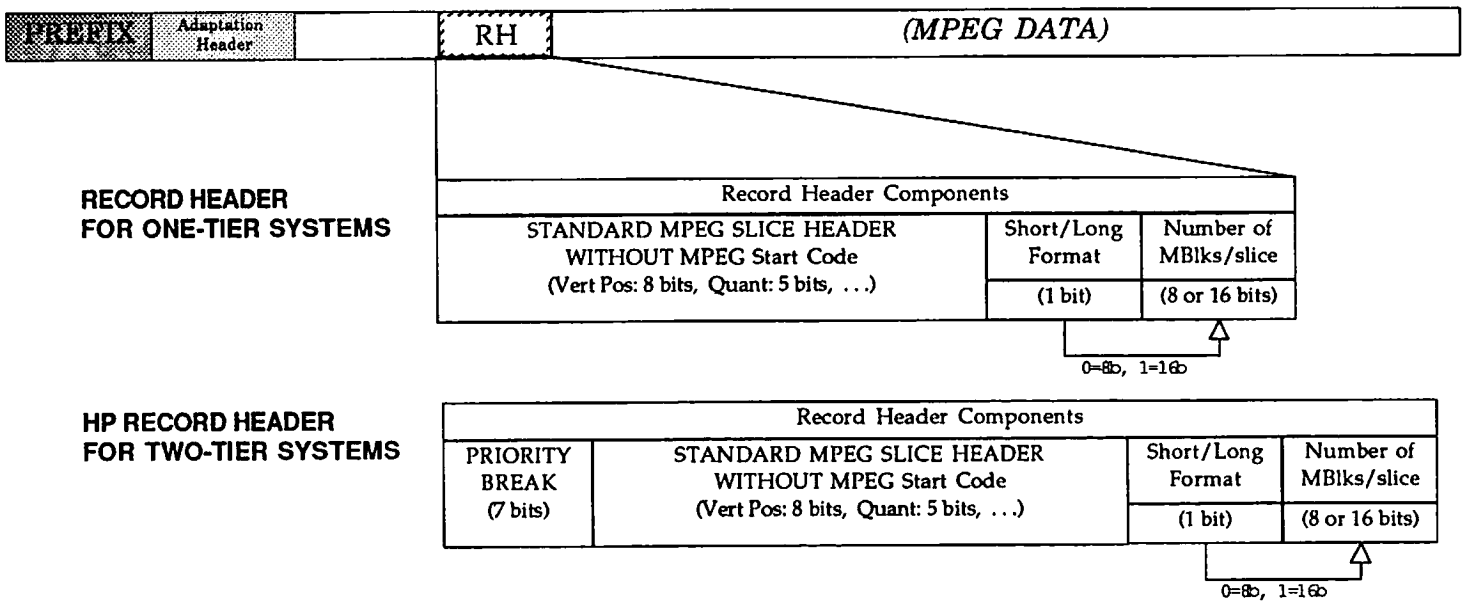
Entry-Point Prefix with Non-MPEG Headers & Basic Service

MPEG VIDEO SERVICE
Entry-Point Data Formats



MPEG VIDEO SERVICE LAYER

Record Header Format used with Entry-Pointer on One-tier systems
and on the HP tier of Two-tier systems



When Entry-Point Data refers to slice number zero, it is not pointing to a Record Header; instead it is identifing the entry-point as either the video encoder supplied Sequence Header, Group of Picture Header, or Picture Header.

While more than one RH (and other Headers) may appear within a Data Block, only the first is used as an entry-point.

The Priority Break data is supplied by a priority processor.