6.1.2.6.2 Type 4 Logical Upstreams[29]

Type 4 Logical Upstreams are identified by UCD Type 35 and may additionally have UCD Type 29. The presence of UCD Type 29 allows use of these logical upstream channels by DOCSIS 2.0 CMs. If the UCD Type 29 is not present, the channel is restricted to use by DOCSIS 3.0 CMs only.

This channel type allows the operator to define burst profiles for five data IUCs (5, 6, 9, 10 and 11) for use by DOCSIS 3.0 CMs. The CMTS is free to select, using proprietary criteria, the most appropriate data IUC for each data burst for 3.0 CMs operating in Multiple Transmit Channel Mode. If UCD Type 29 is present, the operator should configure IUCs 9 and 10 to be appropriate for short and long data bursts for DOCSIS 2.0 CMs.

Additionally, Type 4SR logical upstreams allow the use of Selectable Active Codes Mode 2 and Code Hopping Mode 2 (see Section 6.4.3 and [DOCSIS PHY]).

6.1.3 Future Use

A number of fields are defined as being "for future use" or Reserved in the various MAC frames described in this document. These fields will not be interpreted or used in any manner by this version (3.0) of the MAC protocol.

The CMTS MUST transmit all Reserved or "for future use" fields as zero. The CM MUST silently ignore all Reserved or "for future use" fields.

The CM MUST transmit all Reserved or "for future use" fields as zero. The CMTS MUST silently ignore all Reserved or "for future use" fields.

6.2 MAC Frame Formats

6.2.1 Generic MAC Frame Format

A MAC frame is the basic unit of transfer between MAC sublayers at the CMTS and the cable modem. The same basic structure is used in both the upstream and downstream directions. MAC frames are variable in length. The term "frame" is used in this context to indicate a unit of information that is passed between MAC sublayer peers. This is not to be confused with the term "framing" that indicates some fixed timing relationship.

There are three distinct regions to consider, as shown in Figure 6–1. Preceding the MAC frame is either PMD sublayer overhead (upstream) or an MPEG transmission convergence header (downstream). The first part of the MAC frame is the MAC Header. The MAC Header uniquely identifies the contents of the MAC frame. Following the header is the optional Data PDU region. The format of the Data PDU and whether it is even present is described in the MAC Header.

[29] Section modified per MULPIv3.0-N-08.0629-3 on 5/1/08 by KN.

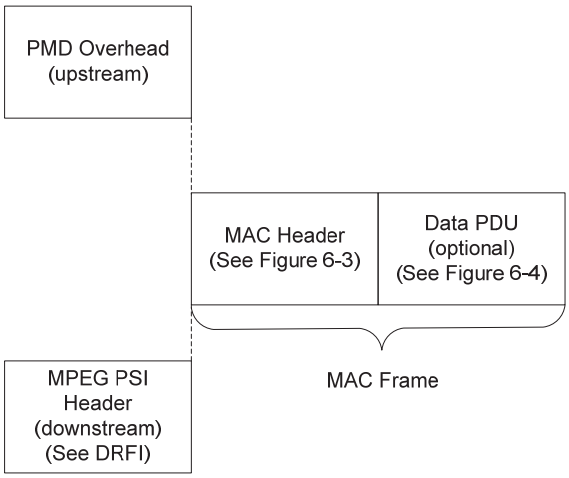


Figure 6–1 - Generic MAC Frame Format

6.2.1.1 PMD Overhead

In the upstream direction, the PHY layer indicates the start of the MAC frame to the MAC sublayer. From the MAC sublayer's perspective, it only needs to know the total amount of overhead so it can account for it in the Bandwidth Allocation process. More information on this may be found in the PMD Sublayer section of [DOCSIS DRFI].

The FEC overhead is spread throughout the MAC frame and is assumed to be transparent to the MAC data stream. The MAC sublayer does need to be able to account for the overhead when doing Bandwidth Allocation. More information on this may be found in the Upstream Bandwidth Allocation section of this document (refer to Section 7.2.1).

6.2.1.2 MAC Frame Transport

The transport of MAC frames by the PMD sublayer for upstream channels is shown in Figure 6–2.

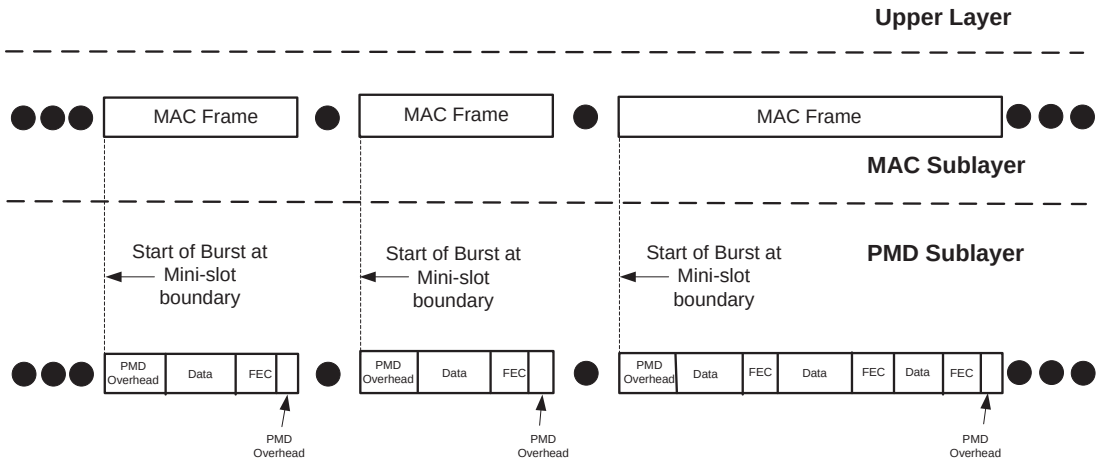


Figure 6–2 - Upstream MAC/PMD Convergence

The layering of MAC frames over MPEG in the downstream channel is described in [DOCSIS DRFI].

Note that the CMTS PHY ensures that, for a given channel, the CMTS MAC receives upstream MAC frames in the same order the CM mapped the MAC frames onto mini-slots. That is to say that if MAC frame X begins in mini-slot n and MAC frame Y begins in mini-slot n+m, then the CMTS MAC will receive X before it receives Y. This is true even when, as is possible with S-CDMA, mini-slots n and n+m are actually simultaneously transmitted within the PHY layer.

6.2.1.3 Ordering of Bits and Octets[30]

Within an octet, the least-significant bit is the first transmitted on the wire. This follows the convention used by Ethernet and [ISO/IEC 8802-3]. This is often called bit-little-endian order.

This applies to the upstream channel only. For the downstream channel, the MPEG transmission convergence sublayer presents an octet-wide interface to the MAC, so the MAC sublayer does not define the bit order.

Within the MAC layer, when numeric quantities are represented by more than one octet (i.e., 16-bit and 32-bit values), the octet containing the most-significant bits is the first transmitted on the wire. This is sometimes called byte-big-endian order.

This specification uses the following textual conventions:

- When tables describe bit fields within an octet, the most significant bits are topmost in the table. For example, in Table 6–2, FC_TYPE occupies the two most-significant bits and EHDR_ON occupies the least-significant bit.
- When figures depict bit positions within an octet, the most significant bits are leftmost in the figure. For example, see the locations of the FC_TYPE and EHDR_ON bits in Figure 6–3.
- When bit-strings are presented in text, the most significant bit is leftmost in the string.
- Unless explicitly indicated otherwise, when bits are enumerated in a bit-field, the least significant bit of the bit-field is bit # 0. The exceptions are certain fields that utilize the BITS Encoding convention.
- When message formats are presented in figures, the message octets are shown in the order in which they are transmitted on the wire, beginning with the field in the upper left and reading left-to-right, one row at a time. For example, in Figure 6–13, the FC byte is transmitted first, followed by the MAC PARM and LEN fields. As mentioned above, the LEN field is transmitted with most-significant octet first, and each octet is transmitted with least-significant bit first.

6.2.1.3.1 Representing Negative Numbers

Signed integer values MUST be transmitted and received by the CM and CMTS in two's complement format.

6.2.1.3.2 Type-Length-Value Fields[31]

Many MAC messages incorporate Type-Length-Value (TLV) fields. Except for the cases of Primary Service Flow selection and MIC calculation among the TLVs encoded in a CM Configuration File, TLV fields are unordered lists of TLV-tuples. Some TLVs are nested (see Annex C). The CM or CMTS MUST set all TLV Length fields, except for EH_LEN (see Section 6.2.5), to be greater than zero. Unless otherwise specified, Type is one byte and Length is one byte.

Using this encoding, new parameters may be added which some devices cannot interpret. A CM or CMTS which does not recognize a parameter type MUST skip over this parameter and not treat the event as an error condition.

[30] Section modified per MULPIv3.0-N-07.0544-2 on 10/31/07 by KN.
[31] Modified per MULPIv3.0-N-08.0687-3 on 1/6/09 by JS.

6.2.1.4 MAC Header Format

The CM or CMTS MUST use the MAC Header format as shown in Figure 6–3.

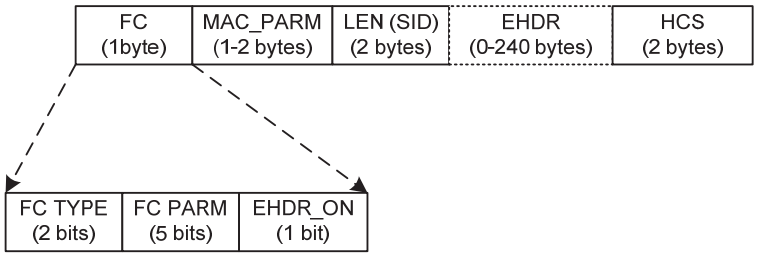


Figure 6–3 - MAC Header Format

The CM MUST comply with Table 6–1 for all MAC Headers. The CMTS MUST comply with Table 6–1 for all MAC Headers. The Frame Control (FC) field is the first byte and uniquely identifies the rest of the contents within the MAC Header. The FC field is followed by 3 bytes of MAC control; an optional Extended Header field (EHDR); plus a Header Check Sequence (HCS) to ensure the integrity of the MAC Header.

Table 6–1 - Generic MAC Header Format

MAC Header Field	Usage	Size
FC	Frame Control: Identifies type of MAC Header	8 bits
MAC_PARM	Parameter field whose use is dependent on FC: if EHDR_ON=1; used for EHDR field length (ELEN) else if for concatenated frames (see Table 6–10) used for MAC frame count else (for Requests only) indicates the number of mini-slots requested	8 bits for all headers except for the Queue-Depth based request header in which this field is 16 bits.
LEN (SID)	The length of the MAC frame. The length is defined to be the sum of the number of bytes in the extended header (if present) and the number of bytes following the HCS field. (For a REQ Header, this field is the Service ID instead).	16 bits
EHDR	Extended MAC Header (where present; variable size).	0-240 bytes
HCS	MAC Header Check Sequence	2 bytes
	Length of a MAC Header	6 bytes + EHDR

FC Field: The FC field is broken down into the FC_TYPE sub-field, FC_PARM sub-field and an EHDR_ON indication flag. The CM MUST comply with the FC field in Table 6–2. The CMTS MUST comply with the FC field in Table 6–2 for the FC field.

Table 6–2 - FC Field Format

FC Field	Usage	Size
FC_TYPE	MAC Frame Control Type field: 00: Packet PDU MAC Header 01: ATM PDU MAC Header 10: Isolation Packet PDU MAC Header 11: MAC Specific Header	2 bits
FC_PARM	Parameter bits, use dependent on FC_TYPE.	5 bits
EHDR_ON	When = 1, indicates that EHDR field is present. [Length of EHDR (ELEN) determined by MAC_PARM field]	1 bit

The FC_TYPE sub-field includes the two MSBs of the FC field. These bits MUST always be interpreted by CMs and CMTSs in the same manner to indicate one of four possible MAC frame formats. These types include: MAC Header with Packet PDU; MAC Header with ATM cells; MAC Header with packet PDU Isolation from Pre-3.0 DOCSIS cable modems; or a MAC Header used for specific MAC control purposes. These types are spelled out in more detail in the remainder of this section.

The five bits following the FC_TYPE sub-field is the FC_PARM sub-field. The use of these bits is dependent on the type of MAC Header. The LSB of the FC field is the EHDR_ON indicator. If this bit is set, then an Extended Header (EHDR) is present. The EHDR provides a mechanism to allow the MAC Header to be extensible in an inter-operable manner.

Note that the Transmission Convergence Sublayer stuff-byte pattern is defined to be a value of 0xFF, which precludes the use of FC byte values which have FC_TYPE = ‘11’ and FC_PARM = ‘11111’.

MAC_PARM: The MAC_PARM field of the MAC Header serves several purposes depending on the FC field. If the EHDR_ON indicator is set, then the MAC_PARM field MUST be used by the CM and CMTS as the Extended Header length (ELEN). The EHDR field may vary from 0 to 240 bytes. If this is a concatenation MAC Header, then the MAC_PARM field represents the number of MAC frames (CNT) in the concatenation (see Section 6.2.5.6). If this is a Request MAC Header (REQ), (see Section 6.2.4.3), then the MAC_PARM field represents the amount of bandwidth being requested. In all other cases, the MAC_PARM field is reserved for future use.

LEN (SID): The third field has two possible uses. In most cases, it indicates the length (LEN) of this MAC frame. In one special case, the Request MAC Header, it is used to indicate the cable modem’s Service ID since no PDU follows the MAC Header.

EHDR: The Extended Header (EHDR) field provides extensions to the MAC frame format. It is used to implement data link security as well as frame fragmentation, and can be extended to add support for additional functions in future releases.

HCS: The HCS field is a 16-bit CRC that ensures the integrity of the MAC Header, even in a collision environment. The CM or CMTS MUST include the entire MAC Header, starting with the FC field and including any EHDR field that may be present for HCS field coverage. The HCS is calculated using CRC-CCITT (x16 + x12 + x5 + 1) as defined in [ITU-T X.25].

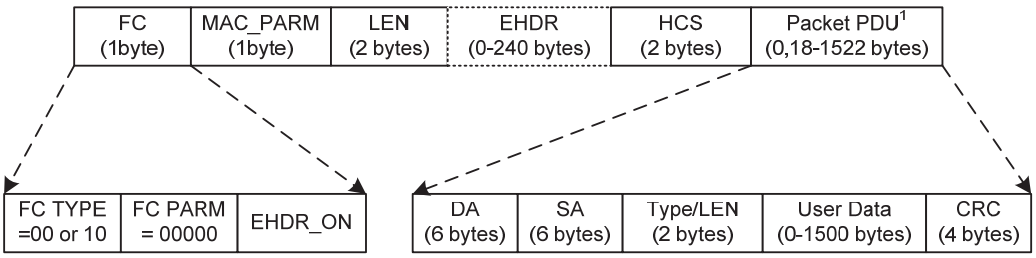
6.2.1.5 Data PDU

The MAC Header may be followed by a Data PDU. The type and format of the Data PDU is defined in the Frame Control field of the MAC Header. The FC field explicitly defines a Packet Data PDU, an ATM Data PDU, an Isolation Packet Data PDU, and a MAC-Specific Frame. All CMs MUST use the length in the MAC Header to skip over any reserved data.

6.2.2 Packet-Based MAC Frames

6.2.2.1 Packet PDU and Isolation Packet PDU

The CM or CMTS MAC sublayer MUST support both, a variable-length Ethernet/[ISO/IEC 8802-3]-type Packet Data PDU MAC Frame and a variable-length Ethernet/[ISO/IEC 8802-3]-type Isolation Packet Data PDU MAC Frame. The Isolation Packet Data PDU MAC Frame is used to prevent certain downstream packets from being received and forwarded by Pre-3.0 DOCSIS cable modems, as described in Section 9.2.2.2.1.[32] Both the Packet PDU and the Isolation Packet PDU can be used to send packets of any type (unicast, multicast, and broadcast). With the exception of packets which have been subject to Payload Header suppression, the Packet PDU MUST be passed across the network in its entirety, including its original CRC. In the case where Payload Header Suppression has been applied to the Packet PDU, all bytes except those suppressed MUST be passed across the network by the CM and CMTS, and the CRC covers only those bytes actually transmitted (refer to Section 6.2.5.4.1). A unique Packet MAC Header is appended to the beginning. The CM MUST comply with Figure 6–4 and Table 6–3 for Packet PDUs and Isolation Packet PDUs. The CMTS MUST comply with Figure 6–4 and Table 6–3 for Packet PDUs and Isolation Packet PDUs.



[1] Packet PDU length is limited to 1518 bytes in the absence of VLAN tagging. When PHS is applied, it is possible for the Packet PDU length to be less than 18 bytes.

Figure 6–4 - Packet PDU or Isolation Packet PDU MAC Frame Format

Table 6–3 - Packet PDU or Isolation Packet PDU MAC Frame Format

Field	Usage	Size
FC	FC_TYPE = 00; Packet PDU MAC Header FC_TYPE = 10; Isolation Packet PDU MAC Header FC_PARM[4:0] = 00000; other values reserved for future use and ignored EHDR_ON = 0 if there is no extended header, 1 if there is an EHDR	8 bits
MAC_PARM	MAC_PARM = x; MUST be set to zero if there is no EHDR; Otherwise set to length of EHDR	8 bits
LEN	LEN = n+x; length of Packet PDU in bytes + length of EHDR	16 bits
EHDR	Extended MAC Header, if present	x (0-240) bytes

[32] Modified per MULPIv3.0-N-06.0371-4 by KN on 1/26/07.

Field	Usage	Size
HCS	MAC Header Check Sequence	16 bits
Packet Data Packet PDU:	DA - 48 bit Destination Address SA - 48 bit Source Address Type/Len - 16 bit Ethernet Type or [ISO/IEC 8802-3] Length Field User Data (variable length, 0-1500 bytes) CRC - 32-bit CRC over packet PDU (as defined in Ethernet/[ISO/IEC 8802-3])	n bytes
	Length of Packet PDU or Isolation Packet PDU MAC frame	6 + x + n bytes

Under certain circumstances it may be necessary to transmit a packet PDU MAC frame without an actual PDU. This is done so that the extended header can be used to carry certain information about the state of the service flow, e.g., a 5-byte Downstream Service Extended Header containing the current Sequence Number for a particular DSID (also known as a "null packet"), or a Service Flow Extended Header containing the number of active grants for a UGS-AD service flow. This could also happen as a result of PHS in the upstream direction (see PHS Section 6.2.5.4.1).[33]

Such a frame will have the length field in the MAC header set to the length of the extended header and will have no packet data, and therefore no CRC.

6.2.3 ATM Cell MAC Frames

The FC_TYPE 0x01 is reserved for future definition of ATM Cell MAC Frames. This FC_TYPE field in the MAC Header indicates that an ATM PDU is present. This PDU MUST be silently discarded by CMs and CMTSs compliant with this version (3.0) of the specification. Compliant version 3.0 CM and CMTS implementations MUST use the length field to skip over the ATM PDU.

6.2.4 MAC-Specific Headers

There are several MAC Headers which are used for very specific functions. These functions include support for downstream timing and upstream ranging/power adjustment, requesting bandwidth, fragmentation and concatenating multiple MAC frames.

Table 6–4 describes FC_PARM usage within the MAC Specific Header.

Table 6–4 - MAC-Specific Headers and Frames

FC_PARM	Header/Frame Type
00000	Timing Header
00001	MAC Management Header
00010	Request Frame
00011	Fragmentation Header
00100	Queue Depth-based Request Frame
11100	Concatenation Header

[33] This paragraph and next modified per MULPIv3.0-N-08.0694-4 on 1/6/09 by JS.

6.2.4.1 Timing Header

A specific MAC Header is identified to help support the timing and adjustments required. In the downstream, this MAC Header MUST be used by the CMTS to transport the Global Timing Reference to which all cable modems synchronize. In the upstream, this MAC Header MUST be used by the CM as part of the Ranging message needed for a cable modem's timing and power adjustments. The Timing MAC Header is followed by a Packet Data PDU. The CM MUST comply with Figure 6–5 and Table 6–5 for Timing Headers. The CMTS MUST comply with Figure 6–5 and Table 6–5 for Timing Headers.

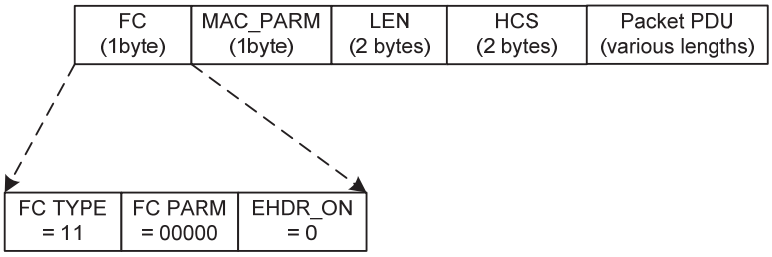


Figure 6–5 - Timing MAC Header

Table 6–5 - Timing MAC Header Format

Field	Usage	Size
FC	FC_TYPE = 11; MAC Specific Header FC_PARM[4:0] = 00000; Timing MAC Header EHDR_ON = 0; Extended header prohibited for SYNC and RNG-REQ	8 bits
MAC_PARM	Reserved for future use	8 bits
LEN	LEN = n; Length of Packet PDU in bytes	16 bits
EHDR	Extended MAC Header not present	0 bytes
HCS	MAC Header Check Sequence	2 bytes
Packet Data	MAC Management Message: SYNC message (downstream only) RNG-REQ (upstream only)	n bytes
	Length of Timing Message MAC frame	6 + n bytes

6.2.4.2 MAC Management Header

A specific MAC Header is identified to help support the MAC management messages required. This MAC Header MUST be used by CMs and CMTSs to transport all MAC management messages (refer to Section 6.4). The CM MUST comply with Figure 6–6 and Table 6–6 for MAC Management Headers. The CMTS MUST comply with Figure 6–6 and Table 6–6 for MAC Management Headers.

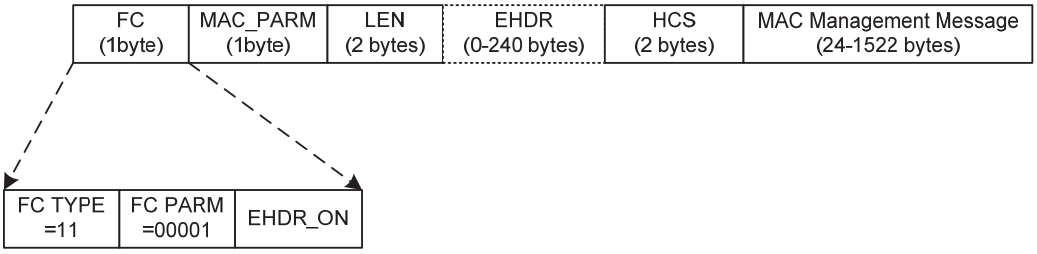


Figure 6–6 - Management MAC Header

Table 6–6 - MAC Management Format

Field	Usage	Size
FC	FC_TYPE = 11; MAC Specific Header FC_PARM[4:0] = 00001; Management MAC Header EHDR_ON = 0 if there is no extended header, 1 if there is an EHDR	8 bits
MAC_PARM	MAC_PARM = x; MUST be set to zero if there is no EHDR; Otherwise set to length of EHDR	8 bits
LEN	LEN = n+x; length of MAC management message + length of EHDR in bytes	16 bits
EHDR	Extended MAC Header, if present	x (0-240) bytes
HCS	MAC Header Check Sequence	16 bits
Packet Data	MAC management message	n bytes
	Length of Packet MAC frame	6 + x + n bytes

6.2.4.3 Request Frame[34]

The Request Frame is the basic mechanism that a cable modem uses to request bandwidth. As such, it is only applicable in the upstream. The CM MUST NOT include any Data PDUs following the Request Frame. The CM MUST comply with Figure 6–7 and Table 6–7 for Request Frames.

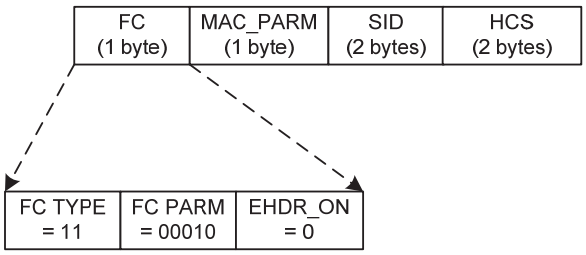


Figure 6–7 - Request Frame Format

Table 6–7 - Request Frame (REQ) Format

Field	Usage	Size
FC	FC_TYPE = 11; MAC-Specific Header FC_PARM[4:0] = 00010; MAC Header only; no data PDU following EHDR_ON = 0; No EHDR allowed	8 bits
MAC_PARM	REQ, total number of mini-slots requested	8 bits
SID	Service ID used for requesting bandwidth. For valid SID ranges, see Section 7.2.1.2.	16 bits
EHDR	Extended MAC Header not allowed	0 bytes
HCS	MAC Header Check Sequence	2 bytes
	Length of a REQ MAC Header	6 bytes

Because the Request Frame does not have a Data PDU following it, the LEN field is not needed. The CM MUST replace the LEN field with a SID. The SID uniquely identifies a particular Service Flow within a given CM.

The CM MUST specify the bandwidth request, REQ, in mini-slots. The CM MUST indicate the current total amount of bandwidth requested for this service queue including appropriate allowance for the PHY overhead in the MAC_PARM field.

The Request Frame is for Pre-3.0 DOCSIS support and MUST NOT be used by CMs operating in Multiple Transmit Channel Mode. CMs operating in Multiple Transmit Channel Mode MUST use queue depth based requests as defined in Section 6.2.4.5.

6.2.4.4 Fragmentation Header

The Fragmentation MAC Header provides the basic mechanism to split a larger MAC PDU into smaller pieces that are transmitted individually and then re-assembled at the CMTS. As such, Fragmentation is only applicable in the upstream. The CM MUST comply with Figure 6–8 and Table 6–8 for Fragmentation MAC Headers.

[34] Revised per MULPIv3.0-N-07.0427-3 by ab on 4/30/07.

A compliant CM MUST support fragmentation. A compliant CMTS MUST support fragmentation. To decrease the burden on the CMTS and to reduce unnecessary overhead, fragmentation headers MUST NOT be used by a CM on unfragmented frames.

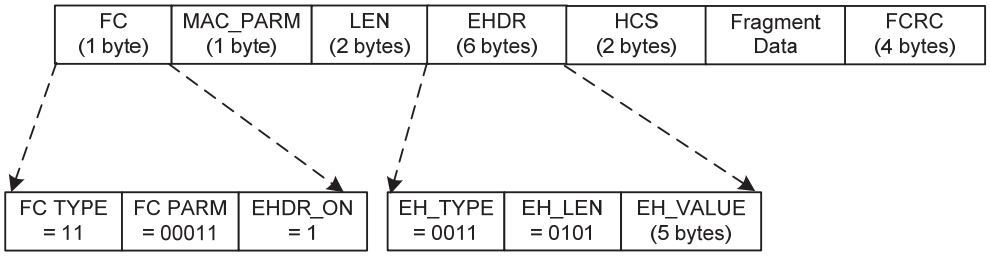


Figure 6–8 - Fragmentation MAC Header Format

Table 6–8 - Fragmentation MAC Frame (FRAG) Format

Field	Usage	Size
FC	FC_TYPE = 11; MAC-Specific Header FC_PARM [4:0] = 00011; Fragmentation MAC Header EHDR_ON = 1; Fragmentation EHDR follows	8 bits
MAC_PARM	ELEN = 6 bytes; length of Fragmentation EHDR	8 bits
LEN	LEN = length of fragment payload + EHDR length + FCRC length	16 bits
EHDR	Refer to Section 6.2.5.3	6 bytes
HCS	MAC Header Check Sequence	2 bytes
Fragment Data	Fragment payload; portion of total MAC PDU being sent	n bytes
FCRC	CRC - 32-bit CRC over Fragment Data payload (as defined in Ethernet/[ISO/IEC 8802-3])	4 bytes
	Length of a MAC Fragment Frame	16 + n bytes

The Fragmentation MAC Frame is for Pre-3.0 DOCSIS support and MUST NOT be used by CMs operating in Multiple Transmit Channel Mode.

6.2.4.5 Queue-depth Based Request Frame

The Queue-depth Based Request Frame is the mechanism that a cable modem uses to request bandwidth in terms of bytes, not including or assuming any physical layer overhead (preamble, FEC, physical layer padding, guard time), which is used when the CM is in Multiple Transmit Channel Mode. This is unlike the Request Frame in which requests are made in units of mini-slots that include physical layer overhead. The Queue-depth Based Request Frame is only applicable in the upstream. The CM MUST NOT include any Data PDUs following the Queue-depth Based Request Frame. The CM MUST comply with Figure 6–9 and Table 6–9 for Queue-depth Based Request Frames.

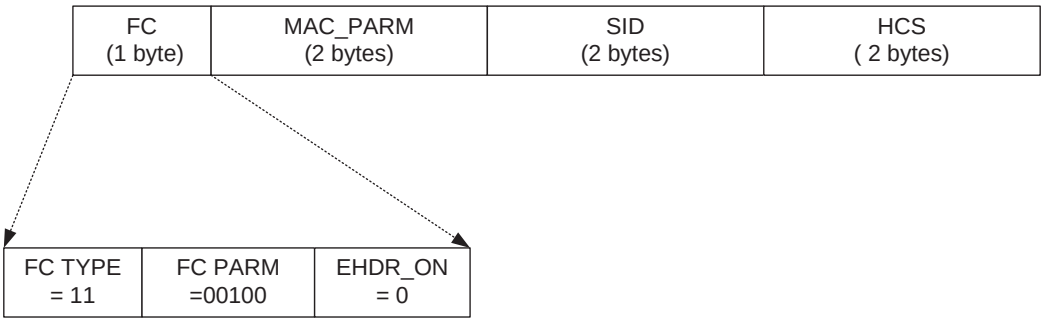


Figure 6–9 - Queue-depth Based Request Frame Format

Table 6–9 - Queue-depth Based Request Frame Format

Field	Usage	Size
FC	FC_TYPE = 11; MAC-Specific Header FC_PARM[4:0] = 00100; MAC Header only; no data PDU following EHDR_ON = 0; No EHDR allowed	1 byte
MAC_PARM	Total number of bytes requested in units of N bytes, where N is a parameter of the service flow for which this request is being made	2 bytes
SID	Service ID (0...0x3DFF)	2 bytes
EHDR	Extended MAC Header not allowed	0 bytes
HCS	MAC Header Check Sequence	2 bytes
	Length of a Queue-depth Based REQ MAC Header	7 bytes

Because the Queue-depth Based Request Frame does not have a Data PDU following it, the LEN field is not needed. The CM MUST replace the LEN field with a SID. The SID uniquely identifies a particular Service Flow within a given CM.

Note: The Queue-depth Based Request Frame is one byte longer than the Pre-3.0 DOCSIS Request Frame.

Queue-depth Based Request Frames MUST NOT be used by CMs operating with Multiple Transmit Channel Mode disabled.

6.2.4.6 Concatenation Header

A Specific MAC Header is defined to allow multiple MAC frames to be concatenated.

The CM MUST comply with Figure 6–10 and Table 6–10 for Concatenation MAC Headers.

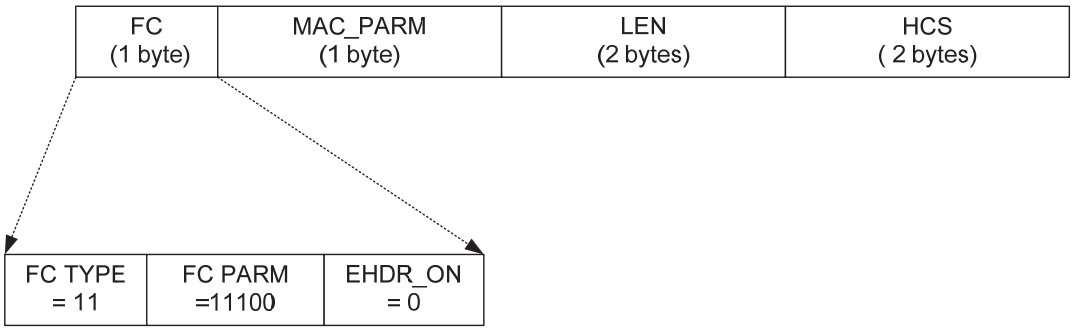


Figure 6–10 - Concatenation MAC Header Format [35]

Table 6–10 - Concatenated MAC Frame Format

Field	Usage	Size
FC	FC_TYPE = 11; MAC Specific Header FC_PARM[4:0] = 11100; Concatenation MAC Header EHDR_ON = 0; No EHDR with Concatenation Header	8 bits
MAC_PARM	CNT, number of MAC frames in this concatenation CNT = 0 indicates unspecified number of MAC frames	8 bits
LEN	LEN = x +... + y; length of all following MAC frames in bytes	16 bits
EHDR	Extended MAC Header MUST NOT be used	0 bytes
HCS	MAC Header Check Sequence	2 bytes
MAC frame 1	First MAC frame: MAC Header plus OPTIONAL data PDU	x bytes
MAC frame n	Last MAC frame: MAC Header plus OPTIONAL data PDU	y bytes
	Length of Concatenated MAC frame	6 + LEN bytes

The MAC_PARM field in the Concatenation MAC header provides a count of MAC frames as opposed to EHDR length or REQ amount as used in other MAC headers. If the field is non-zero, then it indicates the total count of MAC Frames (CNT) in this concatenation burst.

The Concatenation Frame is for Pre-3.0 DOCSIS support and MUST NOT be used by CMs operating in Multiple Transmit Channel Mode.

6.2.5 Extended MAC Headers

Every MAC Header, except the Timing, Concatenation MAC Header, Request Frame, and Queue-depth Based Request Frame, has the capability of defining an Extended Header field (EHDR). The CM or CMTS MUST indicate the presence of an EHDR field by the EHDR_ON flag in the FC field being set. Whenever this bit is set, then the CM or CMTS MUST use the MAC_PARM field as the EHDR length (ELEN). The minimum defined EHDR is 1 byte. The maximum EHDR length is 240 bytes.

A compliant CMTS and CM MUST support extended headers.

The CM MUST comply with Figure 6–11 and Table 6–11 for MAC Headers with an Extended Header. The CMTS MUST comply with Figure 6–11 and Table 6–11 for MAC Headers with an Extended Header.

[35] Figure revised per MULPIv3.0-N-06.0313-5 by GO on 11/20/06.

Note: The CM MUST NOT use Extended Headers in a Concatenation MAC Header, but may be included as part of the MAC Headers within the concatenation.

The CM MUST NOT use Extended Headers in Request Frames or Queue-depth Based Request Frames. The CM and CMTS MUST NOT use Extended Headers in Timing MAC Headers.

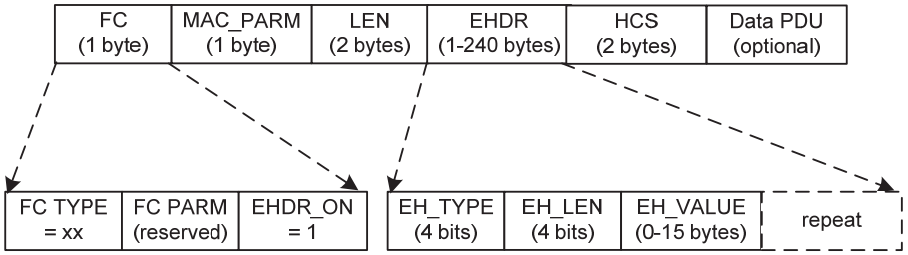


Figure 6–11 - Extended MAC Format

Table 6–11 - Example Extended Header Format

Field	Usage	Size
FC	FC_TYPE = XX; Applies to all MAC Headers FC_PARM[4:0] = XXXXX; dependent on FC_TYPE EHDR_ON = 1; EHDR present this example	8 bits
MAC_PARM	ELEN = x; length of EHDR in bytes	8 bits
LEN	LEN = x + y; length of EHDR plus optional data PDU in bytes	16 bits
EHDR	Extended MAC Header present in this example	x bytes
HCS	MAC Header Check Sequence	2 bytes
PDU	OPTIONAL data PDU	y bytes
	Length of MAC frame with EHDR	6 + x + y bytes

Since the EHDR increases the length of the MAC frame, the CM or CMTS MUST increase the value of the LEN field to include both the length of the Data PDU and the length of the EHDR.

The EHDR field consists of one or more EH elements. The size of each EH element is variable. The CM or CMTS MUST set the first byte of the EH element to contain a type and a length field. Every CM MUST use this length to skip over any unknown EH elements. The CM MUST comply with Table 6–12 for EH elements. The CMTS MUST comply with Table 6–12 for EH elements.

Table 6–12 - EH Element Format

EH Element Fields	Usage	Size
EH_TYPE	EH element Type Field	4 bits
EH_LEN	Length of EH_VALUE	4 bits
EH_VALUE	EH element data	0-15 bytes

The CM MUST support the types of EH element defined in Table 6–13. The CMTS MUST support the types of EH element defined in Table 6–13. The CM MUST comply with Table 6–13 for Extended Header Types. The CMTS MUST comply with Table 6–13 for Extended Header Types. Reserved and extended types are undefined at this point and MUST be ignored by CMs and CMTSs.

The first ten EH element types are intended for one-way transfer between the cable modem and the CMTS. The next five EH element types are for end-to-end usage within a MAC-sublayer domain. Thus, the information attached to EHDR elements 10-14 on the upstream MUST also be left attached by the CMTS when the information is forwarded within a MAC-sublayer domain. The final EH element type is an escape mechanism that allows for more types and longer values, and MUST be used by CMs and CMTSs as shown in Table 6–13.

Table 6–13 - Extended Header Types[36]

EH_TYPE	EH_LEN	EH_VALUE
0	0	Null configuration setting; may be used to pad the extended header. The EH_LEN is zero, but the configuration setting may be repeated.
1	3	Request: mini-slots requested (1 byte); SID (2 bytes) [CM→CMTS]
2	2	Acknowledgment requested; SID (2 bytes) [CM→CMTS]
3 (= BP_UP)	4	Upstream Privacy EH Element [DOCSIS SECv3.0]
	5	Upstream Privacy with Fragmentation[1] EH Element (See [DOCSIS SECv3.0] and Section 7.2.5.2)
4 (= BP_DOWN)	4	Downstream Privacy EH Element[DOCSIS SECv3.0]
5	1	Service Flow EH Element; Payload Header Suppression Header Downstream
6	1	Service Flow EH Element; Payload Header Suppression Header Upstream
	2	Service Flow EH Element; Payload Header Suppression Header Upstream (1 byte), Unsolicited Grant Synchronization Header (1 byte)
7 (= BP_UP2)	3	Upstream Privacy EH version 2 Element with no piggyback request
8	varies	Downstream Service EH Element
9	5	DOCSIS Path Verify EH Element
10 - 14		Reserved [CM <-> CM]
15	XX	Extended EH Element: EHX_TYPE (1 byte), EHX_LEN (1 byte), EH_VALUE (length determined by EHX_LEN)

[1] An Upstream Privacy with Fragmentation EH Element only occurs within a Fragmentation MAC-Specific Header. (Refer to Section 6.2.5.4)

6.2.5.1 Piggyback Requests[37]

Several Extended Headers can be used to request bandwidth for subsequent transmissions. These requests are generically referred to as "piggyback requests". They are extremely valuable for performance because they are not subject to contention as Request Frames generally are (refer to Section 7.2.2).

Requests for additional bandwidth can be included in Request, Upstream Privacy, and Upstream Privacy with Fragmentation Extended Header elements, as well as in Segment Headers.

[36] Table modified per MULPIv3.0-N-07.0487-2 on 7/11/07 and MULPIv3.0-N-0493-4 on 7/13/07 by KN.
[37] Section modified per MULPIv3.0-N-07.0487-2 on 7/11/07 by KN.

6.2.5.2 Request Extended Header[38]

The Request Extended Header (EH_TYPE=1) is used to piggyback requests on packets that do not have the Baseline Privacy extended headers. In that case, when operating with Multiple Transmit Channel Mode disabled, the CM MUST use either the Request Extended Header with EH_LEN=3 or the BP_UP Extended Header to send piggyback requests. When the CM is operating with Multiple Transmit Channel Mode enabled and segment headers are disabled, the CM MUST NOT use piggyback requests. When the CM is operating with Multiple Transmit Channel Mode enabled and segment headers are enabled, the CM MUST only use the request field in the segment header to send a piggyback request.

6.2.5.3 Fragmentation Extended Header

Pre-3.0 DOCSIS fragmented packets use a combination of the Fragmentation MAC header and a modified version of the Upstream Privacy Extended header. Section 6.2.5.4 describes the Fragmentation MAC header. The Upstream Privacy Extended Header with Fragmentation, also known as the Fragmentation Extended Header, transmitted by the CM MUST comply with Table 6–14. CMs operating in Multiple Transmit Channel Mode MUST NOT use fragmentation extended headers.

Table 6–14 - Fragmentation Extended Header Format

EH Element Fields	Usage	Size
EH_TYPE	Upstream Privacy EH element = 3	4 bits
EH_LEN	Length of EH_VALUE = 5	4 bits
EH_VALUE	Key_seq; same as in BP_UP	4 bits
	Ver = 1; version number for this EHDR	4 bits
	BPI_ENABLE If BPI_ENABLE=0, BPI disabled If BPI_ENABLE=1, BPI enabled	1 bit
	Toggle bit; same as in BP_UP [DOCSIS SECv3.0]	1 bit
	SID; Service ID associated with this fragment	14 bits
	REQ; number of mini-slots for a piggyback request	8 bits
	Reserved; set to zero	2 bits
	First_Frag; set to one for first fragment only	1 bit
	Last_Frag; set to one for last fragment only	1 bit
	Frag_seq; fragment sequence count, incremented for each fragment.	4 bits

6.2.5.4 Service Flow Extended Header

The Service Flow EH Element is used to enhance Service Flow operations. It may consist of one or two bytes in the EH_VALUE field. The Payload Header Suppression Header is the only byte in a one byte field or the first byte in a two byte field. The Unsolicited Grant Synchronization Header is the second byte in a two byte field.

[38] Section modified per MULPIv3.0-N-07.0487-2 on 7/11/07 by KN.

6.2.5.4.1 Payload Header Suppression Header

In Payload Header Suppression (PHS), a repetitive portion of the payload headers following the HCS is suppressed by the sending entity and restored by the receiving entity. In the upstream, the sending entity is the CM and the receiving entity is the CMTS. In the downstream, the sending entity is the CMTS and the receiving entity is the CM.

For small payloads, Payload Header Suppression provides increased bandwidth efficiency without having to use compression. Payload Header Suppression may be separately provisioned in the upstream and downstream, and is referenced with an extended header element.

A compliant CM MUST support both PHSI-indexed Payload Header Suppression[39] and DSID-indexed Payload Header Suppression. A CMTS MUST support PHSI-indexed Payload Header Suppression. A CMTS SHOULD support DSID-indexed Payload Header Suppression.

The CM MUST comply with Table 6–15 for Payload Header Suppression Extended Header sub-elements. The CMTS MUST comply with Table 6–15 for Payload Header Suppression Extended Header sub-elements.

Table 6–15 - Payload Header Suppression EHDR Sub-Element Format

EH Element Fields	Usage		Size
EH_TYPE	Service Flow EH_TYPE=5 for downstream and EH_TYPE=6 for upstream		4 bits
EH_LEN	Length of EH_VALUE = 1		4 bits
EH_VALUE	0	Indicates no payload header suppression on current packet.	8 bits
	1-254	Payload Header Suppression Index (PHSI)	
	255	Indicates DSID-indexed PHS	

For PHSI-indexed PHS the Payload Header Suppression Index is unique per service flow in the upstream and unique per CM in the downstream. Payload Header Suppression is disabled if this Extended Header element is omitted or, if included, with the PHSI value set to 0. The Payload Header Suppression Index (PHSI) references the suppressed byte string known as a Payload Header Suppression Field (PHSF).

For DSID-indexed PHS, the EH_VALUE field of the Payload Header Suppression EHDR is set to the static value of 255. Payload Header Suppression is disabled if this Extended Header element is omitted or, if included, with the EH_Value field set to 0. In DSID-indexed PHS, the DSID references the Payload Header Suppression Field (PHSF).

The CM MUST begin the Upstream Suppression Field with the first byte following the MAC Header Checksum. The CMTS MUST begin the Downstream Suppression Field with the thirteenth byte following the MAC Header Checksum. This allows the Ethernet SA and DA to be available for filtering by the CM.

The operation of Baseline Privacy (refer to [DOCSIS SECv3.0]) is not affected by the use of PHS. When Fragmentation is inactive, Baseline Privacy begins encryption and decryption with the thirteenth byte following the MAC Header checksum.

Unless the entire Packet PDU is suppressed, the Packet PDU CRC is always transmitted, and MUST be calculated only on the bytes transmitted. The bytes that are suppressed MUST NOT be included by the CM or CMTS in the CRC calculation.

[39] This is not intended to imply that the CM must be capable of determining when to invoke Payload Header Suppression. Payload Header Suppression support is only required for the explicitly signaled case.

6.2.5.4.2 Unsolicited Grant Synchronization Header

The Unsolicited Grant Synchronization Header may be used to pass status information regarding Service Flow scheduling between the CM and CMTS. It is currently only defined for use in the upstream with Unsolicited Grant and Unsolicited Grant with Activity Detection scheduling services. (Refer to Section 7.2.3.3.)

This extended header is similar to the Payload Suppression EHDR except that the EH_LEN is 2, and the EH_VALUE has one additional byte which includes information related to Unsolicited Grant Synchronization. For all other Service Flow Scheduling Types, the field SHOULD NOT be included by the CM in the Extended Header Element. The CMTS MAY ignore this field.

Table 6–16 - Unsolicited Grant Synchronization EHDR Sub-Element Format

EH Element Fields	Usage		Size
EH_TYPE	Service Flow EH_TYPE = 6		4 bits
EH_LEN	Length of EH_VALUE = 2		4 bits
EH_VALUE	0	Indicates no payload header suppression on current packet.	8 bits (always present)
	1-254	Payload Header Suppression Index (PHSI)	
	Queue Indicator		1 bit
	Active Grants		7 bits

6.2.5.5 BP_UP2 Extended Header[40]

The BP_UP2 EHDR is used when Multiple Transmit Channel Mode is enabled and Baseline Privacy is enabled. When Multiple Transmit Channel Mode is enabled for the CM and segment headers are enabled for a given service flow, the CM MUST use the piggyback opportunity in the segment header for any piggyback requests for that service flow. If segment headers are not enabled for a service flow, the CM is not permitted to create piggyback requests for that service flow. Thus, a piggyback field is not needed in the BP_UP2 EHDR for any service flows. The CM operating in Multiple Transmit Channel Mode with Baseline Privacy Enabled MUST use the BP_UP2 EHDR with a length of 3 for all service flows. The CM MUST comply with Table 6–17 for the BP_UP2 EHDR with length of 3.

Table 6–17 - BP_UP2 EHDR with Length 3

EH Element Fields	Usage	Size
EH_TYPE	Upstream Privacy EH_TYPE = 7	4 bits
EH_LEN	Length of EH_VALUE = 3	4 bits
EH_VALUE	Key_seq; same as in BP_UP	4 bits
	Ver = 1; version number for this EHDR	4 bits

[40] Paragraph and table in section deleted per MULPIv3.0-N-07.0487-2 on 7/11/07 by KN, and modified per MULPIv3.0-N-08.0694-4 on 1/6/09 by JS.

EH Element Fields	Usage	Size
	BPI_ENABLE If BPI_ENABLE=0, BPI disabled If BPI_ENABLE=1, BPI enabled	1 bit
	Toggle bit; same as in BP_UP [DOCSIS SECv3.0]	1 bit
	Reserved, set to zero[41]	14 bits

6.2.5.6 Downstream Service Extended Header

The Downstream Service Extended Header (DS EHDR) communicates to the CM information on how to process downstream packets. The DS EHDR contents vary depending on the EH_LEN, which may be one, three, or five bytes. The CMTS MUST comply with Table 6–18, Table 6–19, and Table 6–20 for DS EHDRs. This header is ignored by CMs which do not implement Downstream Channel Bonding.

Table 6–18 - One-byte DS EHDR Sub-Element Format

EH Element Fields	Usage	Size
EH_TYPE	Downstream Service EH_TYPE = 8	4 bits
EH_LEN	1	4 bits
EH_VALUE	Traffic Priority	3 bits
	Reserved	5 bits

Table 6–19 - Three-byte DS EHDR Sub-Element Format

EH Element Fields	Usage	Size
EH_TYPE	Downstream Service EH_TYPE = 8	4 bits
EH_LEN	3	4 bits
EH_VALUE	Traffic Priority	3 bits
	Reserved	1 bit
	Downstream Service ID (DSID)	20 bits

Table 6–20 - Five-byte DS-EHDR Sub-Element Format

EH Element Fields	Usage	Size
EH_TYPE	Downstream Service EH_TYPE = 8	4 bits
EH_LEN	5	4 bits
EH_VALUE	Traffic Priority	3 bits
	Sequence Change Count	1 bit
	Downstream Service ID (DSID)	20 bits
	Packet Sequence Number	16 bits

[41] Table row text modified per MULPIv3.0-N-07.0529-1 on 10/30 by KN.

When the CMTS classifies a packet to a service flow with a nonzero Traffic Priority (see Annex C.2.2.5.1), it MUST add a DS EHDR and set the Traffic Priority sub-element to the value of the service flow's Traffic Priority parameter.

When the CMTS transmits a packet from a Group Service Flow assigned to a single downstream channel (i.e., non-bonded) it MUST include a three-byte DS EHDR with a DSID. Refer to Section 9.2.2.

When the CMTS transmits a packet from a Service Flow assigned to a Downstream Bonding Group, the CMTS MUST include a five-byte DS EHDR (except if there is a vendor specific configuration to permit the Service Flow to send non-sequenced packets). The DSID in a five-byte DS EHDR is a Resequencing DSID, which identifies a resequencing context. The Packet Sequence Number identifies the sequence number of a packet within the resequencing context identified by the DSID.

A Sequenced Null Packet is defined as a variable-length packet-based MAC frame (Section 6.2.2.1) which includes a five-byte Downstream Service EHDR, does not include any other Extended Header, and has a Packet PDU length of zero. A CMTS MAY send Sequenced Null Packets. A CM MUST accept Sequenced Null Packets. [42]

For a Resequencing DSID, a packet received with a 3-byte DS EHDR MUST be processed by the CM as a non-sequenced packet. For a non-resequencing DSID, a packet received with 5-byte DS EHDR MUST be processed by the CM as a non-sequenced packet. A packet received with a 2-byte DS EHDR MUST be treated by the CM identically to the 1-byte DS EHDR (the extra byte is ignored). A packet received with a 4-byte DS EHDR MUST be treated by the CM identically to the 3-byte DS EHDR (the extra byte is ignored). A packet received with a 6-byte or greater DS EHDR MUST be treated by the CM identically to the 5-byte DS EHDR (the extra byte(s) are ignored).

6.2.5.7 DPV Extended Header

Table 6–21 - DPV Extended Header Format

EH Element Fields	Usage	Size
EH_TYPE	DPV EHDR = 9	4 bits
EH_LEN	Length of EH_VALUE = 5 bytes	4 bits
EH_VALUE	Start Reference Point	8 bits
	Timestamp Start	32 bits

Start Reference Point This is the DPV Reference Point that the DPV measurement originates from (see Section 10.4.2).

Timestamp Start This is the local timestamp at the sender when the DPV packet gets injected into the data stream and departs from the DPV reference point.

The CMTS MAY support the generation of the DPV Extended Header. The CMTS MAY place a DPV EHDR on any packet within any DSID or any Service Flow. The CMTS MUST comply with Table 6–21 for DPV EHDRs. A Modular CMTS Core MAY choose to place a DPV EHDR on any packet within any DEPI flow. This may be done in order to compare the average latency between different Service Flows and/or DEPI flows.

The CM MAY support the generation of the DPV Extended Header.

The CMTS and CM are not required to take any action upon receiving a DPV EHDR other than silently discarding it.

[42] Revised per MULPIv3.0-N-06.0313-5 by GO on 11/21/06.

6.3 Segment Header Format[43]

The CM MUST use a Segment Header when transmitting packets in Multiple Transmit Channel Mode for service flows where use of the segment header is enabled. For these service flows, a Segment Header must appear at the beginning of any transmission made with IUCs 5,6,9,10, or 11. Figure 6–12 shows the segment header format. The segment header is 8 bytes in length. Table 6–22 describes the segment header fields. The CM MUST comply with Figure 6–12 and Table 6–22 for segment headers. [44]

PFI (1 bit)	R (1 bit)	Pointer Field (14 bits)	Sequence # (13 bits)	SC (3 bits)	Request (2 Bytes)	HCS (2 Bytes)

Figure 6–12 - Segment Header Format

Table 6–22 - Segment Header Fields

Field	Usage	Size
PFI	Pointer Field Indicator. This bit is set to a one, to indicate that the pointer field is relevant. When cleared to a zero, this bit indicates that there is no DOCSIS MAC frame starting within this segment and the pointer field is ignored.	1 bit
R	Reserved. This field should be set to a zero by the CM.	1 bit
Pointer Field	When the PFI bit is a one, the value in this field is the number of bytes past the end of the segment header that the receiver will skip when looking for a DOCSIS MAC Header. Thus, a value of zero in the pointer field with the PFI set to one would designate a DOCSIS MAC header beginning just after the segment header.	14 bits
Sequence #	Sequence number that increments by 1 for every segment of a particular service flow.	13 bits
SC	SID Cluster ID of the SID Cluster associated with the Request field of the segment header. The valid SID Cluster ID range is 0 to M-1, where M is the number of SID Clusters per Service Flow supported by the CM.	3 bits
Request	The total number of bytes requested in units of N bytes where N is a parameter of the service flow for which the request is being made. See Annex C.2.2.6.12.	2 bytes
HCS	MAC Header Check Sequence. Similar to HCS used on all MAC headers and is calculated over all other fields in the segment header.	2 bytes

The HCS field is a 16-bit CRC that ensures the integrity of the segment header, even in a collision environment. The CM MUST include all fields within the segment header for the HCS field coverage except the HCS field itself. The HCS is calculated using CRC-CCITT ($x^{16} + x^{12} + x^5 + 1$) as defined in [ITU-T X.25].

For segment header ON operation, the CM may use the piggyback field in the segment header to make piggyback requests for the service flow and MUST NOT use any request EHDR fields within the segment payload.

6.4 MAC Management Messages

6.4.1 MAC Management Message Header[45]

CMs and CMTSs MUST encapsulate MAC Management Messages in an LLC unnumbered information frame per [ISO/IEC 8802-2], which in turn is encapsulated within the cable network MAC framing, as shown in Figure 6–13.

[43] Revised per MULPIv3.0-N-07.0426-1 by ab on 4/30/07.
[44] Revised per MULPIv3.0-N-06.0313-5 by GO on 11/21/06.
[45] Section revised per MULPIv3.0-N-07.0427-3 by ab on 4/30/07 and MULPIv3.0-N-07.0493-4 on 7/13/07 by KN.