International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.9961

Corrigendum 2

(07/2013)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

Access networks – In premises networks

Unified high-speed wire-line based home networking transceivers – Data link layer specification:

Corrigendum 2

Recommendation ITU-T G.9961 (2010) – Corrigendum 2

ITU-T

International Telecommunication Union

ITU-T G-SERIES RECOMMENDATIONS

TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TRANSMISSION MEDIA AND OPTICAL SYSTEMS CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
MULTIMEDIA QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DATA OVER TRANSPORT – GENERIC ASPECTS	G.7000–G.7999
PACKET OVER TRANSPORT ASPECTS	G.8000–G.8999
ACCESS NETWORKS	G.9000–G.9999
In premises networks	**G.9900–G.9999**

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T G.9961

Unified high-speed wire-line based home networking transceivers – Data link layer specification:

Corrigendum 2

Summary

Corrigendum 2 to Recommendation ITU-T G.9961 (2010) contains corrections to the data link layer protocol and security clauses.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T G.9961	2010-06-11	15	11.1002/1000/10706
1.1	ITU-T G.9961 (2010) Cor. 1	2011-12-16	15	11.1002/1000/11145
1.2	ITU-T G.9961 (2010) Amd. 1	2012-09-21	15	11.1002/1000/11144
1.3	ITU-T G.9961 (2010) Cor. 2	2013-07-12	15	11.1002/1000/11899

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

		Page
1)	Clause 2, References	1
2	References	1
2)	Clause 8.1.3.2, Generation of LPDUs	2
3)	Clause 8.1.3.2.1.1, Segment sequence number (SSN)	2
4)	Clause 8.1.3.2.1.2, LLC frame boundary offset (LFBO)	3
5)	Clause 8.1.3.2.1.5, Oldest pending segment flag (OPSF)	3
6)	Clause 8.2.3, TXOP timing	3
7)	Clause 8.3.3.4.4, Use of RTS/CTS signalling	4
8)	Clause 8.3.7, Bidirectional transmissions	4
9)	Clause 8.5.3, Routing of ADPs	7
10)	Clause 8.5.4, Broadcast of LLC frames	9
11)	Clause 8.5.7.1, Relaying of LCDU	15
12)	Clause 8.5.7.2, Relaying of APDU	18
13)	Clause 8.6, Domain master node functional capabilities	20
14)	Clause 8.6.2.1, Description of TSpec parameters	28
15)	Clause 8.6.2.2.3, Flow maintenance	28
16)	Clause 8.6.2.3.1, Format of CL_EstablishFlow.req	29
17)	Clause 8.6.2.3.2, Format of CL_EstablishFlow.cnf	32
18)	Clause 8.6.2.3.8, Format of FL_AdmitFlow.req (from ITU-T G.9961 Corrigendum 1)	34
19)	Clause 8.6.4.3.1, Format of TM_NodeTopologyChange.ind (from ITU-T G.9961 Corrigendum 1)	35
20)	Clause 8.6.4.3.5, Selection and maintenance of the DNI (from ITU-T G.9961 Corrigendum 1)	37
21)	Clause 8.6.6.1, Domain master selection at initialization	40
22)	Clause 8.6.7, Selection of PHY-frame header segmentation	42
23)	Clause 8.6.8.2, Selection and maintenance of the DNI	43
24)	Clause 8.8.1, MAP generation and distribution, and clause 8.8.2, MAP frame transmission (both taken from G.9961 Corrigendum 1)	43
25)	Clause 8.8.3, MAP header	45
26)	Clause 8.8.4.1.1, TXOP attributes extension data	47
27)	Clause 8.8.4.1.5, CBTS nodes information Extension Data	48
28)	Clause 8.8.5, Auxiliary information field	49

		Page
29)	Clause 8.8.5.2, Domain name sub-field	50
30)	Clause 8.8.5.5, PSD-related domain info sub-field	51
31)	New clause 8.8.5.11, NMK_DB_update sub-field	53
32)	Clause 8.8.6, MAP schedule persistence publication	53
33)	Clause 8.9.2.2, Multicast acknowledgement procedure	54
34)	Clause 8.9.4.2, Transmitter variables and control flags, and clause 8.9.4.3, Receiver variables and control flags	54
35)	Clause 8.9.5.3, Acknowledgement protocol state machine for unicast transmission	56
36)	Clause 8.9.5.3.1, Transmission window operation	59
37)	Clause 8.10.1, Management message format (from ITU-T G.9961 Corrigendum 1)	60
38)	Clause 8.10.2, Control message format	67
39)	Clause 8.11, Channel estimation protocol (from ITU-T G.9961 Corrigendum 1)	68
	8.11 Channel estimation protocol	68
40)	Clause 8.11.1.1, Channel estimation initiation (from ITU-T G.9961 Corrigendum 1)	69
41)	Clause 8.11.1.3, Channel estimation initiation confirmation (from ITU-T G.9961 Corrigendum 1)	70
42)	Clause 8.11.2.1, Channel estimation request (from ITU-T G.9961 Corrigendum 1)	70
43)	Clause 8.11.3.2, Partial BAT update (from ITU-T G.9961 Corrigendum 1)	70
44)	Clause 8.11.4, Channel estimation using PROBE frames (from ITUT G.9961 Corrigendum 1)	71
45)	Clause 8.11.7.3, Format of CE_ParamUpdate.req	72
46)	Clause 8.11.7.12, Format of CE_Initiation.req, and clause 8.11.7.13, Format of CE_Initiation.cnf (both taken from ITU-T G.9961 Corrigendum 1)	74
47)	Clause 8.12.7, Reset of a unicast connection with acknowledgements	75
48)	New clause 8.12.9, Multicast data connection	75
49)	Clause 8.16.1, Initialization of a PHY multicast group (from ITU-T G.9961 Corrigendum 1)	75
50)	Clause 8.16.2, Maintenance of multicast binding information (from ITU-T G.9961 Corrigendum 1)	77
51)	Clause 8.17.1, DLL multicast stream establishment (from ITU-T G.9961 Corrigendum 1)	78
52)	New clause 8.18, Inter-bandplan interoperability	78
53)	New clause 8.19, Version control and capabilities exchange	81
54)	Clause 9.1.1.3, Input variables	81
55)	Clause 9.1.2.3, CCMP header	83

Page

56) Clause 9.2.1, Authentication and key management procedures 85
57) Clause 9.2.2, Authentication to the domain 86
58) Clause 9.2.2.1, Authentication 87
59) Clause 9.2.2.2, The PAK protocol parameters 91
60) Clause 9.2.3.1, Generation of point-to-point and point-to-multipoint encryption keys 93
61) Clause 9.2.4, Updating and termination of encryption keys 95
62) Clause 9.2.5, Messages supporting AKM procedures 96
63) Clause 9.2.5.2, Pair-wise authentication messages 99
64) Clause 9.2.5.3, Key updating messages 103
65) Clause A.1.1, Frame conversion 104
66) New Annex V, Versioning dependencies of ITU-T G.9961 106

Annex V – Versioning dependencies of ITU-T G.9961 107

Recommendation ITU-T G.9961

Unified high-speed wire-line based home networking transceivers – Data link layer specification:

Corrigendum 2

1) Clause 2, References

Revise clause 2 "References" as follows:

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T G.9960~~1~~] Recommendation ITU-T G.9960 (2010), *Unified high-speed wire-line based home networking transceivers – system architecture and physical layer specification*.

[ITU-T G.9972~~2~~] Recommendation ITU-T G.9972 (2010), *Coexistence mechanism for wireline home networking transceivers*.

[ITU-T X.1035~~5~~] Recommendation ITU-T X.1035 (2007), *Password-authenticated key exchange (PAK) protocol*.

[IEEE 802.1ad] IEEE 802.1ad-2005, *IEEE Standard for Local and metropolitan area networks: Provider Bridges*.

[IEEE 802.1D~~6~~] IEEE 802.1D-2004, *IEEE Standard for Local and metropolitan area networks Media Access Control (MAC) Bridges* – http://standards.ieee.org

[IEEE 802.1Q] IEEE 802.1Q-2005, *IEEE Standard for Local and Metropolitan Area Networks – Virtual Bridged Local Area Networks – Revision*.

[IEEE 802.3] IEEE 802.3-2008, *IEEE Standard for Information technology-Specific requirements – Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications*.

[NIST FIPS 197~~3~~] FIPS PUB 197 (2002), *Specification for the Advanced Encryption Standard (AES)*, National Institute of Standards and Technology, November, 2001 – http://csrc.nist.gov/publications/.

[NIST 800-38C~~4~~] Special Publication 800-38C, *Recommendation for Block Cipher Modes of Operation*: *the CCM Mode for Authentication and Confidentiality*, **N**ational Institute of Standards and Technology, May 2004 – http://csrc.nist.gov/publications/nistpubs/800-38C_updated-July20_2007.pdf.

[~~7~~NIST FIPS 180-3] FIPS PUB 180-3 (2008), *Secure Hash Standard (SHS)*, National Institute of Standards and Technology, October, 2008 – http://csrc.nist.gov/publications/.

2) Clause 8.1.3.2, Generation of LPDUs

Revise the text of clause 8.1.3.2 "Generation of LPDUs" as follows:

8.1.3.2 Generation of LPDUs

…

Multiple LLC frames carrying APDUs (data LLC frames) associated with the same data connection can be concatenated to form a data LLC frame block. LLC frames containing LCDUs (management LLC frames) that belong to the same management connection can be concatenated to form a management LLC frame block. The LLC frame block may also include LLC frames intended to be relayed that are associated with the same connection. The number of concatenated LLC frames for an LLC frame block is determined by DLL management entity and is vendor discretionary. The order of LLC frames of the same user priority in the LLC frame block (see Figure 8-5) containing APDUs shall be the same as the order of arrival of the APDUs sourcing these frames. The order of LLC frames in the LLC frame block containing LCDUs shall be the same as the order that these LCDUs are generated by the DLL management entity. The order of bytes in the LLC frame payload shall be the same as in sourcing APDUs or LCDUs, and in the same relative order, bytes shall be passed to the MAC as LPDUs that the MAC maps into MPDU. Mixing data LLC frames and management LLC frames into the same LLC frame block (mixed LLC frame block) is allowed only if the lowest user priority associated with the corresponding prioritized data connection is equal to or greater than six and the highest user priority associated with the corresponding prioritized data connection is equal to seven (i.e., in case three or more priority queues are supported as described in Table III.1 of [ITU-T G.9960]). In this case LCDUs shall be mapped to the same prioritized data connection where APDUs with user priority 7 for the same destination are mapped. Mixed LLC frame blocks are not allowed for data connections associated with service flows. A ~~M~~mixed LLC frame block ~~is~~ shall be treated as a data LLC frame block (i.e., The MQF flag in the LPH shall be set to zero (see clause 8.1.3.2.1.4).

NOTE 1 – Mixing data LLC frames and management LLC frames into the same LLC frame block can result in segments containing fragments of both a data LLC frame and a management LLC frame.

…

3) Clause 8.1.3.2.1.1, Segment sequence number (SSN)

Revise the text of clause 8.1.3.2.1.1 "Segment sequence number (SSN)" as follows:

8.1.3.2.1.1 Segment sequence number (SSN)

This field identifies the relative location of the segment within the stream of segments corresponding to a connection. Segment s~~S~~equence n~~N~~umber (SSN) is a 16-bit field indicating the order of segments that are associated with a connection. The SSN shall be initialized to START_SSN (see clause 7.1.2.3.2.2.20 of [ITU-T G.9960]) ~~zero~~ for the first valid segment that belongs to a new connection and shall be incremented by 1 for each new valid segment that is associated with this connection that follows the current segment.

In the case of a PHY frame with payload not belonging to any connection (CNN_MNGMT field of PHY frame header equal to 1111_2), the SSN shall be initialized to a vendor discretionary value for the first valid segment of each MPDU transmitted to a DID and shall be incremented by 1 for each new valid segment of that MPDU.

The SSN shall be expressed as a 16-bit unsigned integer and shall wrap around (goes back to value 0000_{16} after $FFFF_{16}$).

NOTE – A receiver might receive segments in an "out-of-order" manner when lost segments are retransmitted or LPDUs from the management LLC frame block are members of the same MPDU.

4) Clause 8.1.3.2.1.2, LLC frame boundary offset (LFBO)

Revise the text of clause 8.1.3.2.1.2 "LLC frame boundary offset (LFBO)" as follows:

8.1.3.2.1.2 LLC frame boundary offset (LFBO)

This field indicates the location of the start of the first LLC frame within the segment. This enables the receiver to recover when one or more segments are lost (e.g., when the transmitter drops a segment due to timeout). The LFBO is a 10-bit field that carries the offset in octets of the first octet of the first new LLC frame relative to the start of the segment of the LPDU (in case the LLC frame starts at the start of the segment, the LFBO = 0). The first new LLC frame may be type 0 (padding) or any other type (see Table 8-2).

The value of LFBO shall be coded as an unsigned integer as shown in Table 8-4.

Table 8-4 – Format of LFBO

LFBO Value	Description
000_{16} to 213_{16}	The LLC frame boundary offset in bytes
214_{16} to $3FE_{16}$	Reserved by ITU-T
$3FF_{16}$	No LLC frame boundary exists in the LPDU

5) Clause 8.1.3.2.1.5, Oldest pending segment flag (OPSF)

Revise the text of clause 8.1.3.2.1.5 "Oldest pending segment flag (OPSF)" as follows:

8.1.3.2.1.5 Oldest pending segment flag (OPSF)

For connections with acknowledgements, ~~T~~this field indicates whether the segment is the oldest pending segment in the transmitter queue associated with the connection. This enables the receiver to determine that all older segments are dropped, thus enabling it to process the oldest pending segment and subsequent segments without waiting for older segments. When OPSF is set to one, it indicates that the segment is the oldest segment present at the transmitters queue. When set to zero, it indicates that the segment is not the oldest pending segment in the transmitters queue.

For connections without acknowledgements and for payload not belonging to any connection (CNN_MNGMT field of PHY frame header equal to 1111_2), this field shall be set to one for the first segment in the MPDU and shall be set to zero for all other segments in that MPDU.

6) Clause 8.2.3, TXOP timing

Revise the text of clause 8.2.3 "TXOP timing" as follows:

8.2.3 TXOP timing

The start time of a TXOP can be specified (or inferred) in the following two ways:

1) Implicitly, using the start time and duration of the previous TXOP, as specified in clause 8.8.4.1.1.
2) Explicitly, using TXOP absolute timing, as specified in clause 8.8.4.1.2.

By default, the TXOP start-time is implicitly defined as the start time of the TXOP~~[n]~~ associated with the previous TXOP descriptor in the MAP plus the duration of that TXOP~~is defined implicitly and is equal to the start-time of TXOP[n–1] plus the duration of TXOP[n–1]~~. The ~~implicit~~ start-time of the ~~first~~ TXOP associated with the first TXOP descriptor in the MAP is implicitly defined as the start of the MAC cycle.

The explicit specification of the start-time of a particular TXOP is relative to the start time of the MAC cycle.

…

7) Clause 8.3.3.4.4, Use of RTS/CTS signalling

Revise the text of clause 8.3.3.4.4 "Use of RTS/CTS signalling" as follows:

8.3.3.4.4 Use of RTS/CTS signalling

…

Nodes that detected no CTS frame during the time period of $T_{CTS\text{-}MAX}$ microsecond after the RTS frame was transmitted shall declare the status of the CTS frame as "not received". The value of $T_{CTS\text{-}MAX}$ shall be equal to:

$$T_{CTS\text{-}MAX} = T_{RTS} + T_{RCIFG} + T_{CTS} + T_{CCIFG}$$

where T_{RTS} and T_{CTS} are the durations (i.e., transmission times) of the RTS and CTS frames, respectively, and T_{RCIFG} and T_{CCIFG} are the durations of the RCIFG and CCIFG gaps, respectively (see clause 8.4).

…

8) Clause 8.3.7, Bidirectional transmissions

Revise the text of clause 8.3.7 "Bidirectional transmissions" as follows:

8.3.7 Bidirectional transmissions

Bidirectional transmissions between two nodes may be used to improve throughput and minimize latency of a traffic that is bidirectional in nature, such as TCP traffic with acknowledgements. The defined bidirectional mechanism is only applicable to nodes communicating directly (i.e., not via a relay node).

In case of bidirectional transmission, a node originating (sourcing) the bidirectional traffic and the destination node exchange special frames: a bidirectional message (BMSG) frame and a bidirectional acknowledgement (BACK) frame. Both BMSG and BACK carry data, and in the case of acknowledged transmissions, also an acknowledgement on the recently received frame.

If using acknowledged bidirectional transmission, the BMSG PHY frames shall use the format described in Tables 7-43 and 7-5144 of [ITU-T G.9960], and the BACK PHY frames shall use the format described in Tables 7-45 and 7-5246 of [ITU-T G.9960], in which the PHY frame header contains $2\times PHY_H$ information bits (EHI bit, in the PHY frame header, is set to one, see clause 7.1.2.3.1.7 of [ITU-T G.9960]). If using unacknowledged bidirectional transmission, the BMSG and BACK PHY frames shall use the format described in Tables 7-43 and 7-45 of [ITU-T G.9960], respectively, in which the PHY frame header contains PHY_H information bits (EHI bit in the PHY frame header is set to zero).

An exchange of BMSG and BACK frames forms a bidirectional frame sequence that shall last strictly inside the boundaries of the particular TXOP or TS assigned in the MAP for the node sourcing the bidirectional transmission, see Figure 8-24. With an acknowledged bidirectional transmission only immediate acknowledgement is allowed (the valid values of RPRQ field are 00 and 01 only).

A bidirectional transmission may be initiated by either a source node or a destination node using one of the following methods:

- A destination node, in case of acknowledged transmission, transmits to the source node, in response to a MSG frame requesting immediate acknowledgement, an ACK frame with the BTXRQ bit set to one.
- A destination node, in case of un-acknowledged transmission, transmits to the source node a MSG frame with BTXRQ bit set to one.
- A source node transmits to the destination node a BMSG frame with the BTXGL field set to a non-zero value.

If a source node requested by a destination node to initiate bidirectional transmission accepts the request, it shall indicate that the request is granted and shall initiate bidirectional transmission by transmitting a BMSG frame with the BTXGL field set to a non-zero value. A source node requested to initiate bidirectional transmission may decline the request. In this case it indicates that the bidirectional transmission request is declined by continuing to send MSG frames to the requesting node, instead of BMSG frames.

A source node may initiate bidirectional transmission autonomously, without a request from the destination node by transmitting a BMSG frame with the BTXGL field set to a non-zero value.

The acknowledgement information in a BMSG frame that initiates a bidirectional transmission shall be conveyed according to the following rules:

- If the recent MSG frames including the last MSG frame received from the destination node were already acknowledged, the acknowledgement information of the BMSG frame may either repeat the last acknowledgement information sent for this connection, or disable the acknowledgement information by setting the CONNECTION_ID to 255 and MNMTP to 0 in the ACKDATA_BM as described in clause 7.1.2.3.2.3.9.1.4 of [ITU-T G.9960].
- If the last MSG frame for the connection received from the destination node was not acknowledged and an acknowledgement is required for the connection, the BMSG frame shall include acknowledgement information on the recent MSG frames including the last MSG frame received from the destination node.

~~be disabled by setting the FACK field to 111 if the last MSG frame received from the destination node was already acknowledged or no acknowledgement is required; otherwise it shall include acknowledgement information on the recent MSG frames received from the destination node.~~

A source node may at any time terminate bidirectional transmission and re-start it again. The destination node may indicate to the source node when the bidirectional transmission may be stopped, while the decision is up to the source node.

Once bidirectional transmission is initiated by the sourcing node, the following procedure shall be used for bidirectional transmission:

1) A destination node responds to the BMSG frame that initiates bidirectional transmission by transmitting a BACK frame that contains data in the payload intended for the source node. If the source node requested acknowledgement the BACK frame additionally contains acknowledgement information for data previously transmitted by the source node. In the BTXRL field of the frame header the destination node indicates the requested duration of the next BACK frame it expects to transmit.
2) The source node, in response to the received BACK frame, transmits a BMSG frame indicating the granted maximum duration of the next BACK frame in the BTXGL field of the PHY-frame header.
3) The destination node, in response to the BMSG frame, transmits a BACK frame, continuing the exchange between the communicating nodes. The duration (see clause 7.1.2.3.2.10.1 of [ITU-T G.9960]) of the BACK frame shall not exceed the granted duration.

4) The source node may terminate the bidirectional transmission by one of the following methods:

   a) By setting BTXGL = 0 in any of the BMSG frames. In case BTXGL = 0 in the received BMSG frame and the RPRQ field indicates request for immediate acknowledgement, the destination node shall respond by an Imm-ACK frame.

   b) By setting BTXEF = 1 and BTXGL ≠ 0 in any of the BMSG frames. In this case, as BTXGL ≠ 0, the destination node may send a BACK frame prior to the termination of bidirectional transmission.

   c) By sending an Imm-ACK frame, in case of acknowledged transmission, instead of BMSG frame. Previous BMSG frames in the frame sequence shall all carry BTXEF = 0.

5) The destination node may indicate that bidirectional transmission is not further needed (advice for termination of bidirectional transmission) by setting the BTXRL = 0 in the BACK frame. In response, the source node may terminate bidirectional transmission using any of three methods described above.

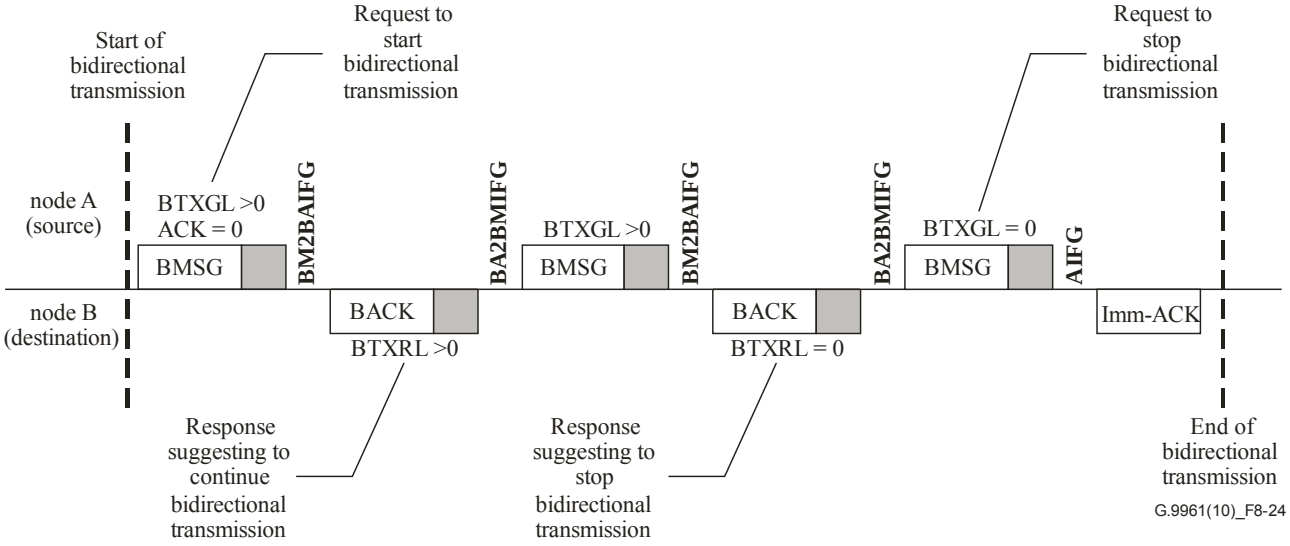


Figure 8-24 – Example of bidirectional transmission (invited by the originating node)

NOTE – Figure 8-24 presents a case when the destination node suggests to terminate bidirectional transmission and the source node requests that termination shall be done by the destination node (the destination node sends Imm-ACK). The source node may also terminate the bidirectional transmission itself by sending Imm-ACK instead on the last BMSG frame with the BTXGL field set to zero.

The maximum duration of a BACK frame is determined by the source node in the BTXGL field of the PHY-frame header. The destination node only indicates the desired duration of BACK frame in the BTXRL field of the PHY-frame header of the previous BACK frame, but the final decision on the BACK frame duration limit (including the following IFG) is done by the source node. If a destination node indicates in the RPRQ field that Imm-ACK is requested, the source node shall set the maximum granted length for BACK transmission so that there is sufficient time for the source node to transmit an Imm-ACK frame at the end of the transmission sequence (in response to the last BACK frame).

A responding BACK frame shall be transmitted $T_{BM2BAIFG}$ after the BMSG frame, and the responding BMSG frame shall be transmitted $T_{BA2BMIFG}$ after the BACK frame. The Imm-ACK frame shall be transmitted T_{AIFG} after the BMSG frame or after the BACK frame, respectively. In all of the following frame sequences:

- BMSG followed by a BACK

- BACK followed by a BMSG
- BMSG followed by an Imm-ACK
- BACK followed by an Imm-ACK

if the transmitter of the first frame has no knowledge of the 'receiver specific' AIFG (see clause 8.6.1.1.4.1 and clause 8.6.4.3.1) or if the first frame in any of the above frame sequences includes less than MIN_SYM_VAR_AIFG ~~payload~~ symbols, the gap between this frame and the following frame shall be T_{AIFG-D} (see clause 8.4), otherwise the gap shall be $T_{AIFG.}$ The parameter MIN_SYM_VAR_AIFG is defined in clause 8.4, for each media. The transmitter indicates usage of either T_{AIFG} or T_{AIFG-D} by using the AIFG_IND bit in the PHY-frame header (see clause 7.1.2.3.2.2.16 of [ITU-T G.9960]).

Bidirectional transmission can be used in CFTXOP, STXOP, and CBTXOP. The source node shall ensure that the total duration of the bidirectional frame sequence does not violate the boundaries of the TXOP or the maximum allowed duration of the TS. Particularly:

– if bidirectional transmission is established in a CFTXOP, the last frame in the sequence shall end at least T_{IFG_MIN} before the end of the CFTXOP;

– if bidirectional transmission is established in a CFTS or in a CBTS, the last frame in the sequence shall end at least T_{IFG_MIN} before the end of the Max_TS_Length assigned in the MAP for the TS and at least T_{IFG_MIN} before the end of the TXOP where this TS is defined.

Both the BMSG frame and the BACK frame may be sent as bursts of frames. The format of burst transmission and associated rules shall be as defined in clause 8.3.5 (all frames in a burst shall be of BMSG type or of BACK type). In case of acknowledged transmission, the acknowledgement information in the BACK and BMSG frame header shall use the format described in clause 8.3.5. All BMSG (or BACK) frames of the same burst shall carry the same acknowledgement information.

Both BMSG and BACK frames indicate their duration in the Duration field of the PHY-frame header as defined in clause 7.1.2.1 of [ITU-T G.9960]. For virtual carrier sense, the end of the bidirectional transmission frame sequence shall be calculated based on the duration of the last BMSG frame sent by the source node and the values of BTXEF, BTXGL and RPRQ depending on how the bidirectional transmission is terminated. When the bidirectional transmission is terminated with a BMSG frame with BTXEF = 1 and BTXGL $\neq$ 0, the total duration of the frame sequence shall include this BTXGL value, regardless of the actual duration of the last BACK frame.

Nodes detecting a bidirectional transmission shall stay silent until the end of the bidirectional transmission sequence or until the expiration of the Max_TS_Length of the corresponding TS, whichever comes first.

Bidirectional transmission is not allowed when RTS/CTS is used.

9) Clause 8.5.3, Routing of ADPs

Revise the text of clause 8.5.3 "Routing of ADPs" as follows:

8.5.3 Routing of ADPs

Each node shall inform the domain master about the nodes of its domain it has detected as defined in clause 8.6.4.3.

Each node can have one or more applications associated with its AE (above its A-interface). Each application is identified by a unique 6-octet MAC address. Each node shall maintain the full list of MAC addresses associated with applications above its A-interface as well as its own MAC address. This list is referred to as a local address association table (LAAT). Each node shall also maintain the list of MAC addresses associated with the AEs of other nodes in the domain and the MAC

addresses of those nodes. This list is referred to as a remote address association table (RAAT). Each node provides its local AAT to the domain master and other nodes of the domain using topology management messages as described in clause 8.6.4.3.

The address association table (AAT) is formed by the aggregation of the LAAT and the RAAT.

Whenever a node receives an ADP from the A-interface, it uses its AAT to determine if the ADP is intended for the node itself (local in-band management message, see Annex A) or for an AE associated with another node.

- If the ADP is intended for a remote AE or is an in-band management message addressed to a different node (case B of Table 8-14.1), the node shall determine the destination DEVICE_ID of the node in its domain through which the remote AE can be reached and send the corresponding ADP directly or via relay nodes to this node. This destination DEVICE_ID is provided to the Flow Mapper (see Figure 8-2) and is further reached either directly or via relays.
- If the ADP is intended for a group MAC address belonging to the AEs of different nodes of the domain (case D of Table 8-14.1), the node shall associate this ADP with a destination MSID and it shall send the APDU using DLL multicast transmission. The node may send the APDU to the appropriate nodes using unicast transmissions until the DLL multicast paths toward the appropriate nodes are established. The node may send the APDU using a combination of DLL multicast and DLL unicast transmissions until the relevant DLL multicast path is established.

  NOTE 1 – The association between the group of MAC addresses and addressed nodes is provided by the DLL management entity. The mechanism of this association is vendor discretionary and may be based on various multicast protocols, such as IGMP.
- If the destination address of the ADP is a standard broadcast address ($FFFFFFFFFFFF_{16}$) (case E of Table 8-14.1), then the BRCTI bit in the LFH of the LLC frame carrying the corresponding APDU shall be set to one, so that the APDU will be broadcast to all nodes in the domain using the procedure described in clause 8.5.4. If the EtherType of the ADP equals $22E3_{16}$, the corresponding APDU shall also be forwarded to the local DLL management entity.

  NOTE 2 – For ADP with EtherType different from $22E3_{16}$ and the standard broadcast address as the DA of that ADP, sending the corresponding APDU to the local DLL management entity is vendor discretionary.
- If the destination address of a received ADP is found in the local AAT and it is not the MAC address of the node (case A of Table 8-14.1), the ADP shall be dropped without notification.
- If the destination address of a received ADP is the MAC address of the node (case C of Table 8-14.1), the node shall pass the corresponding APDU to its DLL management entity.
- If the destination address of a received ADP is the reserved MAC address 01-19-A7-52-76-96 (case F of Table 8-14.1), the node shall pass the corresponding APDU to its DLL management entity
- If the destination MAC address corresponds to a unicast MAC address and the destination node cannot be inferred from previous rules (not covered in cases A, B, C and F), then the BRCTI bit in the LFH of the LLC frame carrying the corresponding APDU shall be set to one, so that the APDU will be broadcast to all nodes in the domain using the procedure described in clause 8.5.4 (case G of Table 8-14.1).
- If the destination MAC address corresponds to a group MAC address for which the destination nodes cannot be inferred or a group MAC address intended to reach all the nodes of the domain (case H of Table 8-14.1), then the BRCTI bit in the LFH of the LLC

frame carrying the corresponding APDU shall be set to one, so that the APDU will be broadcast to all nodes in the domain using the procedure described in clause 8.5.4.

Table 8-14.1 – Routing of ADPs

Case	Ethernet frame type	ADP Destination address	Routing	Example
A	Unicast frame	In LAAT, except node's MAC address	Drop the message	Any kind of traffic
B	Unicast frame	In RAAT	Look for the DestinationNode defined for this DA	Normal routing of frames coming through the A interface (can be normal Ethernet or remote in-band messages)
C	Unicast frame	Node's MAC address	Send to DLL management	Local in-band message
D	Multicast frame	Multicast address mapped to known destination device(s)	The node has the choice to treat this multicast transmission as several DLL unicast transmissions or using a DLL multicast stream	IGMP/MLD Ethernet frames
E	Broadcast frame	Broadcast address	If EtherType = $22E3_{16}$ send to DLL management treat this broadcast transmission using BRT (BRCTI=1; DestinationNode = BROADCAST_ID) and route following the BRT rules	Normal broadcast
F	Unicast frame	Reserved address	Send to DLL management	
G	Unicast Frame	Destination MAC address not covered by cases A, B, C and F	Treat this case as a broadcast transmission using BRT (BRCTI=1; DestinationNode = BROADCAST_ID) and route following the BRT rules	Any kind of traffic
H	Multicast Frame	Destination device(s) cannot be inferred from the DA or Frame intended for all devices	Treat this case as a broadcast transmission using BRT (BRCTI=1; DestinationNode = BROADCAST_ID) and route following the BRT rules	Multicast protocol (IGMP/MLD) control frames

10) Clause 8.5.4, Broadcast of LLC frames

Revise the text of clause 8.5.4 "Broadcast of LLC frames" as follows:

8.5.4 Broadcast of LLC frames

To facilitate broadcast of an LLC frame, every node shall obtain the broadcast routing table (BRT), as defined in clause 8.6.4.1.1.2. The BRT of a particular node contains a list of destination nodes (list of DEVICE_IDs), to which this particular node shall relay a broadcasted APDU or LCDU that

was received from the medium from ~~a~~ specified root nodes. This list depends on the source from which the broadcasted APDU or LCDU was received (see clause 8.6.4.1.1.2). It is up to the node to create PHY multicast groups (see clause 8.16) or use PHY unicast transmissions or PHY broadcast transmissions to reach the destination nodes indicated in the BRT (the DID of the PHY frame could be a DEVICE_ID, or a MULTICAST_ID, or a BROADCAST_ID (FF_{16})).

To broadcast an LLC frame, the node that originates the broadcast shall set the BRCTI bit in the LFH of the transmitted APDU or LCDU to one, and set the DestinationNode of the LFH field to FF_{16}. The DA of the broadcasted frame may be any address, including the standard broadcast address ($FFFFFFFFFFFF_{16}$).

A node that receives a broadcast LLC frame (APDU or LCDU, BRCTI = 1) from the medium, shall first perform the filtering procedure according to the BRT as described in clause 8.5.4.1. If the node does not drop the LLC frame as a result of that filtering procedure, the node shall perform the actions described in the rest of this clause.

A node that receives a broadcast LLC frame from the medium (APDU or LCDU, BRCTI = 1) shall forward this frame to the nodes indicated in the BRT (as indicated in some of the cases specified in Tables 8-14.2 and 8-14.3) without modifying the value of BRCTI.

NOTE – Nodes that are leaf nodes of the tree will have an empty branch path in its BRT (see clause 8.6.4.1.1.2), while non-leaf nodes of the tree will have one or more destination entries in its branch path. Non-leaf nodes are supposed to have relay capabilities in this description.

If a node received from the medium a broadcast LLC frame that contains an LCDU with DestinationNode different from BROADCAST_ID or its own DEVICE_ID, it shall relay the LLC frame as indicated by the BRT (cases 1 and 2 of Table 8-14.2).

If a node received from the medium a broadcast LLC frame that contains an LCDU with DestinationNode equal to the node's DEVICE_ID (case 3 of Table 8-14.2), it shall recover this LCDU and treat it as an unicast frame for relaying purposes (see clause 8.5.7). The node shall not relay the broadcast LLC frame.

If a node received from the medium a broadcast LLC frame that contains an LCDU with DestinationNode equal to BROADCAST_ID:

- If the node is a leaf node:
  - If the DA of that LCDU is the MAC address of the node, or the standard broadcast address, or the reserved MAC address 01-19-A7-52-76-96 (cases 6, 7 and 8 of Table 8-14.2), the node shall recover this LCDU and pass it to the DLL management. In addition, the node shall stop the broadcast of the LLC frame.
  - In all other cases (cases 4, 5 and 9 of Table 8-14.2), the LLC frame shall be dropped and not relayed.
- If the node is a non-leaf node:
  - If the DA of that LCDU is the standard broadcast address or the reserved MAC address 01-19-A7-52-76-96, the node shall recover this LCDU and pass it to the DLL management (cases 13 and 14 of Table 8-14.2). In addition, the node shall relay that LLC frame as indicated by the BRT.
  - If the DA of that LCDU is the MAC address of the node (case 12 of Table 8-14.2), the node shall recover this LCDU and pass it to the DLL management. In addition, the node may relay the LLC frame as indicated by the BRT.
  - In all other cases (cases 10, 11 and 15 of Table 8-14.2), the LLC frame shall be relayed as indicated by the BRT.

If a node received from the medium a broadcast LLC frame that contains an APDU with DestinationNode different from BROADCAST_ID or its own DEVICE_ID, it shall relay the LLC frame as indicated by the BRT (cases 16 and 17 of Table 8-14.3).

If a node received from the medium a broadcast LLC frame that contains an APDU with DestinationNode equal to the nodes DEVICE_ID (cases 18 and 19 of Table 8-14.3), it shall recover this APDU and treat it as an unicast frame for relaying purposes (see clause 8.5.7). The node shall not relay the broadcast frame.

If a node received from the medium a broadcast LLC frame that contains an APDU with DestinationNode equal to BROADCAST_ID, it shall:

- If the node is a leaf node:
  - If the DA of that APDU is the address of the DLL management or the reserved MAC address 01-19-A7-52-76-96 (cases 22 and 24 of Table 8-14.3), the node shall recover this APDU and shall pass it to the DLL management. In addition, the node shall stop the broadcast of the LLC frame.
  - If the DA of that APDU is the standard broadcast address (case 23 of Table 8-14.3), the node shall recover this APDU and shall pass it to the DLL management and to the A-interface. In addition, the node shall stop the broadcast of the LLC frame.
  - If the DA of that APDU is in the LAAT (case 20 of Table 8-14.3), the node shall recover this APDU and shall pass it to the A-interface. In addition, the node shall stop the broadcast of the LLC frame.
  - If the DA of that APDU is in the RAAT (case 21 of Table 8-14.3), the node shall recover this APDU and may pass it to the A-interface. In addition, the node shall stop the broadcast of the LLC frame.
  - In the cases not covered by the previous four4 bullets (i.e., case 25 of Table 8-14.3), the LLC frame shall be passed to the A-interface and not relayed.
- If the node is a non-leaf node:
  - If the DA of that APDU is the address of the DLL management (case 28 of Table 8-14.3), the node shall recover this APDU and shall pass it to the DLL management. In addition, the node may stop relaying the LLC frame.
  - If the DA of that APDU is the address of the reserved MAC address 01-19-A7-52-76-96 (case 30 of Table 8-14.3), the node shall recover this APDU and shall pass it to the DLL management. In addition, the node shall relay the LLC frame as indicated in the BRT.
  - If the DA of that APDU is the standard broadcast address (case 29 of Table 8-14.3), the node shall recover this APDU and shall pass it to the DLL management and to the A-interface. In addition, the node shall relay the LLC frame.
  - If the DA of that APDU is in the LAAT (case 26 of Table 8-14.3), the node shall recover this APDU and shall pass it to the A-interface. In addition, the node may relay the LLC frame as indicated in the BRT.
  - If the DA of that APDU is in the RAAT the node shall relay the LLC frame as indicated in the BRT (case 27 of Table 8-14.3). In addition, the node may recover this APDU and pass it to the A-interface.
  - In the cases not covered by the previous five bullets (i.e., case 31 of Table 8-14.3), the LLC frame shall be passed to the A interface and also relayed following the BRT.

Table 8-14.2 – Broadcast of LLC frames (LCDU case)

Case	Type of broadcast	Leaf/ Non-leaf	LCDU DA	Broadcasting actions	Example
1	Broadcast frame intended for another node in the network (BRCTI = 1; MCSTI = 0; DestinationNode = $DeviceID_{OtherNode}$)	Leaf	–	Drop the frame (Note 2)	Unicast frame not found by a previous relay node in its internal unicast routing tables and relayed in broadcast
2		Non-Leaf	–	Follow BRT rules	Unicast frame not found by a previous relay node in its internal unicast routing tables and relayed in broadcast
3	Broadcast frame intended for this node (BRCTI = 1; MCSTI = 0; DestinationNode = $DeviceID_{Node}$)	Leaf/ Non leaf	–	Consider frame as non-broadcast (unicast) and follow the corresponding rules (cases 1-6 of Table 8-14.4). Stop the broadcast	Endpoint of a unicast frame not found by a previous relay node in its internal unicast routing tables and relayed in broadcast
4	Broadcast frame intended for all the nodes (BRCTI = 1; MCSTI = 0; DestinationNode = BroadcastID)	Leaf	In LAAT except node's MAC address	Drop the frame	Not applicable
5			In RAAT	Drop the frame	
6			Node's MAC address	Pass the frame to DLL management. Stop the broadcast through BRT	
7			Broadcast address	Pass the frame to DLL management. Stop the broadcast through BRT	
8			Reserved address	Pass the frame to DLL management Stop the broadcast through BRT	Management message intended to all nodes
9			Destination MAC address not covered by cases 4-8	Drop the frame	
10		Non-Leaf	In LAAT except node's MAC address	Forward through BRT (Note 1)	Not applicable

Table 8-14.2 – Broadcast of LLC frames (LCDU case)

Case	Type of broadcast	Leaf/ Non-leaf	LCDU DA	Broadcasting actions	Example
11			In RAAT	Forward through BRT (NOTE – RAAT is not consulted)	
12			node's MAC address	Pass the frame to DLL management; Optional: forward through BRT	
13			Broadcast address	Pass the frame to DLL management and forward through BRT	
14			Reserved address	Pass the frame to DLL management and forward through BRT	Management message intended to all nodes
15			Destination MAC address not covered by cases 10--14	Forward through BRT	
NOTE 1 – LAAT is not consulted. NOTE 2 – Following BRT rules leads to a drop.					

Table 8-14.3 – Broadcast of LLC frames (APDU case)

Case	Type of broadcast	Leaf/Non-leaf	APDU DA	Broadcasting actions	Example
16	Broadcast frame intended for another node in the network (BRCTI = 1; MCSTI = 0; DestinationNode = $DeviceID_{OtherNode}$)	Leaf	–	Drop the frame	Unicast frame not found by a previous relay node in its internal unicast routing tables and relayed in broadcast
17		Non-Leaf	–	Follow BRT rules. Apply filtering	Unicast frame not found by a previous relay node in its internal unicast routing tables and relayed in broadcast
18	Broadcast frame intended for this node (BRCTI = 1; MCSTI = 0; DestinationNode = $DeviceID_{Node}$)	Leaf	–	Consider the frame as non-broadcast (unicast) and follow the corresponding rules (cases 1-6 of Table 8-14.5). Stop the broadcasting	End point of a unicast frame not found by a previous relay node in its internal unicast routing tables and relayed in broadcast

Table 8-14.3 – Broadcast of LLC frames (APDU case)

Case	Type of broadcast	Leaf/Non-leaf	APDU DA	Broadcasting actions	Example
19		Non-Leaf		Consider the frame as non-broadcast (unicast) and follow the corresponding rules (cases 1-6 of Table 8-14.5). Stop the broadcast	End point of a unicast frame not found by a previous relay node in its internal unicast routing tables and relayed in broadcast
20	Broadcast frame intended to all the nodes (BRCTI = 1; MCSTI = 0; DestinationNode = BroadcastID)	Leaf	In LAAT except node's MAC address	Pass to A interface; Do not relay (Note 1)	
21			In RAAT	Do not relay (Note 2) Optional : Pass to A interface. ~~Do not relay (Note 2)~~	
22			Node's MAC address	Pass to DLL management. Stop the broadcast through BRT	
23			Broadcast address	Pass to A interface; Pass to DLL management. Stop the broadcast through BRT	Standard broadcast
24			Reserved address	Pass to DLL management. Stop the broadcast through BRT	
25			Destination MAC address not covered by cases 20-24	Pass to A-interface. Do not relay	Unknown Destination frames broadcast
26		Non-Leaf	In LAAT except node's MAC address	Pass to A interface Optional: relay following BRT	
27			In RAAT	Relay following BRT Optional: Pass to A-interface ~~Relay following BRT~~	
28			Node's MAC address	Pass to DLL management Optional: relay following BRT	

Table 8-14.3 – Broadcast of LLC frames (APDU case)

Case	Type of broadcast	Leaf/Non-leaf	APDU DA	Broadcasting actions	Example
29			Broadcast address	Pass to A interface; Pass to DLL management; Relay through BRT	
30			Reserved address	Pass to DLL Management. Relay through BRT	
31			Destination MAC address not covered by cases 26-30	Pass to A interface. Relay through BRT	Unknown Destination frames broadcast
NOTE 1 – LAAT is not consulted. NOTE 2 – RAAT is not consulted.					

The nodes relaying a broadcast message shall associate this message with the same priority as assigned by the sourcing node (communicated in the LPRI field of LFH).

11) Clause 8.5.7.1, Relaying of LCDU

Revise the text of clause 8.5.7.1 "Relaying of LCDU" as follows:

8.5.7.1 Relaying of LCDU

Any LLC frame received from the medium that contains an LCDU shall be relayed according to the following rules:

If the frame has been received with BRCTI = 0, MCSTI = 0 and the DestinationNode is the DEVICE_ID of the receiving node, the node shall extract the LCDU and pass it to the DLL management (cases 1, 2, 3, 4, 5 and 6 of Table 8-14.4).

If the frame has been received with BRCTI = 0, MCSTI = 0 and the DestinationNode is zero, the node shall:

a) If the DA is the same as the address of the DLL management of the node, or the standard broadcast address, or the reserved MAC address 01-19-A7-52-76-96 (cases 9, 10 and 11 of Table 8-14.4), it shall extract the LCDU and pass it to the DLL management. The action taken by the DLL management entity depends on the contents of the LCDU and the role of the node in the domain.

b) In all other cases (cases 7, 8 and 12 of Table 8-14.4), the node shall drop the LLC frame.

If the frame has been received with BRCTI = 0, MCSTI = 0 and the DestinationNode is present in the unicast routing tables (case 13 of Table 8-14.4), the node shall relay it to the appropriate node or nodes as indicated in the routing table.

If the frame has been received with BRCTI = 0, MCSTI = 0 and the DestinationNode is not present in the unicast routing tables:

- If the receiving node is a leaf node (case 14 of Table 8-14.4), the frame is dropped.
- If the receiving node is a non-leaf node (case 15 of Table 8-14.4), the node shall set the BRCTI to 1 and broadcast the received LLC frame to the nodes that are specified in the BRT while keeping the DestinationNode and OriginatingNode fields.

If the frame has been received with BRCTI = 0, MCSTI = 0 and the DestinationNode is not present in the unicast routing tables:

keep DestinationNode and OriginatingNode; set BRCTI = 1 and route using the BRT.

If the frame has been received with BRCTI = 0, MCSTI = 1 and with a known MSID (cases 16 and 18 of Table 8-14.4), the frame shall be relayed as specified in clause 8.17. In addition, if the node is a member of the specified MSID DLL multicast group (case 18 of Table 8-14.4), the node shall extract the LCDU and pass it to the DLL management. If the frame has been received with an unknown MSID (case 17 of Table 8-14.4), the frame shall be dropped and the transmitter shall be informed as specified in clause 8.17.

If the frame has been received with BRCTI = 1, the frame is processed as specified in clause 8.5.4.

Table 8-14.4 – Relaying of LLC frames (LCDU case)

Case	Type of relaying	Leaf/Non-leaf	LCDU DA	Relaying actions	Example
1	Unicast frame intended to the node (BRCTI = 0; MCSTI = 0; Destination Node = $DeviceID_{Node}$)	Leaf/Non-leaf	In LAAT except node's MAC address	Send to DLL management	
2			In RAAT	Send to DLL management	Management message with proxy (ADM_NodeRegisterRequest.req)
3			Node's MAC address	Send to DLL management	Management message (or remote in-band message)
4			Broadcast address	Send to DLL management	
5			Reserved address	Send to DLL management	
6			Destination MAC address not covered by cases 1-5	Send to DLL management	
7	Unicast frame with Destination Node = 0 (BRCTI = 0; MCSTI = 0; Destination Node = 0)	Leaf/Non-leaf	In LAAT except node's MAC address	Drop the frame	Not applicable
8			In RAAT	Drop the frame	Not applicable
9			node's MAC address	Send to DLL management	ADM_DMRegistrResponse.cnf (with/without proxy)
10			Broadcast address	Send to DLL management	Not applicable
11			Reserved address	Send to DLL management	Not applicable
12			Destination MAC address not covered by cases 7-11	Drop	

Table 8-14.4 – Relaying of LLC frames (LCDU case)

Case	Type of relaying	Leaf/Non-leaf	LCDU DA	Relaying actions	Example
13	Unicast frame not intended to the node but with known Destination Node (BRCTI = 0; MCSTI = 0; Destination Node = $DeviceID_{OtherNode}$)	–	–	Use unicast routing tables	Normal "relaying"
14	Unicast frame not intended to the node but with an unknown Destination Node (BRCTI = 0; MCSTI = 0; Destination Node = Unknown)	Leaf	–	Drop the frame (Note 1)	During transient periods
15		Non-Leaf	–	Broadcast the unicast frame (Note 2)	During transient periods
16	Multicast frame where the relay node does not belong to the group (BRCTI = 0; MCSTI = 1; Destination Node = MSID) $DeviceID_{Node} \notin$ MSID group	–	–	Follow multicast rules to forward the frame through the DLL multicast tree	
17	Multicast frame where the relay node does not know the MSID (BRCTI = 0; MCSTI = 1 Destination Node = MSID) Unknown MSID	–	–	Drop the frame	
18	Multicast frame where the relay node belongs to the group (BRCTI = 0; MCSTI = 1 Destination Node = MSID) $DeviceID_{Node} \in$ MSID group	–	–	Apply same rules than unicast frame intended to the node (cases 1-6 of Table 8-14.4) and follow the multicast rules to forward the frame through the multicast tree	

NOTE 1 – Follow the BRT rules, leading to a drop.
NOTE 2 – Keep DestinationNode and OriginatingNode; set BRCTI to = 1 and route using the BRT.

12) Clause 8.5.7.2, Relaying of APDU

Revise the text of clause 8.5.7.2 "Relaying of APDU" as follows:

8.5.7.2 Relaying of APDU

Any LLC frame received from the medium that contains an APDU shall be relayed according to the following rules:

If the frame has been received with BRCTI = 0, MCSTI = 0 and the DestinationNode is the DEVICE_ID of the receiving node, the node shall extract the APDU and:

a) If the DA is the same as the MAC address of the node or the standard broadcast address or the reserved MAC address 01-19-A7-52-76-96 (i.e., cases 3, 4 and 5 of Table 8-14.5), the node shall pass it to the DLL management.

b) If the DA is found in the LAAT of the node (i.e., cases 1 of Table 8-14.5), the node shall pass it to the A-interface.

c) If the DA is found in the RAAT of the node (i.e., cases 2 of Table 8-14.5), the node may pass it to the A-interface.

d) If the DA does not correspond to any of the cases a), b) or c) (i.e., case 6 of Table 8-14.5), the node shall send it through the A-interface.

If the frame has been received with BRCTI = 0, MCSTI = 0 and the DestinationNode is not the DEVICE_ID of the receiving node and it is found in the unicast routing tables (case 7 of Table 8-14.5), the node shall relay it to the appropriate node or nodes as indicated in the routing table.

If the frame has been received with BRCTI = 0, MCSTI = 0 and the DestinationNode is not the DEVICE_ID of the receiving node and it is not found in the unicast routing tables:

- If the receiving node is a leaf node (cases 8 and 10 of Table 8-14.5), the node shall drop the frame.
- If the receiving node is a non-leaf node (cases 9 and 11 of Table 8-14.5), the node shall set the BRCTI to 1 and broadcast the received LLC frame to the nodes that are specified in the BRT while keeping the DestinationNode and OriginatingNode.

If the frame has been received with BRCTI = 0, MCSTI = 1 and with a known MSID (cases 12 and 14 of Table 8-14.5), the frame shall be relayed as specified in clause 8.17. In addition, if the node is a member of the DLL multicast stream (case 14 of Table 8-14.5), the node shall extract the APDU and follow the same rules as the case where the frame has been received with BRCTI = 0, MCSTI = 0 and the DestinationNode is the DEVICE_ID of the receiving node.

If the frame has been received with an unknown MSID (case 13 of Table 8-14.5), the frame shall be dropped and the transmitter shall be informed as specified in clause 8.17.

If the frame has been received with BRCTI = 1, the frame is processed as specified in clause 8.5.4

Table 8-14.5 – Relaying of LLC frames (APDU case)

Case	Type of relaying	Leaf/Non-leaf	APDU DA	Relaying action	Example
1	Unicast frame intended to the node (BRCTI = 0; MCSTI =	Leaf/Non-leaf	In LAAT except node's MAC address	Send through A interface	Normal data frame

Table 8-14.5 – Relaying of LLC frames (APDU case)

Case	Type of relaying	Leaf/Non-leaf	APDU DA	Relaying action	Example
2	0; Destination Node = $DeviceID_{Node}$)		In RAAT	Optional to send frame through A interface	Handover of equipments between different ITU-T G.9960 nodes
3			Node's MAC address	Send to DLL management	firmware upgrade, ping, etc.
4			Broadcast address	Send to DLL management	
5			Reserved address	Send to DLL management	
6			Destination MAC address not covered by cases 1-5	Send through A interface	Can happen in a corner case (e.g., ageing), or for Multicast frames transmitted using DLL unicast
7	Unicast frame not intended to the node but with known Destination Node (BRCTI = 0; MCSTI = 0; Destination Node = $DeviceID_{OtherNode}$)	–	–	Use unicast routing tables	Normal "relaying"
8	Unicast frame with Destination Node = 0 (BRCTI = 0; MCSTI = 0; Destination Node = 0)	Leaf	–	Drop the frame	During transient periods
9		Non-Leaf	–	Broadcast the unicast frame (Note)	During transient periods
10	Unicast frame not intended to the node but with unknown Destination Node (BRCTI = 0; MCSTI = 0; Destination Node = Unknown)	Leaf	–	Drop the frame	During transient periods
11		Non-Leaf	–	Broadcast the unicast frame (Note)	During transient periods
12	Multicast frame where the relay node does not belong to the group (BRCTI = 0; MCSTI = 1; Destination Node = MSID)	–	–	Follow multicast rules to forward the frame through the multicast tree	

Table 8-14.5 – Relaying of LLC frames (APDU case)

Case	Type of relaying	Leaf/Non-leaf	APDU DA	Relaying action	Example
	$DeviceID_{Node} \notin$ MSID group				
13	Multicast frame where the relay node does not know the group (BRCTI = 0; MCSTI = 1; Destination Node = MSID) Unknown MSID	–	–	Drop the frame	
14	Multicast frame where the relay node belongs to the group (BRCTI = 0; MCSTI = 1; Destination Node = MSID) $DeviceID_{Node} \in$ MSID group	–	–	Send to A interface and if it a relay node for this MSID then forward it to the proper nodes	
NOTE – Keep DestinationNode and OriginatingNode; set BRCTI to 1 and route using the BRT.					

13) Clause 8.6, Domain master node functional capabilities

Revise the text of clause 8.6 "Domain master node functional capabilities" as follows:

8.6 Domain master node functional capabilities

A domain master-capable node is a node that, in addition to supporting all of the required capabilities of an endpoint node, is also able to assume the role of a domain master.

A domain master-capable node shall support all of the functions specified in the following clauses.

At any given time, only one node is allowed to act as a domain master for a domain. All other nodes within the domain are managed (coordinated) by this domain master. If a domain master fails, another node of the same domain, capable of operating as a domain master, should pick up the function of the domain master.

The domain master shall perform medium access using the same medium access rules as for endpoint (non-domain master) nodes and using the same MAP distributed to the endpoint nodes.

NOTE – It is not a requirement that every node be domain master capable.

The DM is responsible for communicating the latest versioning information and capabilities supported by the nodes of the domain to all the nodes of the domain. The details of how this is done are described in clause 8.18.

8.6.1 Network admission

To join the network, all nodes shall first "register" with the domain master using the network admission protocol described in clause 8.6.1.1.

Normally, non registered nodes are able to receive successfully the MAP frames only if the MAP is transmitted in the default MAP format (MAP-D). Therefore, the domain master shall transmit periodically MAP-D messages in addition to MAP-A transmission to enable registration.

If a node does not have direct communication with the domain master (i.e., is hidden from the domain master), this node can still register and become part of the network using relayed admission as described in clause 8.6.1.2.

For registration, a unique registration identifier (REGID) is assigned to every node prior to its installation. REGID is intended exclusively for registration and may be communicated unencrypted. The value of the REGID shall be equal to the MAC address of the node.

The registering node shall identify the domain it wishes to join by comparing the domain name information in the received MAP-D frames as described in clause 8.8.3, with the target domain name(s) provided to the node by the user (to distinguish his/her network from neighbouring networks) or obtained during the first registration, if a device has no user interface.

The registering node shall first search for a MAP frame bearing a DNI field whose value coincides with the value of a target DNI in its information database. When a MAP frame meeting the target DNI is detected, the node shall verify the full value of the domain name in the Domain Name field of the MAP (see clause 8.8.5.2) and use the DOD value of this MAP frame as the DOD for its registration messages described in clause 8.6.1.1.4 to indicate the particular domain it intends to join.

If the domain operates in non-secure mode, a node which successfully registered with the domain master can communicate with other nodes in the domain. If the domain operates in a secure mode, a registered node shall also authenticate itself, as described in clause 9.2. After authentication, the node becomes a member of the secure network and is in a position to establish communication with any other node in the domain/network.

In case a ~~device~~ node has no ~~user~~ interface for configuring a target domain(s), the manufacturer shall provide the ~~device~~ node with a 6-byte registration code, which is also supplied explicitly to the user. Such a ~~device~~ node provided with a registration code shall search for a MAP bearing this registration code in the auxiliary information field (see clause 8.8.5.9). After registration, the ~~device~~ node shall memorize (optimally to a non-volatile memory) the domain name communicated in the MAP and the value of DNI, and use it as a target DNI for future registrations.

In case a node has an interface for configuring a target domain(s), it shall be configured with ~~The list of~~the target domain(s) ~~(configured by the user)~~ . The list may include more than one entry. If a node fails to register to one domain ~~from~~ on the list, it ~~may~~ shall try to register to another one on the list ~~(~~if more than one domain was detected~~), until the node is either successfully admitted to one of the target domains or runs out of the list~~.

If no MAP frame meeting the target DNI is found, ~~the~~ a node that is not capable of acting as a domain master may continue searching for the target DNI. A node that is capable of acting as a domain master shall establish a new domain, as described in clause 8.6.6.

The DEVICE_ID of the registering node shall be set to zero. After registration is complete, the DEVICE_ID shall be set to the value assigned by the domain master, as described in clause 8.7.1.1. A node shall not establish connections until it has been assigned a DEVICE_ID. From the first transmitted frame, the node shall comply with the transmission schedule posted in the MAP and shall meet all spectral compatibility requirements described in the PSD-related domain info field of the MAP (SM, PSD mask, Transmission power limit, etc. – See clause 8.8.5.5).

8.6.1.1 Network admission protocol

8.6.1.1.1 Registration into the domain

The protocol diagram of node registration into the domain is presented in Figure 8-25.

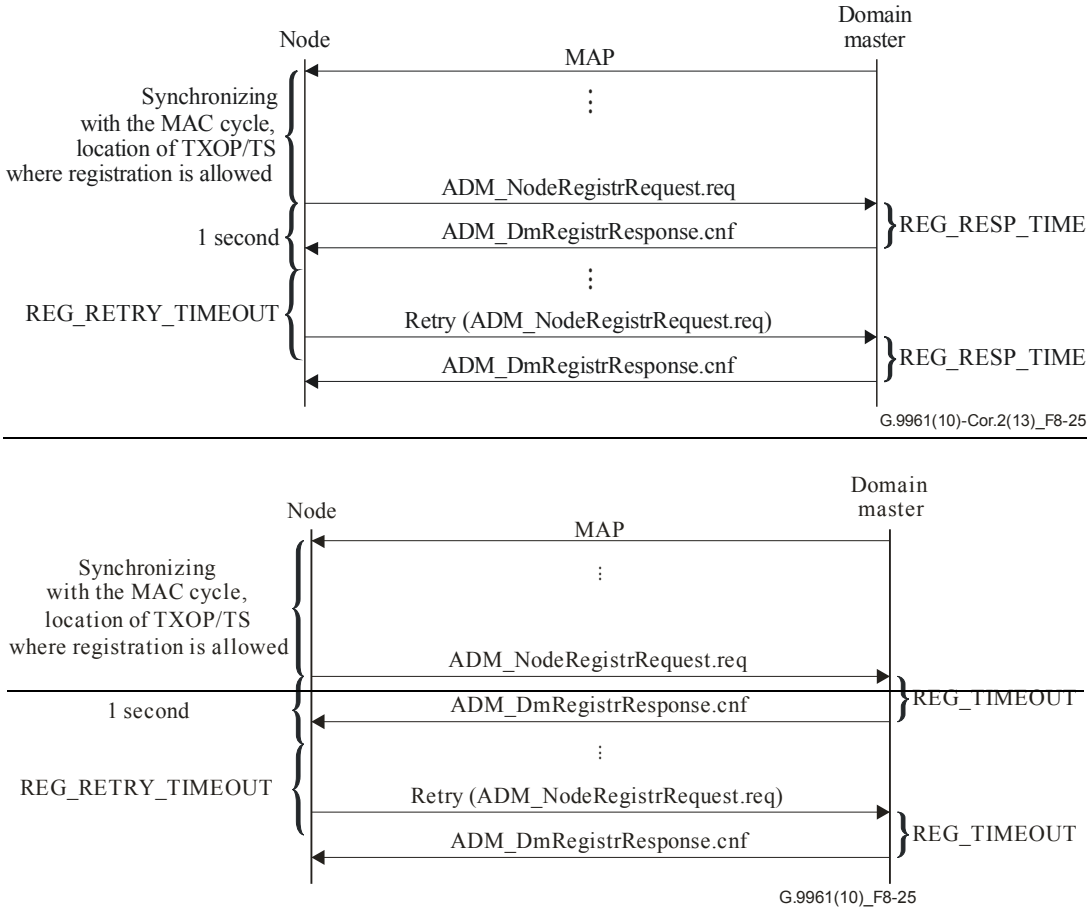


Figure 8-25 – Protocol diagram describing node registration

Prior to registration, the node shall synchronize with the network so that it can identify the MAC cycle, detect the MAP of the domain it intends to register, and locate the registration TS (for RCBTS see clause 8.3, for CBTS that permits registration, see clauses 8.8.4.1.5 and 8.8.4.2).

To start the registration, the node shall send a registration request (ADM_NodeRegistrRequest.req) message to the domain master, which includes the REGID (the source MAC address of the LCDU header, see Figure 8-6) and other registration related parameters, as described in clause 8.6.1.1.4.1. The registering node is allowed to send ADM_NodeRegistrRequest.req during the RCBTS or during any other CBTS for which registration is allowed, using medium access rules for CBTS described in clause 8.3.3.4 with the MA priority associated with MPDU priority = 7.

The domain master shall process the registration request and shall reply within REG_RESP_TIME~~OUT~~ to the node with a registration response (ADM_DmRegistrResponse.cnf) message, which includes a status flag that indicates whether the domain master admitted the node to the domain or not. If the node is admitted, the ADM_DmRegistrResponse.cnf message shall contain a non-zero DEVICE_ID for the registering node assigned by the domain master and relevant configuration data. If the domain master rejects the admission, the ADM_DmRegistrResponse.cnf message shall contain a rejection code, describing the reason of rejection (see Table 8-15) and assigned DEVICE_ID = 0. The details of the ADM_DmRegistrResponse.cnf message are described in clause 8.6.1.1.4.2. The DID in the header of the PHY frame containing the ADM_DmRegistrResponse.cnf message shall be set to zero. The DestinationNode of the LLC frame containing the ADM_DmRegistrResponse.cnf message shall be set to zero.

Registering nodes shall identify the ADM_DmRegistrResponse.cnf message based on its REGID field. The ADM_DmRegistrResponse.cnf message may be sent during the dedicated TSs or TXOPs, if assigned by the domain master, or during any CBTS, using medium access rules for

CBTS described in clause 8.3.3.4 with the MA priority associated with MPDU priority = 7. If the registering node does not receive an ADM_DmRegistrResponse.cnf message from the domain master within one second, the node shall retry registration within REG_RETRY_TIMEOUT. If the registering node does not receive a response after MAX_REG_ATTEMPTS registration attempts, the node shall not continue registration attempts. If the registering node was rejected by the domain master, depending on the rejection code, the node may either retry registration during REG_RETRY_TIMEOUT or shall stop registration attempts. Valid admission rejection codes are presented in Table 8-15.

Table 8-15 – Admission rejection codes

Rejection code (Note 1)	Reason	Retry allowed
000	Unspecified	Yes
001	Insufficient bandwidth resources	Yes
010	Invalid set of registration parameters	No
011	Invalid REGID	No
100	Admission limit expired	Note 2
101	DM not authenticated	Yes
110	Node's reported bandplan is outside the range indicated by the minimal and maximal bandplan ranges allowed in the domain	Yes
NOTE 1 – Other values reserved by ITU-T. NOTE 2 – Retry procedure in case of admission limit expired is for further study.		

Rejection codes associated with "Retry not allowed" requires re-configuration of the node, which includes modification of at least one of registration related parameters. After re-configuration, the node can attempt a new registration.

The domain master shall decide on the admission of the registering node based on the information supplied in ADM_NodeRegistrRequest.req message and the current status of the domain, evaluated by the domain master. The evaluation rules are vendor discretionary. The domain master may then assign resources to the registered node.

8.6.1.1.2 Periodic re-registrations

A node that is not in idle mode (L3) shall re-register with the domain master within the time period indicated in the MAP message (see Table 8-82) after registration (receiving the last ADM_DmRegistrResponse.cnf message) or re-registration (receiving the last ADM_DmReRegistrResponse.cnf message). Re-registration shall use the ~~same message exchange protocol as for registration with~~ ADM_NodeReRegistrRequest.req and ADM_DmReRegistrResponse.cnf message format as described in clause 8.6.1.1.4.

For re-registration, the node shall transmit ADM_NodeReRegistrRequest.req message, with format as described in clause 8.6.1.1.4.6~~1~~, during any of its available TXOP or TS, but not during RCBTS. The domain master recognises a re-registering node by its REGID. The domain master shall reply to the node by sending an ADM_DmReRegistrResponse.cnf message ~~during the dedicated TSs or TXOPs, if assigned by the domain master, or during any CBTS in which the node is allowed to transmit using medium access rules for CBTS described in clause 8.3.3.4~~ with the MA priority associated with MPDU priority = 7. Unlike registration messages, re-registration messages shall be transmitted using the connection (either a management connection or a prioritized data connection) for delivering LCDUs.

The domain master may force resignation from the domain of all nodes that failed periodic re-registration for two consecutive times using the procedure described in clause 8.6.1.1.3.2. The domain master shall cancel all bandwidth resources associated with the resigned nodes.

~~The~~ A resigned node that wishes to register or a node that gets reset ~~may~~ that wishes to register again, shall us~~ing~~ the standard registration procedure, ~~starting from the first available CBTS for which registration is allowed,~~ not the re-registration procedure.

If the domain master receives a registration request instead of a re-registration request from its node (e.g., the node gets reset during re-registration period), the domain master shall request the node to initiate re-registration immediately by sending ADM_DmReRegistrInitiate.ind.

NOTE – This is to ensure that it is a legitimate registration request.

If the node does not respond with ADM_NodeReRegistration.req within 200 ms, the domain master shall send ADM_DmReRegistrInitiate.ind one more time. If the node does not respond with ADM_NodeReRegistration.req within 200 ms, the domain master shall consider the node no longer present and follow the node removal process specified in clause 8.6.1.1.3.2, and then respond to the new registration request from the registering node.

Re-registration of nodes in idle mode (L3) is for further study.

…

8.6.1.1.4.1 Registration request message (ADM_NodeRegistrRequest.req)

The ADM_NodeRegistrRequest.req message is a unicast management message sent by a registering node to the domain master (directly or via a proxy), and is intended to be used for registration ~~and periodical re-registration~~ requests only. The format of the MMPL of the ADM_NodeRegistrRequest.req message shall be as shown in Table 8-16.

Table 8-16 – Format of the MMPL of the ADM_NodeRegistrRequest.req message

Field	Octet	Bits	Description
Attempt	0	[1:0]	00_2 for initial attempt, 01_2, 10_2, 11_2 for the second, third and fourth attempts
ProxyReg		[2]	Proxy registration flag; shall be set to one for registration through proxy (see clause 8.6.1.2) and zero otherwise
Reserved		[7:3]	Reserved by ITU-T (Note 1)
ProxyDevID	1	[7:0]	Device ID of the Registration proxy (Note 2)
Parameters	2	[0]	Set to one if node is capable of operating as a domain master, zero otherwise
		[1]	Set to one if relaying is supported, zero otherwise
		[4:2]	Indicates the bandplan that the node shall use after registration represented as described in clause 7.1.2.3.2.2 of [ITU-T G.9960] (BNDPL/GRP_ID field)
		[5]	Set to one if the device is registering using registration code, zero otherwise
		[7:6]	Reserved by ITU-T (Note 1)
T_AIFG	3	[7:0]	The value of T_{AIFG} supported by the node, represented as n × 1.28 μs; the value of n is an unsigned integer in the range between 4 and 96. Valid values for each medium are specified in Table 8-14

Table 8-16 – Format of the MMPL of the ADM_NodeRegistrRequest.req message

Field	Octet	Bits	Description
NumNodeVersionTLVs	4	[7:0]	Number of versioning (N) TLVs included in this message. Set to 0 if no Versioning TLVs are included which implies that the ~~0 –~~ ~~N~~node only supports version 0 of ITU-T G.9960 and ITU-T G.9961. If N > 0, the first TLV shall be the TLV corresponding to ITU-T version ~~All other values of this field are reserved by ITU-T for indicating support for future versions of the Recommendation. (Note 3)~~
Parameters	5	[0]	Set to one if node is capable of calculating routing tables, zero otherwise
		[7:1]	Reserved by ITU-T (Note 1)
NodeVersionTLVs	Var	Var	Information related to the version and capabilities of the registering node. It shall be coded as described in Table 8-16.1
NOTE 1 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver. NOTE 2 – This field shall be set to zero by the transmitter and ignored by the receiver when the ProxyReg field is set to zero. ~~NOTE 3 – A node indicating support for a certain version of this Recommendation shall also support all earlier versions of this Recommendation.~~			

Table 8-16.1 – Format of NodeVersionTLVs[i]

Field	Octet	Bits	Description
VersioningType	Var	[7:0]	Type of versioning field 0 – ITU-T Versioning information (see Table 8-16.2). 1 – Reserved for HGF Versioning information All other values of this field are reserved by ITU-T for versioning information
VersioningLength	Var	[7:0]	Length in bytes of VersioningValue field
VersioningValue	Var	Var	Value of Versioning field

Table 8-16.2 – Format of the VersioningValue field for ITU-T VersioningType

Field	Octet	Bits	Description
ITUFieldsContents	0	[0-7]	Bits [7:0] represent the information related to the presence of the different components of the ITUVersioning field. The sequence of the fields shall be from LSB to MSB Bit 0: If set to one, AmdVersioning field is present. If set to zero, AmdVersioningField is not present Bit 1: If set to one, Capabilities field is present. If set to zero, Capabilities field is not present Other bits are reserved by ITU-T and shall be set to 0

Table 8-16.2 – Format of the VersioningValue field for ITU-T VersioningType

Field	Octet	Bits	Description
			(Note 1)
AmdVersioning	Variable	See Table 8-16.3	If present, this field contains information on the amendment of the ITU-T Recommendation that the reporting node supports (Note 2). The format of this field is described in Table 8-16.3
Capabilities	Variable	See Table 8-16.4	If present, this field contains the information on specific capabilities that the node implements. The format of this field is described in Table 8-16.4
NOTE 1 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver. NOTE 2 – A node indicating support for a certain amendment of a Recommendation shall also support all earlier amendments of that Recommendation.			

Table 8-16.3 – Format of the AmdVersioning Field

Field	Octet	Bits	Description
AmendmentListLength	0	[7:0]	Number of Recommendations (N) indicated in AmdVersioning field
RecommendationType[0]	1	[7:0]	Recommendation type 0 – ITU-T G.9960 1 – ITU-T G.9961 2 – ITU-T G.9962 3 – ITU-T G.9963 4 – ITU-T G.9964 Other values are reserved by ITU-T for future Recommendations
RecommendationVersion[0]	2	[7:0]	Amendment version of the indicated Recommendation that this node supports, represented as an 8-bit unsigned integer. Value 0 corresponds to the base Recommendation
…			
RecommendationType[N-1]	Variable	[7:0]	Recommendation type 0 – ITU-T G.9960 1 – ITU-T G.9961 2 – ITU-T G.9962 3 – ITU-T G.9963 4 – ITU-T G.9964 Other values are reserved by ITU-T for future Recommendations
RecommendationVersion[N-1]	Variable	[7:0]	Amendment version of the indicated Recommendation that this node supports, represented as an 8-bit unsigned integer. Value 0 corresponds to the base Recommendation

Table 8-16.4 – Format of the Capabilities Field

Field	Octet	Bits	Description
NumCapabilities	0	[7:0]	Number of capabilities (M) indicated in Capabilities field
CapabilityType[0]	1	[7:0]	Capability type. The format of this field is described in Table 8-16.5
CapabilityLength[0]	2	[7:0]	Length of the capability type indicated, represented as an 8-bit unsigned integer (see Table 8-16.5)
CapabilityValue[0]	Variable	Variable	Value of the capability type indicated
…			
CapabilityType[M-1]	1	[7:0]	Capability type. The format of this field is described in Table 8-16.5
CapabilityLength[M-1]	2	[7:0]	Length of the capability type indicated, represented as an 8-bit unsigned integer (see Table 8-16.5)
CapabilityValue[M-1]	Variable	Variable	Value of the capability type indicated

Table 8-16.5 – List of Capabilities

Capability Type	Capability Name	Capability Length Value	Capability Value field
00_{16}	Bandplan Info	4	See Table 8-16.6
01_{16}-FF_{16}	Reserved by ITU-T		Reserved by ITU-T

Table 8-16.6 – Bandplan Info Capability Value field

Field	Octet	Bits	Description
Bandplan ID	0	[2:0]	Indicates the maximum bandplan that the node supports (Note 2), represented as described in clause 7.1.2.3.2.2 of [ITU-T G.9960]
Reserved		[7:3]	Reserved by ITU-T (Note 1)
StartSubCarrier	1 to 3	[11:0]	Index of the lowest frequency sub-carrier that the node can support on the transmit side coded as an unsigned integer
StopSubCarrier		[23:12]	Index of the highest frequency sub-carrier that the node can support on the transmit side coded as an unsigned integer
NOTE 1 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver. NOTE 2 – The Bandplan ID field needs to be equal to the same field in the old message structure (if it exists).			

…

8.6.1.1.4.6 Re-registration request message (ADM_NodeReRegistrRequest.req)

The ADM_NodeReRegistrRequest.req message is a unicast management message sent by a node to the domain master, and is intended to be used for periodical re-registration requests.

The MMPL of the ADM_NodeReRegistrRequest.req message shall be empty.

8.6.1.1.4.7 Re-registration response message (ADM_DmReRegistrResponse.cnf)

The ADM_DmReRegistrResponse.cnf message is a unicast management message sent by the domain master to the node, and is intended to be used for periodical re-registration response. The format of the MMPL of the ADM_DmReRegistrResponse.cnf message shall be empty.

8.6.1.1.4.8 Re-registration initiation message (ADM_DmReRegistrInitiate.ind)

The ADM_DmReRegistrInitiate.ind message is a unicast management message sent by the domain master to a node, and is intended to force the node to initiate re-registration process immediately. The format of the MMPL of the ADM_DmReRegistrInitiate.ind message shall be as shown in Table 8-20.1.

Table 8-20.1 – Format of the MMPL of the ADM_DmReRegistrInitiate.ind message

Field	Octet	Bits	Description
Attempt	0	[1:0]	00_2 for the initial attempt 01_2 for the second attempt 10_2, 11_2 – Reserved by ITU-T
Reserved		[7:2]	Reserved by ITU-T (Note)
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

14) Clause 8.6.2.1, Description of TSpec parameters

Revise the text of clause 8.6.2.1 "Description of TSpec parameters" as follows:

8.6.2.1 Description of TSpec parameters

Terms related to traffic specifications and quality of service are described in this clause.

…

Maximum latency – The value of this parameter specifies the maximum interval in ms between the ~~entry~~ arrival time of an ~~packet~~ ADP at the A-interface ~~APC~~ of the originating node and the ~~forwarding~~ departure time of the ADP at the A-interface of the endpoint~~APDU to its destined~~ node. If defined, this parameter represents a flow commitment (or admission criteria) at the domain master and the involved nodes and shall be guaranteed by the domain master and the nodes. The domain master and the involved nodes do not have to meet this flow commitment for flows that exceed their committed information rate.

…

15) Clause 8.6.2.2.3, Flow maintenance

Revise the text of clause 8.6.2.2.3 "Flow maintenance" as follows:

8.6.2.2.3 Flow maintenance

…

Once a service flow has been established, it shall be maintained by the originating node and by the domain master to fulfil the TSpec contract using the following rules:

- When the bit loading employed between the two endpoints is reduced to such an extent that the bandwidth to support the agreed-upon traffic contract between the originating node and the domain master is insufficient, the originating node shall inform the domain master that it is being provided with insufficient bandwidth in its current CFTXOP by sending the FL_ModifyFlowParameters.ind message and by setting the number of bytes that shall be

transmitted in the BRURQ field (see 7.1.2.3.2.2.19 of [ITU-T G.9960]) conveyed in the transmitted message PFH.

- When the bit loading employed between the two endpoints is increased, such that the flow begins to consume only a small fraction of the bandwidth allocated in the CFTXOP, the originating node shall inform the domain master of the situation by sending the FL_ModifyFlowParameters.ind message. If the domain master infers from inspection of a BRURQ field conveyed in MSG frame PHY-frame header or by receiving FL_ModifyFlowParameters.ind message with indication that the duration of the CFTXOP may be reduced while still complying with the terms of the traffic contract, it shall decrease the CFTXOP allocations accordingly.
- When there are user data traffic flows that are characterized by fixed packet size and fixed intervals between packets arriving via the A-interface, the node may adjust the allocations of the CFTXOP in a MAC cycle for this type of traffic using the FL_ModifyFlowAllocations.req~~cnf~~ message.

…

16) Clause 8.6.2.3.1, Format of CL_EstablishFlow.req

Revise the text of clause 8.6.2.3.1, "Format of CL_EstablishFlow.req" as follows:

8.6.2.3.1 Format of CL_EstablishFlow.req

This message is sent by the application entity residing on the client associated with a node. This message contains the following parameters: flow destination MAC address, the flow classifiers, the flow TSpec and the bidirectional indication. In case the bidirectional indication is set, the following fields for the flow in the reverse direction shall be included in the message as well: the destination address, the Tspec and classifiers for the reverse direction. The format of the MMPL of the CL_EstablishFlow.req message shall be as shown in Table 8-21.

Table 8-21 – Format of the MMPL of the CL_EstablishFlow.req message

Field	Octet	Bits	Description
DA	0 to 5	[47:0]	Flow Destination MAC address. APDUs whose destination MAC address is specified in this field should be transmitted via this flow
Classifiers	6 to (7+j)	See Table 8-22	This field shall contain traffic classifiers. APDUs whose destination MAC address is the specified MAC address and header conforms to the specified classifiers should be transmitted via this flow
TSpec	variable	Table 8-24	Traffic specification for this flow may include the following fields: Traffic Priority, Maximum information Rate, Maximum Traffic Burst, committed information rate, Tolerated Jitter, Maximum Latency, Unsolicited Grant Interval, Unsolicited Polling Interval and APDU Size N – The length of this field is variable according to the actual number of included traffic specification fields. The TSpec format is as specified in Table 8-24
Bidirectional	variable	[7:0]	When set to 01_{16} this field indicates that the flow is a bidirectional flow When set to 00_{16} this field indicates that the flow is a unidirectional flow

Table 8-21 – Format of the MMPL of the CL_EstablishFlow.req message

Field	Octet	Bits	Description
DA_B	variable	[47:0]	Destination MAC address for the established flow in the reverse direction (Note)
TSpec_B	variable	Table 8-24	Contains the TSpec of the flow in the reverse direction (Note)
Classifiers_B	variable	Table 8-22	Contains the traffic classifiers used to classify APDUs to be transmitted in the reverse direction (Note)
NOTE – These fields shall only exist in the message if bidirectional field is set to 01_{16}.			

Table 8-22 – Format of classifiers structure

Field	Octet	Bits	Description
Length	0	[7:0]	Length of the list of classifiers (j) in bytes
Num	1	[7:0]	Number of classifiers (k) in the classifiers list
Classifier[0]	2 to (m+3)	See Table 8-23	First classifier in the list. Format of the classifiers is specified in Table 8-23. m+2 is the classifier length (Note)
…	…	…	…
Classifier[k-1]	variable	See Table 8-23	Last classifier in the list (Note).
NOTE – More than one classifier can carry the same classifier type with different values. For example, Classifier[0] = "(", Classifier[1] = Address X, Classifier[2] = "AND, Classifier[3] = Destination Port 0, Classifier[4] =), Classifier[5] = "OR", Classifier[6] = "(", Classifier[7] = Address Y, Classifier[8] = "AND", Classifier[9] = Destination Port 1, Classifier[10] = ")" implies that any packet with (IP Address X and Destination Port 0) or (IP Address Y and Destination Port 1) belongs to the same flow.			

Table 8-23 – Format of classifier structure

Field	Octet	Bits	Description
Length	0	[7:0]	Length of the classifier parameter (m) in bytes
Classifier_typ	1	[7:0]	Type of classifier: 0: IP v4 Address (m = 4) 1: TOS (m = 1) 2: VLAN priority (m = 1, only the three LSBs are meaningful) 3: VLAN TAG (m = 4) 4: Destination Port (m = 2) 5: Source port (m = 2) 6: IP v6 Address ~~+ Destination Port~~(m = 16) ~~7: IP Address + Source Port~~ ~~8: IP Address + TOS~~ ~~9~~7: Generic Classifier: offset, length, value, where m = offset (2 bytes) + length (1 byte) + value (≤ 252 bytes) is a variable (Note 1) ~~10~~8-254~~5~~: Reserved by ITU-T (Note 2)

Table 8-23 – Format of classifier structure

Field	Octet	Bits	Description
			255: Operator (m = 1) (Note 3)
Classifier parameter	2 to (1+m)	[(m*8)-1:0]	Contains the classifier value, for example 32 bits of IP address. m is the length of the field in bytes and is a function of the Classifier_typ.
NOTE 1 – The offset is the number of bits from the beginning of the APDU where the classifier looks for a match within the APDU, the length is the classifier field size in bits to be matched, the value contains the value of the classifier field to be matched. NOTE 2 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver. NOTE 3 –The coding of the operator classifier parameter is as shown in Table 8-23.1.			

Table 8-23.1 – Coding of operator classifier parameter

Parameter	Operator	Classifier description
0	(	Open parenthesis
1	AND	Logical AND operator
2	OR	Logical OR operator
3	)	Close parenthesis
4 – 255		Reserved by ITU-T

Table 8-24 – Format of TSpec field

Field	Octet	Bits	Description
Length	0	[7:0]	The length of the TSpec sub-fields following this field expressed in number of octets in the range between 2 and 255
TSpecBitMask	1 and 2	[15:0]	Traffic specifications bit mask. Each bit represents one traffic specification attribute field. When a represented bit value is set to one, the associated traffic specification attribute field shall be present in the TSpec field following this mask. When a represented bit value is set to zero, the associated traffic specification attribute field shall not be present. See clause 8.6.2.1 for the definition of these parameters. Traffic specification attribute fields that are present shall appear in the TSpec field in the following order:

Bit	TSpec attribute
0	Traffic Priority
1	Maximum Information Rate
2	Maximum Traffic Burst
3	Committed Information Rate
4	Tolerated Jitter
5	Maximum Latency
6	Grant Interval
7	Polling Interval
8	APDU Size
9 to 15	Reserved by ITU-T

Table 8-24 – Format of TSpec field

Field	Octet	Bits	Description
			If bit 3 is set (CIR field is present), then bit 0 shall also be set (TrafficPriority is present)
TrafficPriority	variable	[7:0]	Specifies the traffic priority, represented as an 8-bit unsigned integer in the range from 0 to 7. The value 7 represents the highest priority. This field shall be present if CIR field is present This field shall only be present if TSpecBitMask bit 0 is set to one
MIR	variable	[31:0]	Specifies the Maximum Information Rate in bit/s, represented as a 32-bit unsigned integer This field shall only be present if TSpecBitMask bit 1 is set to one
MaxTBurst	variable	[15:0]	Specifies the Maximum Traffic Burst (see clause 8.6.2.1) in kbytes, represented as a 16-bit unsigned integer This field shall only be present if TSpecBitMask bit 2 is set to one
CIR	variable	[31:0]	Specifies the Committed Information Rate (see clause 8.6.2.1) in bit/s, represented as a 32-bit unsigned integer This field shall only be present if TSpecBitMask bit 3 is set to one
ToleratedJitter	variable	[7:0]	Specifies the Tolerated Jitter in ms, represented as an 8-bit unsigned integer This field shall only be present if TSpecBitMask bit 4 is set to one
MaxLatency	variable	[7:0]	Specifies the Maximum Latency in ms, represented as an 8-bit unsigned integer This field shall only be present if TSpecBitMask bit 5 is set to one
GrantInterval	variable	[7:0]	Specifies Grant Interval in ms, represented as an 8-bit unsigned integer This field shall only be present if TSpecBitMask bit 6 is set to one
PollingInterval	variable	[7:0]	Specifies the Polling Interval in ms, represented as an 8-bit unsigned integer This field shall only be present if TSpecBitMask bit 7 is set to one
APDU Size	variable	[15:0]	APDU Size in bytes, represented as a 16-bit unsigned integer This field shall only be present if TSpecBitMask bit 6 (GrantInterval) and bit 8 are both set to one

17) Clause 8.6.2.3.2, Format of CL_EstablishFlow.cnf

Revise the text of clause 8.6.2.3.2 "Format of CL_EstablishFlow.cnf" as follows:

8.6.2.3.2 Format of CL_EstablishFlow.cnf

This message is sent by the node associated with a client to the application entity residing on the client, in response to a CL_EstablishFlow.req message. This message contains the status of the attempt to establish a flow. If successful, this message also contains the tuple (DeviceID, FlowID) that uniquely identifies the flow in the domain. If the status is a failure due to inability to meet the TSpec requirements in the CL_EstablishFlow.req message, then the rejected or wrong TSpec attributes shall be indicated by TSpecReject. In case the established flow is a bidirectional flow and the status is successful, this message shall also contain additional tuple (DeviceID, FlowID) with DeviceID corresponding to the endpoint node's DEVICE_ID, uniquely identifying the reverse flow in the domain. In case the request is for establishing a bidirectional flow and the status is failure due

to the inability to establish the reverse flow, the StatusCode shall show the corresponding failure in establishing that flow.

The format of the MMPL of the CL_EstablishFlow.cnf message ~~shall be as~~is shown in Table 8-25.

Table 8-25 – Format of the MMPL of the CL_EstablishFlow.cnf message

Field	Octet	Bits	Description
DeviceID	0	[7:0]	DEVICE_ID of the originating node
FlowID	1	[7:0]	FLOW_ID assigned by the originating node
StatusCode	2	[7:0]	Status of the request to establish a flow: 00_{16} = Success 01_{16} = Failure – maximum number of flows already started by the node 02_{16} = Failure – error in TSpec passed in CL_EstablishFlow.req 03_{16} = Failure – insufficient capacity to admit the flow 04_{16} = Failure – failed to establish flow in reverse direction because maximum number of flows already started by the endpoint node 05_{16} = Failure – error in TSpec passed in CL_EstablishFlow.req for the flow in the reverse direction 06_{16} = Failure – insufficient capacity to start the flow in the reverse direction 07_{16} = Failure – classifier rule is not supported 08~~7~~$_{16}$ – FF_{16} = Reserved (Note 1)
TSpecReject	3 and 4	[15:0]	This field contains TSpec failure bit mask. In case StatusCode indicates failure, this field specifies which TSpec attributes are wrong or were rejected. Each bit represents one traffic specification attribute. When a represented bit value is set to one, the associated traffic specification field is wrong or could not be delivered 0: if bit 0 is set to one then Traffic Priority was rejected 1: if bit 1 is set to one then Maximum Information Rate was rejected 2: if bit 2 is set to one then Maximum Traffic Burst was rejected 3: if bit 3 is set to one then Committed Information Rate was rejected 4: if bit 4 is set to one then Tolerated Jitter was rejected 5: if bit 5 is set to one then Maximum Latency was rejected 6: if bit 6 is set to one then Grant Interval was rejected 7: if bit 7 is set to one then Polling Interval was rejected 8: if bit 8 is set to one then APDU Size was rejected 9-15: reserved by ITU-T (Note 1)
Bidirectional	5	[7:0]	Set to 01_{16} if bidirectional flow establishment was requested in CL_EstablishFlow.req
DeviceID_B	6	[7:0]	DEVICE_ID of the Endpoint node (Note 2)

Table 8-25 – Format of the MMPL of the CL_EstablishFlow.cnf message

Field	Octet	Bits	Description
FlowID_B	7	[7:0]	FLOW_ID assigned by the endpoint node in case of a bidirectional flow. In case it is a unidirectional flow this field shall contain zero (Note 2)
NOTE 1 – If StatusCode is lower than 2_{16} then the TSpecReject field shall be ignored. NOTE 2 – If Bidirectional field is set to zero these fields shall not appear in the message.			

18) Clause 8.6.2.3.8, Format of FL_AdmitFlow.req (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.6.2.3.8 "Format of FL_AdmitFlow.req" (from ITU-T G.9961 Corrigendum 1) as follows:

8.6.2.3.8 Format of FL_AdmitFlow.req

…

Table 8-32 – Format of the MMPL of the FL_AdmitFlow.req message

Field	Octet	Bits	Description
DeviceID	0	[7:0]	DEVICE_ID of the originating node
FlowID	1	[7:0]	FLOW_ID assigned by the originating node
TSpec	Variable	See Table 8-24	See Table 8-24
TX Rate	Variable	See Table 8-33	The actual PHY data rate used by the transmitter, specified in bits per second for each channel estimation window, based on the bit loading per symbol, the symbol time, the FEC rate, <u>and</u> the number of repetitions~~, and the overhead according to the block size~~ The format of the TX rate field is described in Table 8-33. The offset of this field depends on the actual length of the previous (TSpec) field Note that the TX Rate should be specified per each channel estimation window
Bidirectional	Variable	[7:0]	Set to 01_{16} in case the established flow to be admitted is a bidirectional flow
DeviceID_B	Variable	[7:0]	DEVICE_ID of the endpoint node (Note)
TSpec_B	Variable	See Table 8-24	The TSpec of the flow in reverse direction (Note)
TX Rate_B	Variable	See Table 8-33	TX Rate for the reverse direction (Note)
Tunnel	Variable	[7:0]	00_{16} – direct flow admission is requested 01_{16} – tunnel flow admission is requested
EndPoint	Variable	[7:0]	DEVICE_ID of the endpoint node
NOTE – These fields appear only if Bidirectional field is set to 01_{16}.			

Table 8-33 – Format of the TX rate field

Field	Octet	Bits	Description
NumCEWindows	0	[4:0]	Number of items in the following list. Each item contains information for one channel estimation window. Each item includes three fields: CE_STime, CE_ETime and BitsPerSecond. The list shall not exceed n=32 items
EstimOverhead		[7:5]	Estimated DLL overhead in percentage represented as an unsigned integer minus 1 (Note 1). A value of zero represents 1% overhead. A value of 7 represents ≥ 8 % overhead
CE_STime	1	[7:0]	Start time as specified in Table 8-98 for first channel estimation window
CE_ETime	2	[7:0]	End time as specified in Table 8-99 for first channel estimation window
BitsPerSecond	3 and 4	[15:0]	The PHY data rate in bits per second for the first channel estimation window in steps of 32 kbit/s (Note 2).
CE_STime	4n–3	[7:0]	Start time as specified in Table 8-98 for last channel estimation window.
CE_ETime	4n–2	[7:0]	End time as specified in Table 8-99 for last channel estimation window.
BitsPerSecond	4n–1 to 4n	[15:0]	The PHY data rate in bits per second for the last channel estimation window in steps of 32 kbit/s (Note 2).
NOTE 1 – Defined as (Number of bytes crossing the PMI – number of bytes crossing the A-interface)/Number of bytes crossing the A-interface * 100% associated with a flow, including retransmission. The estimation of this parameter shall be vendor discretionary. NOTE 2 – BitsPerSecond = ($\underline{floor(k_P/N_{REP})} \cdot R \cdot F_{SC}$) / (1 + N_{GI}/N) where k_P is the number of loaded bits (see clause 7.1.3.3.1 of [ITU-T G.9960]), $\underline{N_{REP}}$ is the number of repetitions (see clause 7.1.3.3.1 of [ITU-T G.9960]), R is the code rate (see clause 7.1.3.2 of [ITU-T G.9960]), F_{SC} is the sub-carrier spacing, N_{GI} is the guard interval, and N is the number of sub-carriers (see clause 7.1.4.6 of [ITU-T G.9960]) for a payload OFDM symbol transmitted over a specified channel estimation window.			

19) Clause 8.6.4.3.1, Format of TM_NodeTopologyChange.ind (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.6.4.3.1 "Format of TM_NodeTopologyChange.ind" (from ITU-T G.9961 Corrigendum 1) as follows:

8.6.4.3.1 Format of TM_NodeTopologyChange.ind

…

Table 8-47 – Format of a NodeRec_Info field of the TM_NodeTopologyChange.ind message

Field	Octet	Bits	Description
NodeParam	0 to 2	[23:0]	A 24-bit field describing parameters and capabilities of the reporting node, formatted as described in Table 8-47.1

Table 8-47 – Format of a NodeRec_Info field of the TM_NodeTopologyChange.ind message

Field	Octet	Bits	Description
NodeAIFG	3	[7:0]	The value of T_{AIFG} supported by the node, represented as n x 1.28 µs; the value of n is an unsigned integer in the range between 4 and 96 (Note 1)
NumNodeVersion TLVs	4	[7:0]	Number of Versioning (N) TLVs included in this message Set to 0 if no Versioning TLVs are included.~~0 – Node supports version 0 of ITU-T G.9960 and ITU-T G.9961.~~ ~~All other values of this field are reserved by ITU-T for indicating support for future versions of the Recommendation (Note 2).~~
AAT_Size	5 and 6	[15:0]	Number (k) of local AAT entries associated with the reporting node (Note ~~3~~2)
AAT [0]	7 to 13	[47:0]	The first entry in the AAT. It contains the first local MAC address (Note 3~~4~~)
…	…	…	….
AAT [k–1]	7+(6*K–1)+1 to (7+6*k)	[47:0]	The last entry in the AAT (for k>1). It contains the last local MAC address
RemAAT_Size	Variable	[15:0]	Number (p) of AAT entries that are removed from the node AAT (Note ~~5~~4)
RemAAT [0]	Variable	[47:0]	First entry in the RemAAT. It contains the first MAC address that has been removed from the node AAT
…	…	…	….
RemAAT [p–1]	Variable	[47:0]	Last entry in the RemAAT. It contains the last MAC address that has been removed from the node AAT
NewAAT_Size	Variable	[15:0]	Number of AAT entries (q) that were added to the node AAT (Note ~~6~~5)
NewAAT [0]	Variable	[47:0]	First entry in the NewAAT. It contains the first MAC address that has been added to the node AAT
…	…	…	….
NewAAT [q–1]	Variable	[47:0]	Last entry in the NewAAT. It contains the last MAC address that has been added to the node AAT
Visibility_Size (Note 6~~7~~)	Variable	[7:0]	Number of nodes M in the domain which were detected by the reporting node, represented as an unsigned integer in the range between 1 and 249
Visibility_List	Variable	[39:0]	List of M fields, 5 octets each, describing a single detected node, formatted as described in Table 8-48
NodeVersionTLVs	Var	Var	Information related to the version and capabilities of the registering node. The format of this field shall be as described in Table 8-16.1

NOTE 1 – Once registered or upon re-registration in accordance with the topology update interval (see Table 8-82), a node shall not change the value of this field. Valid values for each medium are specified in Table 8-14.

~~NOTE 2 – A node indicating support for a certain version of this Recommendation shall also support all earlier versions of this Recommendation.~~

Table 8-47 – Format of a NodeRec_Info field of the TM_NodeTopologyChange.ind message

Field	Octet	Bits	Description
NOTE ~~3~~ 2 – If this field is zero, no AAT fields shall be included in the message. Otherwise, it contains the number of entries in the full local AAT that are specified in the message. The first time the node reports this message to the domain master, it shall include in the message its full local AAT. NOTE ~~4~~ 3 – The first MAC address shall be the REGID of the reporting node. NOTE ~~5~~ 4 – If this field is zero, no entries have been removed from the local AAT since the previous transmitted report for that node, and no RemAAT fields shall be included. Otherwise, it contains the number of removed entries from the local AAT. This field shall be set to zero if AAT_Size field is non-zero. NOTE ~~6~~ 5 – If this field is zero, no entries have been added to the local AAT since the previous transmitted report for that node and no NewAAT fields shall be included. Otherwise, it contains the number of added new entries to the local AAT since last transmitted report for that node. This field shall be set to zero if AAT_Size field is non-zero. NOTE ~~7~~ 6 – Value 255 indicates that no record on visibility is attached (while a node possesses information on visibility). Value 0, and values 251-254 are reserved by ITU-T.			

…

Table 8-48 – Format of a Visibility_List field

Field	Octet	Bits	Description
DEVICE_ID	0	[7:0]	DEVICE_ID of a node that the reporting node detected.
~~BitRate~~BitsPerSecond	1 to ~~43~~	[~~2~~31:0]	Bits [15~~1~~:0] indicate the PHY data rate in bits per second from the reporting node to the detected node; Bits [~~2~~31:16~~2~~] indicate the PHY data rate from the detected node to the reporting node ~~Both data rates shall be represented as 12-bit unsigned integers, in steps of 0.5 Mbit/s~~ (Notes 1, 2)
NOTE 1 – Both data rates shall follow the formula $\sum_{i=1}^{N} \text{BitsPerSecond}_i * \frac{T_i}{T_{Cycle}}$ where N is the number of channel estimation windows in the MAC cycle, BitsPerSecond_i is the result of applying the same formula as Note 2 in Table 8-33 to the *i*-th channel estimation window, T_i is the duration of the *i*-th channel estimation window and T_{Cycle} is the duration of the MAC cycle. If no channel estimation is available for a particular window, RCM parameters shall be used for that window. If the data rate with the particular detected node is not available, the value of this parameter shall be set to $FFFF_{16}$~~; The value shall be set to zero if the detected data rate is less than 0.5 Mbit/s~~. NOTE 2 – In case the DM receives conflicting information in the BitsPerSecond field from the Visibility_List of two different nodes for a given link, the DM shall take into account the minimum value between both.			

…

20) Clause 8.6.4.3.5, Selection and maintenance of the DNI (from ITU-T G.9961 Corrigendum 1)

Revise the text in clause 8.6.4.3.5 "Selection and maintenance of the DNI" (from ITU-T G.9961 Corrigendum 1) as follows:

…

Table 8-50 – Format of the MMPL of the TM_DomainRoutingChange.ind message

Field	Octet	Bits	Description
NumTmInd	0	[7:0]	This value indicates the number (m) of node topology change messages received by the ~~node~~ domain master after the previous transmission of domain routing change message that requested an acknowledgement via this message (see Table- 8-46) (Note 4)
DEVICE_ID[0]	1	[7:0]	DEVICE_ID of the first node that requested an acknowledgement via this~~whose topology information was used in generating this routing change~~ message
SeqNumber[0]	2 and 3	[15:0]	Sequence number of ~~the first node that sent~~ the topology change message of the first node that requested an acknowledgement via this~~that was used in generating this routing change~~ message
…	…	…	
DEVICE_ID[m–1]	1+3*(m–1)	[7:0]	DEVICE_ID of the m-th node that requested an acknowledgement via this~~whose topology information was used in generating this routing change~~ message
SeqNumber[m–1]	2+3*(m–1) and 3+3*(m–1)	[15:0]	Sequence number of ~~the m-th node that sent~~ the topology change message of the m-th node that requested an acknowledgement via this message~~was used in generating this routing change~~
NumNodesRecs	Variable	[7:0]	Number of source node records (n) in the message (Note 1)
NodeRec[0]_ID	Variable	[7:0]	DEVICE_ID of the first source node in the list
NodeRec[0]_Size	Variable	[15:0]	Size of the first record in bytes represented as an unsigned integer (Note 3)
NodeRec[0]_Info	Variable	See Table 8-51	First record information field, with a format as defined in Table 8-51
…	…	…	
NodeRec[n–1]_ID	Variable	[7:0]	DEVICE_ID of the last source node in the list
NodeRec[n–1]_Size	Variable	[~~7~~15:0]	Size of the last record in bytes represented as an unsigned integer
NodeRec[n–1]_Info	Variable	See Table 8-51	Last record information field, with a format as defined in Table 8-51
NumResignNodes	Variable	[7:0]	Number of resigned nodes (k~~m~~) in the resigned node list (Note 2)
ResignedNodes[0]	Variable	[7:0]	DEVICE_ID of the first resigned node in the list
…	…	…	…
ResignedNodes[~~m~~k–1]	Variable	[7:0]	DEVICE_ID of the last resigned node in the list

NOTE 1 – The number of node records in the list includes the domain master.

NOTE 2 – If there are no nodes that resigned from the domain since the last update, this field shall be set to zero and the list of resigned nodes shall have no entries.

NOTE 3 – All NodeRec[i]_Size fields shall be > 0.

Table 8-50 – Format of the MMPL of the TM_DomainRoutingChange.ind message

Field	Octet	Bits	Description
NOTE 4 – A node can have more than one entry as it can send multiple node topology change messages.			

Table 8-51 – Format of NodeRec[i]_Info

Field	Octet	Bits	Description
NumDestNodes	0	[7:0]	Number of destination hidden node pairs (n) of the unicast routing table. Each pair contains the DEVICE_ID of the destination hidden node and the DEVICE_ID of the relay node toward the specified destination hidden node
DestNodeID[0]	1	[7:0]	DEVICE_ID of the first destination hidden node
RelNodeID[0]	2	[7:0]	DEVICE_ID of the relay node toward the first specified destination hidden node
…	…	…	…
DestNodeID[n–1]	2×(n–1) + 1	[7:0]	DEVICE_ID of the last destination hidden node
RelNodeID[n–1]	2×(n–1) + 2	[7:0]	DEVICE_ID of the relay node toward the last specified destination hidden node
BRT_Size	Variable	[15:0]	Length in bytes of all the BRT entries of the node plus one. This length includes the NumBRTEntries and the BRTEntry[i] fields
NumBRTEntries	Variable	[7:0]	Number of entries (b) of the BRT of the node
BRTEntry[0]	Variable	Table 8-52	Content of the first entry of the BRT as described in Table 8-52
…	…	…	…
BRTEntry[b–1]	Variable		Content of the last entry of the BRT as described in Table 8-52
NodeAIFG	Variable	[7:0]	The value of T_{AIFG} supported by the node, represented as n × 1.28 µs; the value of n is an unsigned integer in the range between 4 and 96
IsMpr	Variable	[0]	Set to one if node is an MPR, otherwise set to zero
HopCount	Variable	[7:1]	Set to the (number of hops – 1) that the node is from the domain master. It is set to zero, if the node has a direct link to the domain master
AAT_Size	Variable	[15:0]	Number (k) of local AAT entries associated with the reporting node (Note 1)
AAT [0]	Variable	[47:0]	The first entry in the AAT. It contains the first local MAC address
…	…	…	….
AAT [k–1]	Variable	[47:0]	The last entry in the AAT. It contains the last local MAC address
RemAAT_Size	Variable	[15:0]	Number (p) of AAT entries that are removed from the node AAT (Note 2)

Table 8-51 – Format of NodeRec[i]_Info

Field	Octet	Bits	Description
RemAAT [0]	Variable	[47:0]	First entry in the RemAAT. It contains the first MAC address that has been removed from the node AAT
…	…	…	….
RemAAT [p–1]	Variable	[47:0]	Last entry in the RemAAT. It contains the last MAC address that has been removed from the node AAT
NewAAT_Size	Variable	[15:0]	Number of AAT entries (q) that were added to the node AAT (Note 3)
NewAAT [0]	Variable	[47:0]	First entry in the NewAAT. It contains the first MAC address that has been added to the node AAT
…	…	…	….
NewAAT [q–1]	Variable	[47:0]	Last entry in the NewAAT. It contains the last MAC address that has been added to the node AAT
NumNodeVersionTLVs	Variable	[7:0]	Number of versioning (N) TLVs included in this message Set to 0 if no Versioning TLVs are included
NodeVersionTLVs	Var	Var	Information related to the version and capabilities of the registering node. The format of this field shall be as described in Table 8-16.1

NOTE 1 – If this field is zero, no AAT fields shall be included in the message. Otherwise, it contains the number of entries in the full local AAT that are specified in the message.

NOTE 2 – If the AAT_Size field is non-zero, this field ~~is~~ shall be set to zero and ignored by the receiver. If the AAT_Size field is zero, this field contains the number of removed entries from the local AAT and a value of -zero means that~~,~~ no entries have been removed from the local AAT since the previous transmitted report for that node, and no RemAAT fields shall be included. ~~Otherwise, it contains the number of removed entries from the local AAT. This field shall be set to zero if AAT_Size field is non-zero.~~

NOTE 3 – If the AAT_Size field is non-zero, this field ~~is~~ shall be set to zero and ignored by the receiver. If the AAT_Size field is zero, this field contains the number of added new entries to the local AAT and a value of zero means that~~,~~ no entries have been added to the local AAT since the previous transmitted report for that node and no NewAAT fields shall be included. ~~Otherwise, it contains the number of added new entries to the local AAT since the last transmitted report for that node. This field shall be set to zero if AAT_Size field is non-zero.~~

…

21) Clause 8.6.6.1, Domain master selection at initialization

Revise the text of clause 8.6.6.1 "Domain master selection at initialization" as follows:

8.6.6.1 Domain master selection ~~at initialization~~

8.6.6.1.1 Domain master selection at initialization

Following its initialization, a node shall not transmit and shall try to detect MAP frames or RMAP frames associated with one of the domains the node targets to join during a time interval ~~up to~~ t_0. The values of t_0 (i.e., JOIN_INTERVAL_T_0) are specified in clause 8.4.

NOTE 1 – The node identifies a domain it intends to join by comparing the domain name in the detected MAP or RMAP messages with the parameter "Target Domain Name" in its information database (see clause 8.6.1).

NOTE 2 – The value of t_0 is selected taking into account that with relayed admission the time period between two RMAP frames may be up to 200 MAC cycles (see clause 8.5.6)

If MAP or RMAP frames of the target domain are detected within t_0, the node shall start the admission procedure to join the domain, as defined in clause 8.6.1. If no MAP or RMAP frames of the target domain have been detected within t_0 time, the node shall infer that there is no active domain master present in the domain and, after the t_0 interval expires, shall act using the following rules:

- It may start a new domain by becoming its domain master and shall start transmitting MAP frames within duration of one MAC cycle after a t_1 time interval following the expiration of t_0.

  The value of t_1 shall be randomly generated by the node and shall be the range between 0 and 1 seconds. The method of generation of t_1 values is left to the discretion of the implementer.
- If either a MAP or an RMAP frame of the target domain is detected during the t_1 time interval, the node shall not transmit the MAP frame and shall try to synchronize with the detected MAP or RMAP frames and register to the domain using the procedure specified in clause 8.6.1 (as an endpoint node).

8.6.6.1.2 Domain master maintenance

A node that receives a MAP-D with an SA that is different from its own DM's REGID and that has the same domain name as its own domain name, shall send an ADM_NodeReportMAPD.ind message containing the received MAP-D to its DM.

A DM that detects a MAP-D with an SA that is different from its own REGID and that has the same domain name (either directly or via an ADM_NodeReportMAPD.ind message sent by a node in its domain) shall initiate the procedure to merge the two domains by applying the following rules.

The DM shall first compare the following attributes in order of priority from 1 to 4 with that of the DM corresponding to the domain that it needs to merge with, and shall then rank itself relative to that DM.

1) By configuration setting – If the DM was preferentially selected (designated by the user or remote management system) to operate as a domain master, it shall be ranked higher.
2) By number of nodes belonging to its domain – The DM that is managing a higher number of nodes shall be ranked higher.
3) By profile number – The node advertising its compliance to the higher profile number shall be ranked higher.
4) By capability to operate as a security controller – A node that is capable of operating as a security controller shall be ranked higher.

The DM that has the lower ranking after applying the above criteria shall follow the merging procedure described in clause 8.6.6.1.3.

If the ranks of the DMs are equal, the DM shall compare its REGID (in bit-reversed order) with that of the other DM (in bit-reversed order).

- If its REGID (in bit-reversed order) is less than that of the other DM (in bit-reversed order), the DM shall be ranked lower and shall follow the merging procedure described in clause 8.6.6.1.3.
- If its REGID (in bit-reversed order) is greater than that of the other DM (in bit-reversed order), the DM shall ignore the received MAP-D, since the other DM is ranked lower and is expected to complete the merging procedure described in clause 8.6.6.1.3.

NOTE – For example, if a DM has a REGID 00-B0-D0-86-BB-F7, the bit-reversed order of the REGID is the number EF-DD-61-0B-0D-00.

A node that decodes a MAP-A with an SA that is different from its own DM's REGID and with a DNI (contained in the received MAP-A) equal to the value of the DNI calculated by using the hash key indicated in the DNI_KeyID field of the MAP header and its own domain name (see clause 8.6.8.2.1), shall send an ADM_NodeReportMAPA.ind message containing the relevant information from the received MAP-A. The DM should then try to detect MAP-Ds from the DM that transmitted the MAP-A, to confirm the existence of another domain with the same domain name.

A DM that decodes a MAP-A with an SA that is different from its own REGID and with a DNI (contained in the received MAP-A) equal to the value of the DNI calculated by using the hash key indicated in the DNI_KeyID field of the MAP header and its own domain name (see clause 8.6.8.2.1) should try to detect MAP-Ds from the DM that transmitted the MAP-A to confirm the existence of another domain with the same domain name.

8.6.6.1.3 Merging procedure

A DM that has a lower ranking, after applying the criteria described in clause 8.6.6.1.2, shall refrain from sending new MAP-Ds. Also, this DM shall force the resignation of all the nodes in its domain using the mechanisms described in clause 8.6.1.1.3.2.

Upon reception of the resignation confirmation from all the endpoint nodes of the domain or after the timeouts, the DM shall consider itself as resigned and shall try to register to the new domain.

8.6.6.1.4 Management message formats for domain master maintenance

8.6.6.1.4.1 Format of ADM_NodeReportMAPD.ind

The format of the MMPL of the ADM_NodeReportMAPD.ind message shall be as shown in Table 8-56.1.

Table 8-56.1 – Format of the MMPL of the ADM_NodeReportMAPD.ind

Field	Octet	Bits	Description
IncomingMAPD	var	var	LCDU of the MAP-D received by the node sending this report

8.6.6.1.4.2 Format of ADM_NodeReportMAPA.ind

The format of the MMPL of the ADM_NodeReportMAPA.ind message shall be as shown in Table 8-56.2.

Table 8-56.2 – Format of the MMPL of the ADM_NodeReportMAPA.ind

Field	Octet	Bits	Description
RX_SA	0 to 5	[47:0]	MAC address of the received MAP-A

Table 8-56.2 – Format of the MMPL of the ADM_NodeReportMAPA.ind

Field	Octet	Bits	Description
RX_DNI	6 and 7	[15:0]	DNI specified in the received MAP-A
RX_HASH	8	[2:0]	Hash key indicated in the DNI_KeyID field of the received MAP-A
Reserved		[7:3]	Reserved by ITU-T (Note).
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

22) Clause 8.6.7, Selection of PHY-frame header segmentation

Revise the text of clause 8.6.7 "Selection of PHY-frame header segmentation" as follows:

8.6.7 Selection of PHY-frame header segmentation

The domain master shall use $D = 2$ (as defined in clause 7.1.3.5.2 of [ITU-T G.9960]) for the MAP-D frame transmission regardless of its medium type. Any node relaying the MAP-D frame shall also use $D = 2$.

The value of D used in a specific TXOP shall be indicated in TXOP attributes extension (clause 8.8.4.1.1) and field HSI~~EG~~ of the PHY-frame header (see Table 7-1 of [ITU-T G.9960]). Selection of D for a given TXOP is vendor discretionary.

23) Clause 8.6.8.2, Selection and maintenance of the DNI

Revise the text of clause 8.6.8.2 "Selection and maintenance of the DNI" as follows:

8.6.8.2 Selection and maintenance of the DNI

The domain master, prior to sending its first MAP frame, shall compute the DNI using the default value of the hash key and using the procedure defined in clause 8.6.8.2.1. Further, the domain master shall monitor the DNI of all visible neighbouring domains and re-compute the DNI using one of the alternative hash keys if the same value of DNI is discovered, after verifying that the domain name of the neighbouring domain is different from its own domain name (see clause 8.6.6.1.2).

If during domain operation, a neighbouring domain with the same DNI is discovered by the domain master or reported by a node of the domain (in its topology report), the domain master shall change the DNI by applying one of the alternative hash keys. Further, the domain master shall transmit MAPs with a new DNI.

Nodes that join the domain shall verify the DNI using the hash key indicated in the DNI_KeyID field of the MAP header.

24) Clause 8.8.1, MAP generation and distribution, and clause 8.8.2, MAP frame transmission (both taken from G.9961 Corrigendum 1)

Revise the text of clause 8.8.1 "MAP generation and distribution" and clause 8.8.2 "MAP frame transmission" (both taken from G.9961 Corrigendum 1) as follows:

8.8.1 MAP generation and distribution

The domain master shall generate and manage distribution of a MAP each MAC cycle. The MAP may vary from one MAC cycle to another.

The domain master shall transmit at least one MAP-A frame each MAC cycle and may transmit additional MAP frames (MAP-A or MAP-D) each MAC cycle. However, the MAP transmitted by the domain master shall not change within a MAC cycle, except the sub-fields of the Auxiliary Information field that are not related to scheduling and persistence information.

In addition, the domain master may designate one or more nodes as MAP relays. Designated nodes shall transmit RMAP frames containing the MAP, as described in clauses 8.5.6, 8.6.1 and in Table 8-70.

The domain master ~~may~~ shall distribute the MAP for all nodes registered to a domain by transmitting ~~default MAP frames (MAP-D) or both default and active~~ MAP-A frames ~~(MAP-A)~~. The domain master shall distribute the MAP and other information necessary for registration by transmitting default MAP-D frames. The payload bits of the MAP-D frame (and RMAP-D frame) and MAP-A frame (and RMAP-A frame) shall be mapped to sub-carriers as described in clause 7.1.2.3.2.1.10 of [ITU-T G.9960] ~~using Pre-defined BAT Type 1, while the payload bits of the MAP-A frame (and RMAP-A frame) shall be mapped to sub-carriers using Pre-defined BAT Type 2~~. The type of the MAP frame (MAP-D or MAP-A) is indicated by the MAP_TYPE field of the MAP frame header (see clause 7.1.2.3.2.1.10 of [ITU-T G.9960]) and in the TXOP allocated for the MAP frame transmission (see MAP Type field in Table 8-63).

The decision to transmit a MAP-D frame in addition to~~or~~ a MAP-A frame ~~or both~~ in a particular MAC cycle is left to the discretion of the domain master. If the domain master transmits both MAP-D(s) and MAP-A(s) in the same MAC cycle, the scheduling information and persistence information defined in the MAP messages of those MAP frames shall not conflict.

The MAP-D frame is transmitted to facilitate admission of new nodes to the domain. Therefore, the MAP-D frame ~~may~~ shall include only the relevant information needed by the registering nodes to synchronize with the MAC cycle, to learn the regional transmission parameters, and to learn the header segmentation (see clauses 8.6.7 and 8.8.4.1.1) of MAP-A and RMAP-A. The content of the MAP-D shall include the followings:

- TXOP descriptor(s) and TXOP attributes extensions describing all MAP-A, RMAP-A and MAP-D transmissions,
- TXOP descriptor(s) and TXOP attributes extensions describing transmit opportunity (e.g., RCBTS) for registering node, and
- Auxiliary information necessary for registration – Domain name, PSD-related domain info, Registration code, and Timer-related domain info (see Table 8-73).

~~and to locate the TXOP to transmit the registration request message. A MAP-D message that only contains this reduced information is referred to as a reduced MAP-D. It is recommended that the MAP-D frame be transmitted in the reduced format to save bandwidth.~~ The TXOPs descriptors included in a ~~reduced~~ MAP-D frame shall be described using the absolute timing extension (see clause 8.8.4.1.1) ~~when it is needed to skip over TXOPs that are specified in the complete (not reduced) MAP of the same MAC cycle~~. The RMAP-D frame shall be constructed following the same requirements as MAP-D.

NOTE – The MAP-D frame is ~~normally~~ transmitted to facilitate admission of new nodes to the domain; rare transmission of a MAP-D may result in unacceptably long admission time and failure of the admission procedure.

8.8.2 MAP frame transmission

During each MAC cycle, the domain master shall allocate at least one CFTXOP assigned for MAP-A frame transmission. The domain master may allocate additional CFTXOPs and/or CFTSs in STXOPs assigned for MAP transmission.

The domain master shall transmit only one MAP frame in each allocated CFTXOP assigned for MAP transmission. The domain master may transmit additional MAP frames in CFTSs in STXOPs

assigned for MAP transmission. The first transmitted MAP frame in a MAC cycle shall be a ~~complete~~ MAP-A frame. The domain master may transmit MAP frames in CBTS. MAP frames transmitted in CBTS shall use a medium access priority of MA3.

At least one MAP frame shall be transmitted during each MAC cycle that describes the complete schedule of the immediately following MAC cycle except for cases where the part of the MAP corresponding to Persistent TXOPs might not contain the scheduling information for the immediately following MAC cycle (see clause 8.8.6). Once the MAP for a particular MAC cycle is announced, the scheduling for that MAC cycle shall not be changed by any subsequent transmissions of MAP/RMAP frames. Transmission of MAP or RMAP frames shall be completed at least MAP_TX_SETUP_TIME before the start of the MAC cycle that it describes. The value of MAP_TX_SETUP_TIME is defined in clause 8.4. The scheduler shall ensure a gap of INTER_MAP_RMAP_GAP (see clause 8.8.6) between the end of the transmission of a MAP or RMAP frame and the start of the RMAP that has to be derived from that MAP or RMAP. The destination identifier (MI and DID fields) in transmitted MAP frames shall indicate the broadcast address.

NOTE 1 – Nodes already registered to the domain are familiar with the domain specific parameters, such as the regional PSD masks. For these nodes, decoding MAP-As is likely to result in improved performance compared to decoding MAP-Ds. ~~It is therefore recommended that the MAP carrying the complete schedule be a MAP-A.~~

To enable potential hidden nodes to join the domain, the domain master shall schedule the transmission of RMAP-D frames by the MAP relay capable nodes. For each MAP relay capable node the domain master shall schedule RMAP-D transmission in three consecutive MAC cycles. The domain master shall schedule the RMAP-D transmissions so that during each (JOIN_INTERVAL_T_0)/2 interval all nodes that are MAP relay capable transmit RMAP-D~~,~~ at least once in a round-robin manner ~~and with a maximum interval of 1 s, a different "MAP relay capable" node each time it schedules an RMAP-D transmission~~.

NOTE 2 – This ensures that a joining node that can detect RMAP-D from only one node in a domain that it intends to join, can still detect at least two consecutive RMAP-D transmissions within the JOIN_INTERVAL_T_0, which is sufficient to synchronize its transmit clock with the node transmitting the RMAP-D and decoding the MAP.

The transmission parameters used for transmission of MAP PHY frames are specified in clause 7.1.2.3.2.1.10 of [ITU-T G.9960]. In addition, the following shall apply to MAP transmissions:

- MAP-Ds (and RMAP-Ds) shall be sent in the lowest configured bandplan (the minimal bandplan configured for the domain, as specified in clause 7.4.9 of [ITU-T G.9962]), or at a lower bandplan.
- MAP-As (and RMAP-As) shall be sent in a bandplan which is lower than or equal to the maximal configured bandplan (as specified in clause 7.4.10 of [ITU-T G.9962]).
- The PSD-related info in MAP-D (and in MAP-A if present) shall carry information relating to the highest bandplan configured (e.g., for powerlines the default being 100 MHz), regardless of the bandplan used for transmission of the MAP itself.

NOTE 3 – For robustness, it is recommended to avoid using bandplans higher than the lowest configured bandplan for MAP-A and RMAP-A transmissions.

If the domain master intends to change some of the sub-fields of the auxiliary information field, it shall use the mechanism of the auxiliary information validity counter (AUX_VALID) described in clause 8.8.5. During this time, the domain master shall avoid scheduling RMAP-D transmission since it affects the content of the MAP-D (e.g., dynamic SM changes).

…

25) Clause 8.8.3, MAP header

Revise the text of clause 8.8.3 "MAP header" as follows:

8.8.3 MAP header

…

Table 8-62 – MAP header format

Field	Octet	Bits	Description
Sequence Number	0 and 1	[15:0]	A MAP sequence number. The sequence number shall be incremented by one for each MAC cycle (modulo 2^{16}). An RMAP shall keep the sequence number of the original MAP (shall not increment the sequence number).
MAP Header Length	2	[7:0]	The length of the MAP header expressed in a number of 32 bit words.
Number of entries	3 and 4	[15:0]	Number of TXOP descriptors, including TXOP extensions, in the MAP.
TICK_Factor	5	[2:0]	A time shift factor that shall be used to determine the resolution of a TXOP TIME_UNIT (see clause 8.2.3). The resolution of a TIME_UNIT is determined as follows: TIME_UNIT = TICK * 2^{TICK_Factor} The values of TICK are defined in clause 8.4.
Reserved		[3]	Reserved by ITU-T (Note 1).
RoutingAuthorization		[4]	Relevant to CRTM mode in case of broken link (see clause 8.6.4.2.1). 0 – nodes are not authorized to calculate routing 1 – nodes are authorized to calculate routing temporarily until routing information arrived from the domain master.
Topology Mode		[5]	0 – CRTM mode. 1 – DRTM mode.
Handover In Progress (HOIP)		[6]	When set to one, indicates that the present domain master is handing over its role to a newly registered node. At other times, is set to zero.
MAP Modified		[7]	The domain master shall set this bit <u>to one</u> when the <u>schedule (current, future or both) indicated in the</u> MAP <u>frame</u> is <u>different from</u>~~modified compared with~~ the <u>one indicated in a</u> ~~previous~~ MAP <u>frame from the previous MAC cycle. Otherwise, it shall be set to zero</u>~~and reset otherwise~~.
Future Schedule Life Time (FSLT)	6	[3:0]	If FSLT is non-zero, the MAP frame carries the future TXOP schedule which shall take effect when CSLT reaches zero and shall remain valid for FSLT plus one consecutive MAC cycles after it takes effect.
Current Schedule Life Time (CSLT)		[7:4]	CSLT + 1 is the number of consecutive MAC cycles in which the TXOP schedule described in the MAP shall remain valid. The value of CSLT shall be reduced by one after each MAC cycle with FSLT > 0.

Table 8-62 – MAP header format

Field	Octet	Bits	Description
Domain Name Identifier (DNI)	7 and 8	[15:0]	The generation of DNI value and its format are defined in clause 8.6.8.2.
RoutingSequenceNumber	9 and 10	[15:0]	The sequence number of the last transmitted routing message.
Reserved	11	[4:0]	Reserved by ITU-T.
Routing Algorithm		[6:5]	Contains a specified standard algorithm (Note 2).
PrvRouting Algorithm		[7]	Bit 7 – If set to one it means that the domain master uses a vendor-specific algorithm. If it is zero, then bits 6:5 contains a specified standard algorithm.
MAC cycle duration	12 to 14	[23:0]	The duration of the MAC cycle in TICK units. There are two cases: If the MAP includes a future persistent schedule, then the duration is of this future MAC cycle. In all other cases the duration is of the next MAC cycle. This duration covers the time period between two consecutive CYCSTARTs (see clause 7.1.2.3.2.1.3 of [ITU-T G.9960]). The minimal and maximal durations of the MAC cycle are defined in clause 8.4.
DNI_KeyID	15	[2:0]	A value of DNI key (m) encoded as an unsigned integer minus 2; this key shall be used to compute the DNI as defined in clause 8.6.8.2.1.
Reserved		[7:3]	Reserved by ITU-T (Note 1).
NOTE 1 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver. NOTE 2 – These bits shall be set to zero. Specification of standard routing algorithms is for further study.			

26) Clause 8.8.4.1.1, TXOP attributes extension data

Revise the text of clause 8.8.4.1.1 "TXOP attributes extension data" as follows:

8.8.4.1.1 TXOP attributes extension data

A TXOP attributes extension shall be identified by extension type 0 and shall be used to specify the TXOP duration and restrictions on the type of traffic that can be sent within the TXOP.

The TXOP start time is defined as the start time of the TXOP associated with the previous TXOP descriptor in the MAP plus the duration of that~~sum of the durations of all preceding~~ TXOP~~s~~, unless the start time is marked as the same as the start time of the TXOP associated with the previous TXOP ~~start time~~descriptor in the MAP (by setting the 'Start_Time_Type' bit to one) and unless the TXOP absolute timing extension is used. The Start_Time_Type bit in the extension shall be ignored if the TXOP absolute timing extension is present. The end time of each TXOP in the MAP shall be equal to or smaller than the end of the MAC cycle.

The format of the TXOP attributes extension is described in Table 8-65.

Table 8-65 – TXOP attributes extension data format

Field	Octet	Bits	Description
Length	0 to 2	[17:0]	Duration allocated to the TXOP in TIME_UNIT units where the size of a TIME_UNIT is equal to the base TICK size (the values of TICK are defined in clause 8.4) multiplied by a constant factor defined in the MAP header (see TICK_Factor in clause 8.8.3).
Traffic Limitation		[19:18]	Restrictions on the type of traffic that can be sent in the TXOP: 0 – No restriction (default). 1 – Channel estimation only. 2-3 – Reserved by ITU-T.
Non-Persistent/Persistent		[20]	0 – Non-persistent TXOP (Default). 1 – Persistent TXOP.
Start_Time_Type		[21]	0 – TXOP start time is at the start time of the TXOP associated with the previous TXOP descriptor in the MAP plus the duration of that TXOP~~the end of the previous TXOP and shall be computed as the sum of the durations of all preceding TXOPs~~ (default). 1 – TXOP start time is the same as the start time of the TXOP associated with the previous TXOP descriptor in the MAP (e.g., spatial reuse). This field shall be ignored if the TXOP absolute timing extension is appended to the TXOP descriptor.
Header segmentation		[22]	0 – PHY-frame header is segmented into one symbol ($D = 1$). 1 – PHY-frame header is segmented into two symbols ($D = 2$). (see clause 7.1.3.5.2 of [ITU-T G.9960])
Enhanced frame detection (EFD) STXOP Indicator		[23]	0 – Indicates a non-EFD STXOP (see clause 8.3.3). 1 – Indicates an EFD STXOP (see clause 8.3.3.5).
TS_Grid_Resync	3	[0]	0 – A node that inferred loss of synchronization with the TS grid of this STXOP shall attempt to resynchronize with the TS grid (as described in clause 8.3.3.6) (Default). 1 – A node that inferred loss of synchronization with the TS grid of this STXOP shall refrain from transmission until the end of the STXOP (as described in clause 8.3.3.6) (Note).
INUSE signal required		[1]	This bit instructs nodes contending for transmission in a CBTS in this TXOP whether to use INUSE signal: 0 – INUSE signal shall not be used. 1 – INUSE signal is required.
RTS/CTS required		[2]	This bit instructs the transmitter to use RTS/CTS prior to the data: 0 – RTS/CTS shall not be used. 1 – RTS/CTS is required.
Extension Type and Extension		[7:3]	See Table 8-64.

Table 8-65 – TXOP attributes extension data format

Field	Octet	Bits	Description
NOTE – This bit does not apply to CBTXOP without INUSE.			

27) Clause 8.8.4.1.5, CBTS nodes information Extension Data

Revise the text of clause 8.8.4.1.5 "CBTS nodes information Extension Data" as follows:

8.8.4.1.5 CBTS nodes information Extension Data

A CBTS nodes information extension shall be identified by extension type 4 and shall be used to specify the specific list of nodes that are allowed to contend in a particular CBTS as specified via a TXOP descriptor (see clause 8.8.4.2). The list of nodes shall be described by indicating the DEVICE_IDs. Several CBTS nodes information extension may be used for ~~describing~~ a TXOP descriptor that describes a CBTS~~list~~.

The CBTS nodes information extension is described in Table 8-69.

Table 8-69 – CBTS nodes information Extension Data format

Field	Octet	Bits	Description
Include_Exclude	0	[0]	0 – All nodes indicated in the following entries ~~can~~ may contend in this CBTS 1 – All nodes indicated in the following entries shall ~~can~~not contend in this CBTS
Entry format		[1]	0 – byte map format 1 – bit map format
Reserved		[7:2]	Reserved by ITU-T (Note)
Byte map format			
Entry number 1	1	[7:0]	0 = New nodes joining network 1 to 250 identifies the DEVICE_ID of a registered node 251 to 254 – Reserved by ITU-T 255 – this entry shall be ignored
Entry number 2	2	[7:0]	0 = New nodes joining network 1 to 250 identifies the DEVICE_ID of a registered node 251 to 254 – Reserved by ITU-T 255 – this entry shall be ignored
~~Entry number 3~~	~~3~~	~~[7:0]~~	~~0 = New nodes joining network~~ ~~1-250 identifies the DEVICE_ID of a registered node~~ ~~251-254 – Reserved by ITU-T~~ ~~255 – this entry shall be ignored~~
Reserved	3~~4~~	[2:0]	Reserved by ITU-T (Note)
Extension Type and Extension		[7:3]	See Table 8-64

Table 8-69 – CBTS nodes information Extension Data format

Field	Octet	Bits	Description
Bit map format			
Entry number 1	1	[7:0]	0 – New nodes joining network 1-250 identifies the DEVICE_ID of a registered node 251-255 – Reserved by ITU-T
Entry number 2	2~~ & 3~~	[0]	Identifies status for DEVICE_ID = Entry number 1+1 0 – node included in the list 1 – node not included in the list
Entry number 3		[1]	Identifies status for DEVICE_ID = Entry number 1+2 0 – node included in the list 1 – node not included in the list
…		…	…
Entry number ~~158~~		[~~136~~]	Identifies status for DEVICE_ID = Entry number 1+~~147~~ 0 – node included in the list 1 – node not included in the list
Entry number ~~169~~		[~~147~~]	Identifies status for DEVICE_ID=Entry number 1+~~158~~ 0 – node included in the list 1 – node not included in the list
~~Reserved~~		~~[15]~~	~~Reserved by ITU-T (NOTE)~~
Reserved	~~4~~3	[2:0]	Reserved by ITU-T (Note)
Extension Type and Extension		[7:3]	See Table 8-64
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

28) Clause 8.8.5, Auxiliary information field

Revise Table 8-73 of clause 8.8.5 "Auxiliary information field" as follows:

Table 8-73 – Types of auxiliary information sub-fields

Type	Value	Description
Reserved	00_{16}	Reserved by ITU-T
Domain name	01_{16}	A sub-field indicating domain name represented in ASCII characters, as described in clause 8.8.5.2
Long inactivity schedule	02_{16}	A sub-field indicating long inactivity schedules, as described in clause 8.8.5.3
Short inactivity schedule	03_{16}	A sub-field indicating short inactivity schedules, as described in clause 8.8.5.4
PSD-related domain Info	04_{16}	A sub-field carrying PSD-related domain information, as described in clause 8.8.5.5
New domain master ID	05_{16}	A sub-field carrying the DEVICE_ID and the REGID of the node that will take the role of the domain master after the handover is complete, as described in clause 8.8.5.6

Table 8-73 – Types of auxiliary information sub-fields

Type	Value	Description
Backup domain master ID	06_{16}	A sub-field carrying the DEVICE_ID and the REGID of the node assigned as a backup domain master for the domain, as described in clause 8.8.5.7
Timer-related domain info	07_{16}	A sub-field carrying timer-related domain information, as described in clause 8.8.5.8
Reserved	08_{16}	Reserved by ITU-T
Registration code	09_{16}	A sub-field indicating registration code to register nodes to which domain name cannot be provided by the user, as described in clause 8.8.5.9
DOD update	$0A_{16}$	The new value of DOD
Reserved	$0B_{16}$	Used in Amendment 1 to this Recommendation
Reserved	$0C_{16}$	Used in Amendment 1 to this Recommendation
NMK_DB_update	$0D_{16}$	The NMK or DB key are going to be updated
Reserved	$0E_{16}$ to $7F_{16}$	Reserved by ITU-T

29) Clause 8.8.5.2, Domain name sub-field

Revise the text of clause 8.8.5.2 "Domain name sub-field" as follows:

8.8.5.2 Domain name sub-field

The format of the domain name sub-field shall be as presented in Table 8-74. The length of the sub-field data is 36~~2~~ octets.

Table 8-74 – Format of domain name sub-field

Field	Octet	Bits	Description
Type	0	[6:0]	Set to 01_{16}
ModificationFlag		[7]	This flag shall be set to zero
Length	1	[7:0]	Set to $36~~2~~_{10}$
Domain name	2 to 33	[255:0]	32-octet domain name represented in ASCII characters (Note 1)
NumNodes	34	[7:0]	Number of nodes that are registered with the domain master, represented as an unsigned integer in the range from 1 to 249
NodeParam	35 to 37	[23:0]	A 24-bit field describing parameters and capabilities of the DM of the domain. It shall be formatted as described in Table 8-47.1
NumDmVersionTLVs	38	[7:0]	Number of versioning (N) TLVs included in this message corresponding to the versioning information of the DM. Set to 0 if no Versioning TLVs are included which implies that the node only supports version 0 of ITU-T G.9960 and ITU-T G.9961. If N>0, the first TLV shall be the TLV corresponding to ITU-T version

Table 8-74 – Format of domain name sub-field

Field	Octet	Bits	Description
DmVersionTLVs	Var	Var	Information related to the version and capabilities of the registering node. The format of this field shall be as described in Table 8-16.1 (Note 2)

NOTE 1 – The ASCII characters shall be mapped onto the bytes of the domain name in the following way:
- the LSB of the 7-bit ASCII character is mapped onto bit b0 of the corresponding byte of the domain name;
- the MSB of all bytes shall be set to zero;
- the first ASCII character of the domain name shall be mapped on the least significant byte of the domain name (e.g., if the domain name is "Network", the first ASCII character is letter "N" that shall be mapped at byte 0 of the domain name);
- if the number of provided ASCII characters is less than 32, the rest of the domain name field bytes shall be set to 00_{16}.

NOTE 2 – A domain master indicating support for a certain version of a Recommendation shall mean that it also supports all the earlier versions of that Recommendation.

30) Clause 8.8.5.5, PSD-related domain info sub-field

Revise the text of clause 8.8.5.5 "PSD-related domain info sub-field" as follows:

8.8.5.5 PSD-related domain info sub-field

The format of the PSD-related domain info sub-field shall be as presented in Table 8-77. The length of the sub-field data is variable.

Table 8-77 – Format of PSD-related domain info sub-field

Field	Octet	Bits	Description
Type	0	[6:0]	Set to ~~03~~04$_{16}$
ModificationFlag		[7]	This flag shall be set to one.
Length	1	[7:0]	Length of the field in octets (range 3-199).
~~DmVersion~~Reserved	2	[7:0]	Reserved by ITU-T (Note 1)~~0 – Domain master supports version 0 of ITU-T G.9960 and ITU-T G.9961~~ ~~All other values of this field are reserved by ITU-T for indicating support for future versions of the Recommendation (Note 1)~~
Regional PSD shaping mask	3	[0]	0, when PSD shaping descriptor sub-field is not ~~used~~ present 1, when PSD shaping descriptor sub-field is ~~used~~ present
Regional SM		[1]	0, when ~~sub-carrier masking~~SM descriptor sub-field is not ~~used~~ present 1, when ~~sub-carrier masking~~SM descriptor sub-field is ~~used~~ present
Regional TX power limit		[2]	0, when ~~standard transmit~~TX power limit sub-field is ~~used~~not present (see clause 7.2.6 of [ITU-T G.9960])

Table 8-77 – Format of PSD-related domain info sub-field

Field	Octet	Bits	Description
			1, when TX power limit sub-field is ~~used~~ present
Regional Amateur radio bands		[3]	0, when ~~all international~~ Amateur radio band~~s are masked~~ descriptor sub-field is not present 1, when ~~one or more bands are not masked~~ Amateur radio band descriptor sub-field is present
Symbol boost indicator		[4]	0, when Symbol boost parameters sub-field is not present 1, when Symbol boost parameters sub-field is present
Reserved		[7:5~~4~~]	Reserved by ITU-T (Note 1~~2~~)
Amateur radio band descriptor	~~4 & 5~~variable	[9:0]	~~Zero octets~~This field shall not be present if the regional Amateur radio bands field~~bit 3 of octet 3~~ is set to zero, otherwise it represents a bit map representing usage of international amateur bands (0 = masked, 1 = unmasked). The LSB represents the lowest band (1.8-2.0 MHz), the second LSB represents the second lowest band (3.5-4.0 MHz), etc. Masked amateur bands are part of RMSC (see clause 7.1.4.2.1 of [ITU-T G.9960])
Reserved		[15:10]	Reserved by ITU-T (Note 1~~2~~)
TX power limit	~~6~~variable	[7:0]	~~Zero octets~~This field shall not be present if the regional TX power limit field~~bit 2 of octet 3~~ is set to zero, otherwise it represents the value of maximum transmit power in dBm, represented as 0.1dBm per unit
PSD shaping descriptor	~~7 to (6+L)~~variable	[(8*L) – 1:0]	~~Zero octets~~This field shall not be present if the regional PSD shaping mask field~~bit 0 of octet 3~~ is set to zero, otherwise see Table 8-78 (Note 2~~3~~)
SM descriptor	~~(7+L) to (6+L+M)~~ variable	[(8*M) – 1:0]	~~Zero octets~~This field shall not be present if the regional SM field~~bit 1 of octet 3~~ is set to zero, otherwise see Table 8-79. Masked bands are part of RMSC (see clause 7.1.4.2.1 of [ITU-T G.9960]) (Note 3~~4~~)
Symbol boost parameters	variable	[7:0]	This field shall not be present if the symbol boost indicator field is set to zero, otherwise see Table 8-79.1
Minimal bandplan	variable	[2:0]	This field indicates the value of the minimal bandplan capability for a node that is allowed to register to the domain. It shall be formatted as shown in Table 7-10 of [ITU-T G.9960]. (Note 5). Also see clause 7.4.9 of [ITU-T G.9962]
Maximal bandplan		[5:3]	This field indicates the value of the maximal bandplan capability for a node that is allowed to register to the domain. It shall be formatted as shown in Table 7-10 of [ITU-T G.9960]. (Note 5). Also see clause 7.4.10 of [ITU-T G.9962]
Reserved		[7:6]	Reserved by ITU-T (Note 1)

Table 8-77 – Format of PSD-related domain info sub-field

Field	Octet	Bits	Description
~~NOTE 1 – A domain master indicating support for a certain version of this Recommendation shall mean that it also supports all the earlier versions of the Recommendation.~~ NOTE 1~~2~~ – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver. NOTE 2~~3~~ – The value of L equals to the value of the first octet of the PSD shaping descriptor multiplied by 3 plus 1. The value of M equals to the value of the first octet of the SM descriptor multiplied by 3 plus 1. NOTE 3~~4~~ – The SM is intended to incorporate masked sub-carriers defined by the regional Annex to comply with local regulations and masked sub-carriers defined by the user or service provider to facilitate local deployment practices. NOTE 5 – A node is allowed to register to a domain only if its bandplan is within the range indicated by the Minimal bandplan and Maximal bandplan.			

…

31) New clause 8.8.5.11, NMK_DB_update sub-field

Add new clause 8.8.5.11 "NMK_DB_update sub-field" as follows:

8.8.5.11 NMK_DB_update sub-field

The format of the NMK_DB_update sub-field shall be as presented in Table 8-85.1. The length of the sub-field data is 3 octets.

Table 8-85.1 – Format of NMK_DB_update sub-field

Field	Octet	Bits	Description
Type	0	[6:0]	Set to $0B_{16}$.
ModificationFlag		[7]	This flag shall be set to one.
Length	1	[7:0]	Set to 03_{16}.
KEY_update	2	[0]	If set to 0 the DB key is going to be updated If set to 1 the NMK key is going to be updated
Reserved		[7:1]	Reserved by ITU-T (Note)
UpdateMacCycle	3 and 4	[15:0]	This field contains the MAP sequence number that the updated DB key or NMK key shall start to be used
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

32) Clause 8.8.6, MAP schedule persistence publication

Revise the text of clause 8.8.6 "MAP schedule persistence publication" as follows:

8.8.6 MAP schedule persistence publication

Schedule persistence shall be indicated in the MAP using the following counters:

– CSLT (current schedule life time): The validity of the currently applied schedule shall be CSLT+1 MAC cycles.

– FSLT (future schedule life time): The validity of the future schedule, to be applied right after the current schedule persistence period ends, shall be FSLT+1 MAC cycles.

The CSLT and FSLT counters shall only apply to persistent TXOPs (see clause 8.3.1.2). A single set of CSLT and FSLT counters shall be used for all the persistent TXOPs and is indicated in the MAP header (see MAP frame format in clause 8.8.3, Table 8-62).

To apply a persistent schedule, the domain master shall use the CSLT counter. CSLT is set to the desired duration of the persistence period in MAC cycles minus one. Once CSLT is set to a non-zero value, it shall not be decreased by more than one in every successive MAP.

To terminate a persistent schedule, the domain master shall decrease the CSLT by one in every successive MAP until it reaches zero whilst maintaining FSLT = 0.

If the domain master intends to continue with the current persistent schedule, it may keep or increase the validity of the currently applied persistent schedule by maintaining or increasing the value of the CSLT counter in subsequent MAP messages. FSLT shall be set to zero in this case. If the domain master intends to change the persistent schedule, it shall set the FSLT counter to a non-zero value. The CSLT counter shall then be decremented by one each MAC cycle and the current persistent schedule shall only be valid while the CSLT counter is greater than or equal to zero. Once FSLT is set to a non-zero value, the future schedule is published.

…

33) Clause 8.9.2.2, Multicast acknowledgement procedure

Revise the text of clause 8.9.2.2 "Multicast acknowledgement procedure" as follows:

8.9.2.2 Multicast acknowledgement procedure

…

$T_{sequence} = T_{frame} + T_{AIFG-D} + M \times T_{Mc-ACK} + (M-1) \times T_{McAIFG}$ (if NACK signalling is not used) (see Figure 8-45),

$T_{sequence} = T_{frame} + T_{AIFG-D} + M \times T_{Mc-ACK} + (M-1) \times T_{McAIFG} + T_{AIFG-D} + T_{NACK}$, if NACK signalling is used (see Figure 8-46),

where T_{frame} is the duration of the multicast frame and M is the number of nodes assigned for Mc-ACK, which shall be at least 1 (see clause 8.9.2.1).

Table 8-86 summarizes the types of Mc-ACK depending on RPRQ settings (see also Table 7-11 of [ITU-T G.9960]).

…

34) Clause 8.9.4.2, Transmitter variables and control flags, and clause 8.9.4.3, Receiver variables and control flags

Revise the text of clause 8.9.4.2 "Transmitter variables and control flags" and clause 8.9.4.3 "Receiver variables and control flags" as follows:

8.9.4.2 Transmitter variables and control flags

The transmission window is formed by the segments that are eligible for transmission; each segment is identified by its SSN.

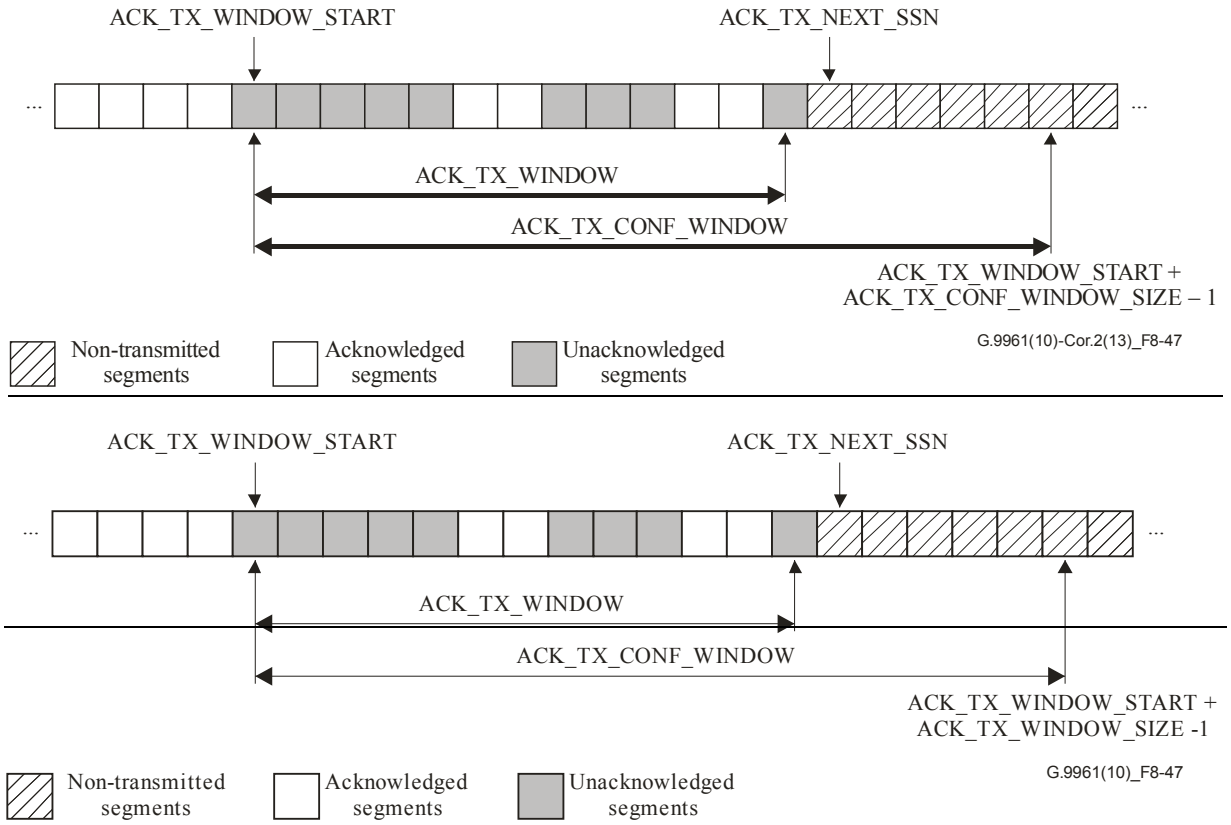


Figure 8-47 – Transmission window

ACK_TX_WINDOW_START is the SSN of the oldest unacknowledged segment: all segments with SSNs up to (ACK_TX_WINDOW_START – 1) have been acknowledged. A segment is called unacknowledged if it has been transmitted but no positive acknowledgement has been received.

ACK_TX_CONF_WINDOW is the maximum range of SSNs corresponding to segments the transmitter is permitted to send. This range is defined by ACK_TX_WINDOW_START and ACK_TX_CONF_WINDOW_SIZE as shown in Figure 8-47. ACK_TX_CONF_WINDOW_SIZE is a parameter that depends on the connection and shall be initialized as described in clause 8.12. ACK_TX_CONF_WINDOW_SIZE shall not exceed ACK_MAX_WINDOW_SIZE.

ACK_TX_WINDOW is the range of SSNs between the oldest unacknowledged segment and the newest unacknowledged segment, inclusive. This range is defined by ACK_TX_WINDOW_START and ACK_TX_NEXT_SSN, as shown in Figure 8-47, and may contain acknowledged and unacknowledged segments. The run-time size of the ACK_TX_WINDOW is ACK_TX_NEXT_SSN – ACK_TX_WINDOW_START.

ACK_TX_NEXT_SSN is the SSN of the next segment to send. This value shall belong to the interval ACK_TX_WINDOW_START to (ACK_TX_WINDOW_START + ACK_TX_CONF_WINDOW_SIZE), inclusive.

ACK_BLOCK_LIFETIME is the maximum time interval a segment shall be kept in the ACK_TX_WINDOW after this segment was transmitted the first time. If the segment is not acknowledged by the receiver within ACK_BLOCK_LIFETIME, the segment shall be discarded. Multiple retransmissions are allowed during this time.

NOTE – The value of ACK_BLOCK_LIFETIME may affect the latency and jitter of a flow. When selecting a value for it, implementers should take into account the delay and delivery (effect of losing LPDUs) requirements of the flow associated with the connection.

ACK_TX_RESET is the transmission window reset flag. When set to one, the transmitter state machine is in TX_RESET state and no segments shall be transmitted. When set to zero, the transmitter state machine is not in TX_RESET state and segments may be transmitted.

8.9.4.3 Receiver variables and control flags

The reception window is formed by the segments that can be accepted in the receiver to wait for retransmission of missing segments.

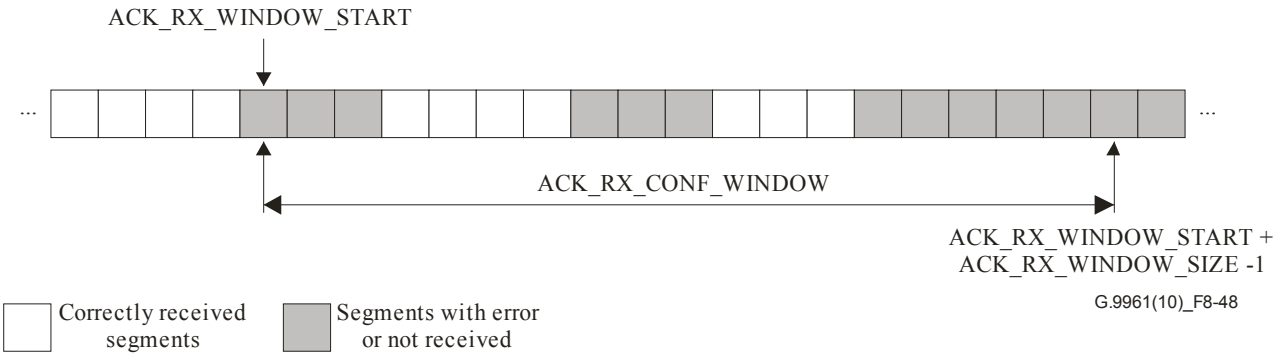


Figure 8-48 – Reception window

ACK_RX_WINDOW_START is the SSN of the oldest segment received in error or not received: All segments with SSNs up to (ACK_RX_WINDOW_START – 1) have been received correctly or have been discarded by the transmitter.

ACK_RX_CONF_WINDOW is the maximum range of SSNs corresponding to segments that the receiver is expecting to receive and accept. This range is defined by ACK_RX_WINDOW_START and ACK_RX_CONF_WINDOW_SIZE as shown in Figure 8-48. ACK_RX_CONF_WINDOW_SIZE shall be greater than or equal to the number of segments that the receiver can buffer for a connection as described in clause 8.12. ACK_RX_CONF_WINDOW_SIZE shall not exceed ACK_MAX_WINDOW_SIZE.

ACK_RX_RESET is the reception window reset flag. When set to one, the receiver state machine is not in RX_WIN_SYNC state and received segments shall be discarded. When set to zero, the received segments may be accepted.

35) Clause 8.9.5.3, Acknowledgement protocol state machine for unicast transmission

Revise the text of clause 8.9.5.3 "Acknowledgement protocol state machine for unicast transmission" as follows:

8.9.5.3 Acknowledgement protocol state machine for unicast transmission

The protocol to be used between nodes to facilitate unicast transmission with acknowledgements is initialized as presented in Figure 8-50 (which shows the case where no transmissions have been lost) and Figure 8-51 (which shows an example of a case where some transmissions have been lost). The procedure includes the establishment of the connection as defined in clause 8.12. The initialization is based on the exchange of ACK_TX_RESET and ACK_RX_RESET flags. ACK_TX_RESET is sent in the PHY-frame header of the MSG frame (see clause 7.1.2.3.2.2.18 of [ITU-T G.9960]). ACK_RX_RESET is sent in the PHY-frame header of the ACK frame (see clauses 7.1.2.3.2.3.5 and 7.1.2.3.2.3.6 of [ITU-T G.9960]) according to clause 8.9.1.1 or clause 8.9.1.2.

A transmitting node may be in any one of the following states: TX_RESET, TX_WAIT_SYNC or TX_WIN_SYNC. A receiving node may be in any one of the following states: RX_RESET, RX_WAIT_SYNC or RX_WIN_SYNC.

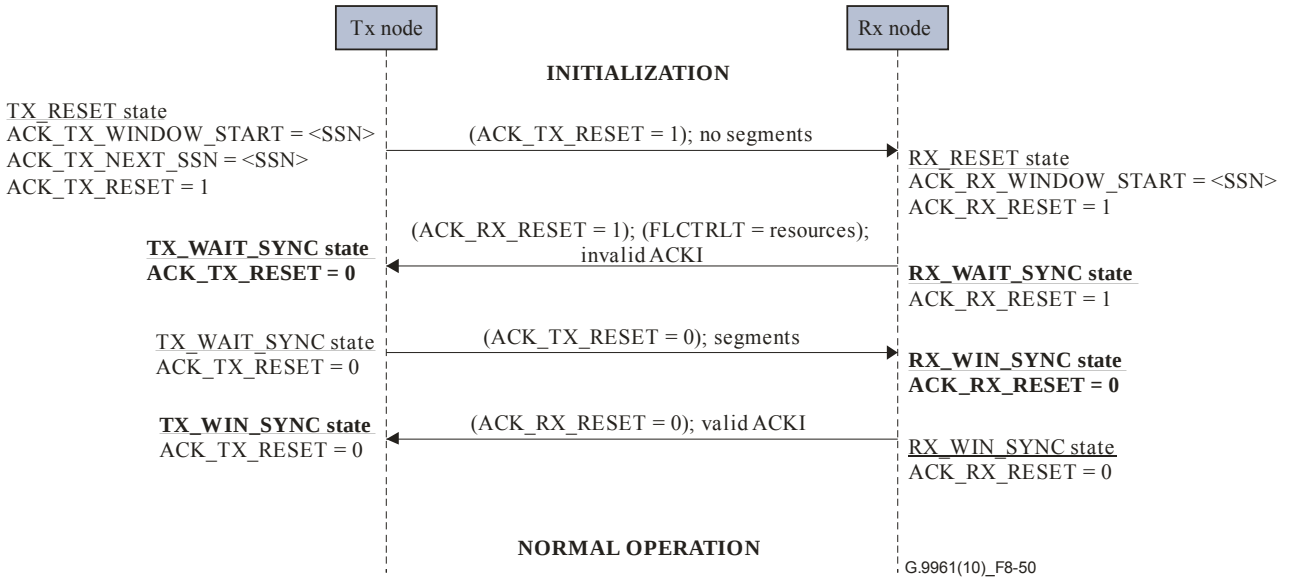


Figure 8-50 – Initialization of the acknowledgement protocol

First, the transmitting and receiving nodes state machines shall be in reset state (TX_RESET and RX_RESET). The flag ACK_TX_RESET = 1 shall be transmitted prior to the PHY frame carrying the first data segment of the established connection. This flag indicates that the transmitting node is in TX_RESET state. In TX_RESET state, ACK_TX_WINDOW_START and ACK_TX_NEXT_SSN shall be an arbitrary <SSN> value set by the transmitter.

Upon reception of ACK_TX_RESET = 1 in any state, the receiver shall reset its ARQ state machine and reply with the flag ACK_RX_RESET = 1 and shall indicate the availability of resources (see clause 8.12). This flag indicates that the receiver is in RX_RESET state. In RX_RESET state, ACK_RX_WINDOW_START shall be set to the <SSN> value specified by the transmitter in the START_SSN field. After sending the flag, if a status report was indicated in the flow control information (see clause 8.12) the receiving node shall transition to RX_WAIT_ SYNC state. Otherwise, the receiver shall remain in RX_RESET state.

Segments of the established connection shall not be sent while the transmitting node is in TX_RESET state.

Once in TX_RESET state, if the receiver indicated the availability of resources (see clause 8.12) and after receiving the flag ACK_RX_RESET = 1, the transmitter shall set the flag ACK_TX_RESET to zero and transition into the TX_WAIT_SYNC state. Segments of the established connection may be sent in TX_WAIT_SYNC state.

If in TX_RESET state the transmitter does not receive the ACK_RX_RESET = 1 in the requested Imm-ACK, the transmitter shall resend ACK_TX_RESET = 1. If the receiver signalled a hold time, the transmitter shall wait that time before resending the PHY frame with ACK_TX_RESET = 1. If the receiver indicated the unavailability of resources (see clause 8.12), the transmitter shall remain in the TX_RESET state keeping ACK_TX_RESET = 1. Then, the initialization of the acknowledgement protocol for that connection cannot be completed.

If in TX_RESET state the transmitting node receives an ACK frame with ACK_RX_RESET = 0, the transmitter shall ignore this ACK frame and resend ACK_TX_RESET = 1.

After resending two times ACK_TX_RESET = 1 in TX_RESET state, the segments of the established connection shall be discarded and the initialization of the acknowledgement protocol for the connection cannot be completed.

If the receiving node receives ACK_TX_RESET = 0 while being in RX_WAIT_SYNC state, it shall set the flag ACK_RX_RESET to zero, process the segments included in the PHY frame as described in clause 8.9.5.3.2, transition into RX_WIN_SYNC state and send ACK_RX_RESET = 0 to the transmitter.

The transmitting node shall transition from TX_WAIT_SYNC state into TX_WIN_SYNC state after the reception of an ACK frame with ACK_RX_RESET = 0. The transmitter shall process the ACK information as described in clause 8.9.5.3.1.

If in TX_WAIT_SYNC state the transmitting node does not receive the ACK_RX_RESET = 0 in the requested Imm-ACK or after inferring that the acknowledgement is lost or not sent (see clause 8.9.5.2), the transmitter shall resend ACK_TX_RESET = 0.

If in TX_WAIT_SYNC state the transmitting node receives ACK_RX_RESET = 1 with a status report in the flow control information, it shall resend the PHY frame with ACK_TX_RESET = 0. If the flow control information contains a valid hold time (see clause 8.12), the transmitter shall wait that time before resending the PHY frame with ACK_TX_RESET = 0.

After resending two times ACK_TX_RESET = 0 in TX_WAIT_SYNC state, the segments of the established connection shall be discarded and the initialization of the acknowledgement protocol for the connection cannot be completed.

When transmitting and receiving nodes are in TX_WIN_SYNC state and RX_WIN_SYNC state, the initialization of the acknowledgement protocol is completed. After the initialization, the protocol enters its normal operation.

When the transmitter is in a state where it can start sending the segments, it shall transmit all segments beginning with the <SSN> it has specified in the START_SSN field.

NOTE – This allows the receiver to flush all pending segments in its queue that were received before the reception of ACK_TX_RESET = 1.

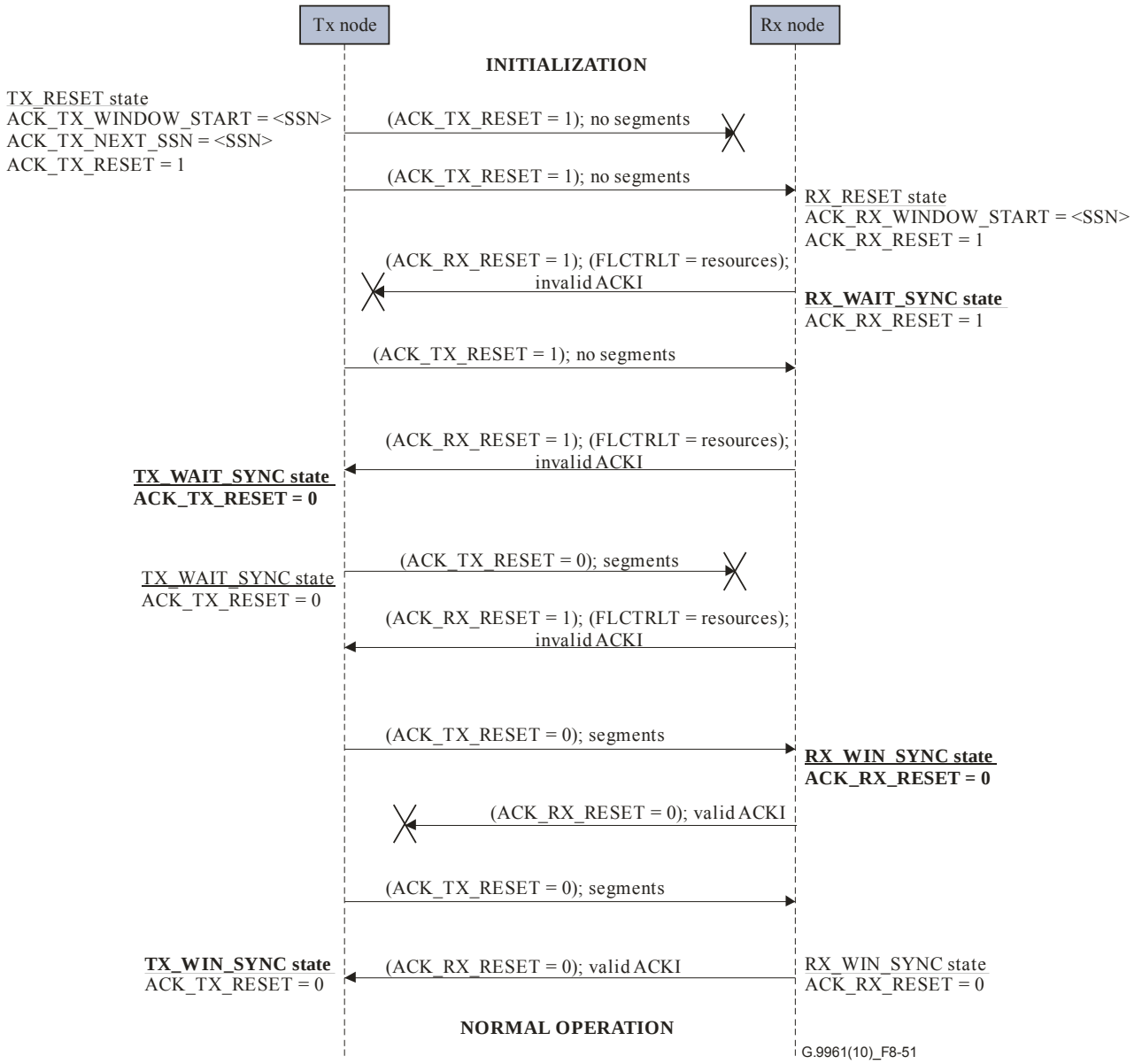


Figure 8-51 – Detailed initialization of the Acknowledgement protocol

If in TX_WIN_SYNC state the transmitting node receives ACK_RX_RESET = 1, the receiving node is in RX_RESET state. In this case, if the flow control information coveys a status report or a valid hold time, the transmitting node may transition into the TX_RESET state, send ACK_TX_RESET = 1 and follow again the initialization procedure described in this clause; or it may discard the segments of the established connection and terminate that connection.

If in any state the transmitter receives flow control information indicating the unavailability of resources (see clause 8.12), the transmitter shall discard the segments of the established connection, terminate that connection and transition into the TX_RESET state.

36) Clause 8.9.5.3.1, Transmission window operation

Revise the text of clause 8.9.5.3.1 "Transmission window operation" as follows:

8.9.5.3.1 Transmission window operation

Comparisons of SSNs that appear in this clause assume a previous normalization as described in clause 8.9.5.1. The term LSSN is used in this clause to refer to the value conveyed in the ACK

frame fields LSSN (see clause 7.1.2.3.2.3.9.1.6 of [ITU-T G.9960]) and MNMT_LSSN (clause 7.1.2.3.2.3.9.1.3.1 of [ITU-T G.9960]).

The transmitter shall maintain an ACK_TX_WINDOW per connection established with the receiver.

In TX_WAIT_SYNC or TX_WIN_SYNC state, when an acknowledgement with ACK_RX_RESET = 0 is received, the transmitter shall process the conveyed acknowledgement data. The transmitter shall discard the acknowledgement data if the LSSN does not satisfy any of the following conditions:

- ACK_TX_WINDOW_START ≤ LSSN < ACK_TX_NEXT_SSN;
- LSSN is equal to the *N* LSB bits of ACK_TX_NEXT_SSN and there is no valid selective acknowledgement information (the ACKI field is set according to ~~zero,~~ ~~see~~clause 7.1.2.3.2.3.9.1.7 of [ITU-T G.9960] to indicate that all data units have been received with errors).

NOTE – The previous conditions assure that either the LSSN is contained in ACK_TX_WINDOW or that the receiver is acknowledging all the contents of it. Then, ACK_RX_WINDOW_START is equal to ACK_TX_NEXT_SSN.

Otherwise, the transmitter shall continue processing the received acknowledgement information.

If an acknowledgement message is not discarded, the transmitter shall interpret the contents (see clause 7.1.2.3.2.3 of [ITU-T G.9960]) and update the ACK_TX_WINDOW as described below.

The transmitter shall change to done state all the segments with SSNs that satisfy the condition ACK_TX_WINDOW_START ≤ SSN < LSSN and shall then update ACK_TX_WINDOW_START to the SSN whose *N* LSB bits are equal to the received LSSN. After that, the transmitter shall interpret the contents of the selective acknowledgement information (ACKI) and shall change to done state the indicated segments whose SSNs fulfil the condition ACK_TX_WINDOW_START ≤ SSN < ACK_TX_NEXT_SSN.

ACK_TX_WINDOW_START and ACK_RX_WINDOW_START shall be kept synchronized so that the receiver never awaits the reception of a segment that has been removed from the transmission window (passed to discarded state) and has never been received correctly in the receiver side. Therefore, the oldest pending segment flag (OPSF) is used to avoid this. The transmitter shall always set the OPSF of the oldest segment pending acknowledgement (not in done or discarded state) to one to inform the receiver. The OPSF of an LPDU shall not be modified between the transmission of a PHY-frame and the reception of the Imm-ACK in case it was requested.

When a segment is discarded after ACK_BLOCK_LIFETIME (see clause 8.9.4.2) the transmitting node shall proceed to the next segment that is not in the done or discarded state and shall set its OPSF to one.

When ACK_TX_WINDOW is equal to ACK_TX_CONF_WINDOW and all the segments in ACK_TX_WINDOW are in done or discarded state and the LPDU corresponding to ACK_TX_WINDOW_START is in the discarded state, the transmitting node shall transition into the TX_RESET state, send ACK_TX_RESET = 1 and reset the connection (see clause 8.12.7).

37) Clause 8.10.1, Management message format (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.10.1 "Management message format" (from ITU-T G.9961 Corrigendum 1) as follows:

8.10.1 Management message format

…

Table 8-87 – Format of management messages

	Content	Octet	Bits	Description
MMH	Length	0 to 2	[11:0]	Length (LG) of the MMPL segment in octets, encoded as a 12-bit unsigned integer. The value of LG shall not exceed 1492
	OPCODE		[23:12]	12-bit OPCODE, indicates message type (Note 1)
	~~STD Version~~Reserved	3	[7:0]	Reserved by ITU-T (Note 4).~~The format of this message is according to the specified G.9961 version~~
	Number of segments	4	[3:0]	Number of segments minus 1, represented as an unsigned integer between 0 and F_{16}. It shall be set to 0_{16} if the message is not segmented (Note 2)
	Segment number		[7:4]	Segment number, represented as an unsigned integer between 0_{16} and F_{16}; set to 0_{16} for the first segment and if message is not segmented (Note 2)
	Sequence number	5 and 6	[15:0]	Sequence number of the segmented message in a format of 16-bit unsigned integer (Note 2, Note 3)
	Repetition number	7	[3:0]	Repetition number of the ~~segmented~~ message formatted as a 4-bit unsigned integer whose initial value is 0. Each time a ~~segment~~ message is retransmitted by the originating node this field shall be incremented. See clause 8.10.1.2
	FSB		[4]	Force Sequence Bit. See clause 8.10.1.2
	Reserved		[7:5]	Reserved by ITU-T (Note 4)
MMPL	Message Parameters	8 to (LG+7)	[(8×LG–1):0]	Depends on the OPCODE, see Table 8-88

NOTE 1 – The OPCODES are defined in Table 8-88.

NOTE 2 – This field is not applicable for a MAP message, and shall be set to zero.

NOTE 3 – The meaning of the sequence number depends on the OPCODE. See clause 8.10.1.2.

NOTE 4 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.

8.10.1.1 Management message OPCODEs

Management message OPCODEs are formatted as 12-bit unsigned integers. Valid values of OPCODEs are presented in Table 8-88. OPCODEs are categorized (typically by their associated protocol or procedure) according to the value of their eight MSBs.

Table 8-88 – OPCODEs of management messages

Category	Message name	OPCODE (hex)	Description	MMPL Reference
Admission (01X)	ADM_NodeRegistrRequest.req	010	Registration request	Clause 8.6.1.1.4.1
	ADM_DmRegistrResponse.cnf	011	Registration response	Clause 8.6.1.1.4.2
	ADM_NodeResignRequest.req	012	Resignation request	Clause 8.6.1.1.4.3
	ADM_DmResign.cnf	013	Registration announcement	Clause 8.6.1.1.4.4
	ADM_DmForcedResign.req	014	Forced resignation request	Clause 8.6.1.1.4.5
	ADM_NodeReRegistrRequest.req	015	Periodic re-registration request	Clause 8.6.1.1.4.6
	ADM_DmReRegistrResponse.cnf	016	Periodic re-registration response	Clause 8.6.1.1.4.7
	ADM_DmReRegistrInitiate.ind	017	Re-registration initiation request	Clause 8.6.1.1.4.8
	ADM_NodeReportMAPD.ind	018	Report the reception of a MAP-D with matching domain name	Clause 8.6.6.1.4.1
	ADM_NodeReportMAPA.ind	019	Report the reception of a MAP-A with matching DNI	Clause 8.6.6.1.4.2
AKM (02X)	AUT_~~NodeRequest~~NodeAuthentication.req	020	Request for authentication	Clause 9.2.5.1.1
	AUT_Promp.ind	021	Delivers authentication prompt	Clause 9.2.5.1.2
	AUT_Verification.rsp~~es~~	022	Authentication prompt verification	Clause 9.2.5.1.3
	AUT_Confirmation.cnf	023	Authentication confirmation message	Clause 9.2.5.1.4
	AKM_KeyRequest.req	024	Request for secure communication with another node(s)	Clause 9.2.5.2.1
	AKM_NewKey.req	025	Message delivers the encryption key to the supplicant node	Clause 9.2.5.2.2
	AKM_KeyConfirmation.req	026	Message delivers the encryption key to the addressee node(s)	Clause 9.2.5.2.4
	AKM_KeyUpdate.req	027	Request for re-authentication and update the keys	Clause 9.2.5.3.1
	AKM_NewKey~~Ack~~.cnf	028	Addressee confirmation that encryption key was delivered	Clause 9.2.5.2.3
	SC_DMRes.req	029	Request to resign a node from the domain	Clause 9.2.5.2.5
	SC_DMRes.cnf	02A	Confirmation of resignation from the domain master	Clause 9.2.5.2.6
	AKM_~~Key~~AddClient~~Request~~.req	02B	Request to join a node to a multicast group	Clause 9.2.5.2.1.1
	AKM_DomainKeyUpdate.ind	02C	Indication to update the domain-wide encryption keys	Clause 9.2.5.3.2

Table 8-88 – OPCODEs of management messages

Category	Message name	OPCODE (hex)	Description	MMPL Reference
	AKM_NewKey.ind	02D	Indication that the new encryption key is available for use	Clause 9.2.5.2.7
	AKM_DomainKeyUpdate.req	02E	Request to update the domain-wide encryption key, from SC to DM	Clause 9.2.5.3.3
	AKM_DomainKeyUpdate.cnf	02F	Confirmation for the request to update the domain-wide encryption key, from DM to SC	Clause 9.2.5.3.4
Topology maintenance (03X)	TM_NodeTopologyChange.ind	030	Topology report from a node	Clause 8.6.4.2.1
	TM_NodeTopologyChange.req	031	Request sent by the domain master to a particular node requesting its topology report	Clause 8.6.4.3.2
	TM_NodeTopologyChange.cnf	032	Topology report from a node in response to the message TM_NodeTopologyChange.req	Clause 8.6.4.3.3
	TM_DomainRoutingChange.ind	033~~1~~	Optimal routing update from the domain master	Clause 8.6.4.3.5
	TM_ReturnDomainRouting.req	034~~2~~	Request for routing update from the node to the domain master	Clause 8.6.4.3.6
	TM_ReturnDomainRouting.cnf	035~~3~~	Reply on routing request by the Domain master	Clause 8.6.4.3.7
	TM_DMBackup.ind	036~~4~~	Topology report from a node sent by backup domain master to a node	Clause 8.6.4.3.4
Power-line coexistence with alien networks (04X)	Reserved for use by [ITU-T G.9972]			
Multicast binding (05X)	MC_GrpInfoUpdate.ind	050	Multicast binding information update	Clause 8.16.5.1
	MC_GrpInfoUpdate.cnf	051	Multicast binding information update confirmation	Clause 8.16.5.2
	MC_GrpRemove.req	052	Multicast leave request from the transmitter	Clause 8.16.5.3
	MC_GrpRemove.cnf	053	Multcast leave confirmation from the receiver	Clause 8.16.5.4
	DMC_Path.req	054	DLL multicast path establishment request	Clause 8.17.6.1
	DMC_Path.cnf	055	DLL multicast path establishment confirmation	Clause 8.17.6.2
	DMC_PathReject.cnf	056	DLL multicast path establishment rejection	Clause 8.17.6.3

Table 8-88 – OPCODEs of management messages

Category	Message name	OPCODE (hex)	Description	MMPL Reference
	DMC_EnforcePath.req	057	DLL multicast enforced path establishment request	Clause 8.17.6.4
	DMC_ReleasePath.req	058	A request to release a DLL multicast client node from its MSID	Clause 8.17.6.5
	DMC_ReleasePath.cnf	059	Confirmation of the release of a DLL multicast client node from its MSID	Clause 8.17.6.6
	DMC_PathAlive.ind	05A	DLL multicast path alive indication	Clause 8.17.6.7
	DMC_BrokenLink.ind	05B	DLL multicast broken link indication	Clause 8.17.6.8
Domain master selection and backup domain master (06X)	DM_Handover.req	060	Domain master role handover request	Clause 8.6.6.5.1
	DM_Handover.cnf	061	Domain master role handover confirmation	Clause 8.6.6.5.2
	DM_Handover.ind	062	Domain state update	Clause 8.6.6.5.3
	DM_Handover.rsp	063	Domain state update confirmation	Clause 8.6.6.5.4
	DM_BackupAssign.req	064	Backup domain master assignment request	Clause 8.6.5.2
	DM_BackupAssign.cnf	065	Backup domain master assignment confirmation	Clause 8.6.5.2
	DM_BackupData.ind	066	Domain state update	Clause 8.6.5.2
	DM_BackupRelease.req	067	Release of a backup domain master	Clause 8.6.5.2
	DM_BackupRelease.cnf	068	Backup domain master release confirmation	Clause 8.6.5.2
Channel estimation (07X)	CE_ProbeSlot~~Request.ind~~Assign.req	070	Channel estimation bandwidth assignment request	Clause 8.11.7.1
	CE_ProbeSlotRelease.~~ind~~req	071	Channel estimation bandwidth release request	Clause 8.11.7.2
	CE_ParamUpdate.~~ind~~req	072	Channel estimation parameters update request	Clause 8.11.7.3
	CE_ParamUpdateRequest.ind	073	Request for c~~C~~hannel estimation parameter ~~request~~update	Clause 8.11.7.4
	CE_PartialBatUpdate.~~ind~~req	074	Partial BAT update ~~indication~~request	Clause 8.11.7.5
	CE_ACESymbols.ind	075	Request for an ACE symbol attachment	Clause 8.11.7.6
	CE_ProbeSlotAssign.cnf	076	Channel estimation bandwidth assignment confirmation	Clause 8.11.7.7
	CE_ProbeSlotRelease.cnf	077	Channel estimation bandwidth release confirmation	Clause 8.11.7.8

Table 8-88 – OPCODEs of management messages

Category	Message name	OPCODE (hex)	Description	MMPL Reference
	CE_ParamUpdate.cnf	078	Channel estimation parameters update confirmation	Clause 8.11.7.9
	CE_PartialBatUpdate.cnf	079	Partial BAT update confirmation	Clause 8.11.7.10
Neighbouring networks coordination (08X)	For further study	For further study	For further study	For further study
Inactivity scheduling (09X)	IAS_LongInactivity.req	090	Long inactivity scheduling request	Clause 8.3.6.1.1
	IAS_LongInactivity.cnf	091	Long inactivity scheduling confirmation	Clause 8.3.6.1.1
	IAS_ShortInactivity.req	092	Short inactivity scheduling request	Clause 8.3.6.2.1
	IAS_ShortInactivity.cnf	093	Short inactivity scheduling confirmation	Clause 8.3.6.2.1
Flow establishment (0AX)	~~CL_EstablishFlow.req~~Reserved	0A0	~~Flow establishment request~~Reserved by ITU-T	~~Clause 8.6.2.3.1~~
	~~CL_EstablishFlow.cnf~~Reserved	0A1	~~Flow establishment confirmation~~Reserved by ITU-T	~~Clause 8.6.2.3.2~~
	FL_AdmitFlow.req	0A2	Flow admission request	Clause 8.6.2.3.8
	FL_AdmitFlow.cnf	0A3	Flow admission confirmation	Clause 8.6.2.3.9
	FL_AdmitFlow.ind	0A4	Flow admission indication	Clause 8.6.2.3.10
	FL_AdmitFlow.rsp	0A5	Flow admission acknowledgement	Clause 8.6.2.3.18
	FL_OriginateFlow.req	0A6~~4~~	Flow origination request	Clause 8.6.2.3.6
	FL_OriginateFlow.cnf	0A7~~5~~	Flow origination confirmation	Clause 8.6.2.3.7
Flow maintenance (0BX)	FL_ModifyFlowParameters.req	0B0	Modification of flow parameters and allocation	Clause 8.6.2.3.11
	FL_ModifyFlowParameters.cnf	0B1		Clause 8.6.2.3.12
	FL_ModifyFlowParameters.ind	0B2		Clause 8.6.2.3.15
	FL_ModifyFlowAllocations.req	0B3	Modification of flow allocation	Clause 8.6.2.3.16
	FL_ModifyFlowAllocations.cnf	0B4		Clause 8.6.2.3.17
Flow termination (0CX)	~~CL_TerminateFlow.req~~Reserved	0C0	~~Flow termination request and confirmation~~Reserved by ITU-T	~~Clause 8.6.2.3.3~~
	~~CL_TerminateFlow.cnf~~Reserved	0C1	Reserved by ITU-T	~~Clause 8.6.2.3.4~~
	~~CL_FlowTerminated.ind~~Reserved	0C2	Reserved by ITU-T	~~Clause 8.6.2.3.5~~
	FL_TerminateFlow.req	0C3	Request flow termination	Clause 8.6.2.3.13
	FL_TerminateFlow.cnf	0C4	Confirm flow termination	Clause 8.6.2.3.14
	FL_BrokenTunnel.ind	0C5	Indicate broken tunnel	Clause 8.6.2.3.19

Table 8-88 – OPCODEs of management messages

Category	Message name	OPCODE (hex)	Description	MMPL Reference
	FL_BrokenTunnel.rsp	0C6	Response to indication	Clause 8.6.2.3.20
	FL_ReleaseTunnel.req	0C7	Request Release Tunnel	Clause 8.6.2.3.21
	FL_ReleaseTunnel.cnf	0C8	Confirm Release Tunnel	Clause 8.6.2.3.22
	FL_DM_RenewTunnel.req	0C9	DM renew tunnel request	Clause 8.6.2.3.23
	FL_DM_RenewTunnel.cnf	0CA	Confirm DM renew tunnel	Clause 8.6.2.3.24
	FL_RenewTunnel.req	0CB	Renew tunnel request	Clause 8.6.2.3.25
	FL_RenewTunnel.cnf	0CC	Confirm Renew tunnel	Clause 8.6.2.3.26
	FL_DeleteFlow.req	0CD	Delete Flow request	Clause 8.6.2.3.27
	FL_DeleteFlow.cnf	0CE	Confirm Delete Flow	Clause 8.6.2.3.28
Media Access Plan (0DX)	MAP	0D0	MAP message	Clause 8.8
Channel Estimation 2 (0EX)	CE_Request.ind	0E0	Channel estimation trigger	Clause 8.11.7.11
	CE_Initiation.req	0E1	Channel estimation initiation request	Clause 8.11.7.12
	CE_Initiation.cnf	0E2	Channel estimation initiation confirmation	Clause 8.11.7.13
	CE_ProbeRequest.ind	0E3	Request for PROBE frame transmission	Clause 8.11.7.14
	CE_Cancellation.req	0E4	Channel estimation cancellation request	Clause 8.11.7.15
	CE_BatIdMaintain.ind	0E5	BAT ID maintenance	Clause 8.11.7.16
	CE_Cancellation.cnf	0E6	Channel estimation cancellation confirmation	Clause 8.11.7.17
	Reserved	0E7 – 0EF	Reserved by ITU-T	
Transmission Profile (0FX)	Reserved for amendments	0F0-0FF	Reserved by ITU-T	
Neighbouring network coordination (10X to 13~~1~~X)	Reserved for amendments	100-13F	Reserved by ITU-T	
AKM 2 (14X)	AUT_NodeAuthenticated.req	140	Indication from the SC to DM that the node has been authentication	Clause 9.2.5.1.5
	AUT_NodeAuthenticated.cnf	141	Confirmation of the AUT_NodeAuthenticated.req message	Clause 9.2.5.1.6
	Reserved	142 – 14F	Reserved by ITU-T	
Reserved	Reserved	~~0~~100~~A0-F~~7FF	Reserved by ITU-T	
MIMO (8XX – 9XX)	Reserved for use by ITU-T G.9963 [x]	800 – 9FF	Reserved by ITU-T	
Reserved	Reserved	A00 – FFF	Reserved by ITU-T	

8.10.1.2 Management of message sequence numbers and segmentation

The sequence number space shall be unique for each {OPCODE, OriginatingNode} tuple. The sequence number shall be incremented for each transmitted message except as follows:

- When the same message is retransmitted (e.g., when a message has been lost), the message sequence number shall be the same as the original transmitted message and the repetition number shall be incremented by 1.
- When a message is relayed, the sequence number and the repetition number fields shall not be modified.

NOTE 1 – The sequence number space used by an originating node for a given OPCODE is the same regardless of the destination (e.g., single counter per OPCODE).

When the field Force Sequence Bit (FSB) of the MMH is set to one, it indicates that the receiver shall process this message without performing any sequence filtering. The receiver shall also consider the sequence number of this message as the latest valid sequence number associated with the transmitter's DeviceID and OPCODE of the message.

NOTE 2 – The increment in the value of the message sequence number is independent of the value of the FSB field.

~~However, t~~The following segmentation rules apply to any segmented LCDU:

- ~~t~~The segmentation shall be done in the ascending order of octets.
- ~~a~~All the segments shall have the same sequence number.
- ~~t~~The segmentation shall not be changed if the LCDU is retransmitted, unless a new sequence number is generated.
- The segmentation shall not be changed if the LCDU is relayed (the sequence number shall be maintained the same).

Segmentation shall only be done for LCDUs with payload greater than 1500 bytes.

Some management protocols may require knowing if the sequence number of a received LCDU is older, equal or newer than the last correctly received LCDU. The sequence number is a 16-bit unsigned integer used for that purpose and ~~it shall be incremented by one for each new message. It~~ shall be in the range 0 to (SequenceModulus -1), where SequenceModulus is equal to 2^{16}. When it is equal to 2^{16}, it wraps-around to zero. If the FSB field of the MMH is set to one, the received LCDU shall be considered as the newest. If the FSB field of the MMH is set to zero, ~~S~~sequence numbers of LCDUs with the same OPCODE shall be compared according to the following rules:

- The first LCDU received from a node shall be considered as a new message containing new information. The node shall perform the operations required by the protocol that defines that OPCODE.
- If the sequence number of the new received LCDU is the same as the sequence number of the LCDU already kept by the node, the new received LCDU shall be considered to be equal to the LCDU kept by the node.
- If the sequence number of the new received LCDU is higher than the sequence number of the LCDU already kept by the node and the difference between the numbers is, in absolute value, less than half of SequenceModulus, the new received LCDU shall be considered to be newer. Otherwise it shall be considered to be older.
- If the sequence number of the new received LCDU is lower than the sequence number of the LCDU already kept by the node and the difference between the numbers is, in absolute value, lower than half of SequenceModulus, the new received LCDU shall be considered to be older. Otherwise it shall be considered to be newer.

In any of the above cases, the actions to perform by the node that receives the LCDU depend on the protocol that defines that OPCODE.

NOTE 3 – A transmitter may use the FSB bit to force synchronization with the receiver. Once the transmitter gets confirmation that the receiver is synchronized, it should set FSB to zero.

…

38) Clause 8.10.2, Control message format

Revise the text of clause 8.10.2 "Control message format" as follows:

8.10.2 Control message format

This clause describes the format of short control messages, intended for communication between nodes of the same domain. All control messages carried over CTMG frames (clause 7.1.2.3.2.6 of [ITU-T G.9960]) shall be formatted as shown in Figure 8-55, including a control message header (CMH) and a control message parameter list (CMPL). A control message is carried in the PHY-frame header of CTMG frame, hence protected by the HCS and E_HCS (clauses 7.1.2.3.1.9 and 7.1.2.3.3.2 of [ITU-T G.9960]). The control messages carried over CTMG frames are not subject to relay. The first byte (octet 0) of the CMH shall be the first byte passed to the PHY layer. A CTMG frame transmitted in CBTS shall be considered as having an MPDU priority equal to 7.

G.9961(10)_F8-55

Figure 8-55 – Format of a control message

The CMH defines the length and other parameters of the message. The type of the message is identified by an OPCODE associated with a particular control function as presented in Table 8-90. The CMPL includes a list of control message parameters depending on the control function. The format of control message shall be as shown in Table 8-89.

Table 8-89 – Format of control messages

	Content	Octet	Bits	Description
CMH	Length	0 and 1	[5:0]	Length of the CMPL in octets (*V*), encoded as a 6-bit unsigned integer. The valid range of *V* is 1 to 31
	OPCODE		[15:6]	10-bit OPCODE, indicates control message type (Note 1)
	Reserved	2	[7:0]	Reserved by ITU-T (Note 2)
CMPL	Message parameters	3 to (*V*+2)	[(8*V*–1):0]	Depends on the OPCODE
NOTE 1 – The OPCODEs are defined in Table 8-90. NOTE 2 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.				

The format of CMPLs may be revised in future versions of this Recommendation by appending additional fields. Furthermore, fields may be defined using bits that are currently indicated as reserved for ITU-T. Nodes indicate the version of the Recommendation that they support during registration (see Table 8-16) and topology updates (see Table 8-47). Nodes shall be able to parse the CMPL (the length of the CMPL is specified in the CMH) but shall ignore the content of fields that they do not understand, i.e., those associated with later versions of the Recommendation.

39) Clause 8.11, Channel estimation protocol (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.11 "Channel estimation protocol" (from ITU-T G.9961 Corrigendum 1) as follows:

8.11 Channel estimation protocol

The channel estimation protocol describes the procedure of measuring the characteristics of the channel between the transmitter (source) and the receiver (destination) nodes. The procedure involves initiation of channel estimation, transmissions of PROBE frames, and selection of parameters.

Channel estimation can be done in two phases:

- Channel discovery – Initial channel estimation.
- Channel adaptation – Subsequent channel estimation to adapt changing channel.

The protocols used for channel discovery and channel adaptation can be started either by the transmitter or by the receiver. The core part of the channel estimation protocol is identical in these two cases, and is always initiated by the receiver (receiver-initiated channel estimation). The transmitter can request the receiver to initiate channel estimation (transmitter-requested channel estimation).

During the initiation process, the transmitter and receiver jointly determine input parameters for channel estimation such as channel estimation window (a fraction of a MAC cycle over which channel estimation should be executed), the minimum value of G (G_{min}, see clause 7.1.4.2.4 of [ITU-T G.9960]), and parameters for the PROBE frame. The receiver selects the BAT_ID associated with the BAT to be updated. This BAT_ID is used for an identifier for a particular channel estimation process throughout the rest of the process. The receiver shall consider its own bandplan information (namely the StartSubCarrier and StopSubCarrier) and that of the transmitter when calculating the BAT. More specifically, the range of sub-carriers of the BAT sent in the CE_ParamUpdate.req message shall be within the intersection of the sub-carrier ranges determined by the StartSubCarrier and StopSubCarrier of both the receiver and transmitter.

Once the channel estimation process is initiated, the receiver may request the transmitter to send one or more PROBE frames. The receiver can change parameters of a PROBE frame at each request. If the receiver requests a PROBE frame without specifying its parameters (e.g., ~~probe~~ request for PROBE frame transmission request via ACK_CE_CTRL as described in clause 8.11.1.4), the transmitter transmits the PROBE frame using parameters previously selected by the receiver. The receiver is not required to request PROBE frames if it chooses other means such as MSG frames or PROBE frames transmitted to other nodes to estimate the channel. ~~The protocol provides numerous options to expedite the channel estimation process for faster channel adaptation.~~

The receiver terminates the channel estimation process by sending the outcome of channel estimation to the transmitter. This includes, but is not limited to, the following parameters:

- Bit allocation table (BAT);
- FEC coding rate and block size;
- ~~g~~Guard interval for payload;
- PSD ceiling.

The receiver may cancel the channel estimation process without generating new channel estimation parameters.

The protocol provides several options to expedite the channel estimation process for faster channel adaptation. For example, the channel estimation initiation process (clause 8.11.1.1) can be omitted in case of channel adaptation where no new input parameter negotiation is necessary. The receiver

can create a new BAT by sending an unsolicited CE_ParamUpdate.req (clause 8.11.3.1) or update the existing BAT by sending a CE_PartialBatUpdate.req (clause 8.11.3.2). The receiver can request PROBE frame transmission without going through channel estimation initiation process (clause 8.11.4).

40) Clause 8.11.1.1, Channel estimation initiation (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.11.1.1 "Channel estimation initiation" (from ITU-T G.9961 Corrigendum 1) as follows:

8.11.1.1 Channel estimation initiation

The receiver initiates the channel estimation process by sending the transmitter a ~~CM~~ CE_Initiation.req message. ~~This message shall be carried using a CTMG frame.~~

The receiver shall select CE_GRP ~~MIN~~ (G_{min}), which indicates the ~~minimum~~ value of GRP_ID (G) associated with the BAT to be updated. The receiver shall select CE_STIME and CE_ETIME, which determines the start and end time of the channel estimation window. During the rest of channel estimation process, the transmitter shall send PROBE frames inside this window. The receiver shall select CE_BAT_ID from ones that are currently invalid. This value shall be used to differentiate multiple channel estimation processes being executed at the same time. The receiver may request PROBE frame transmission by setting CE_PRB_RQST field. The CE_PRB_PARM field specifies parameters for the default PROBE frame. If the CE_PRB_RQST field is not set to one, parameters for the default PROBE frame shall be as follows: CE_PR_PRBTYPE = 0001_2; CE_PR_PRBFN = 0000_2; CE_PR_PRBSYM = 0011_2; CE_PR_PRBGI = 111_2 and CE_PR_APSDC = 31.

The receiver may resend the ~~CM~~ CE_Initiation.req message, if it does not receive the ~~CM~~ CE_Initiation.cnf message within 200 ms~~ee~~.

41) Clause 8.11.1.3, Channel estimation initiation confirmation (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.11.1.3 "Channel estimation initiation confirmation" (from ITU-T G.9961 Corrigendum 1) as follows:

8.11.1.3 Channel estimation initiation confirmation

The transmitter shall confirm~~s~~ the channel estimation initiation request ~~either process~~ by sending the receiver a ~~CM~~ CE_Initiation.cnf message.

The transmitter shall indicate whether it grants or rejects the channel estimation initiation request ~~by setting CE_CNF_TYPE and CE_CNF_CODE~~. The transmitter shall set CE_BAT_ID to the value selected by the receiver ~~via channel estimation initiation~~in the CE_Initiation.req message~~. The transmitter~~, and shall ~~finalize~~ set CE_GRP ~~MIN, which shall be larger than or~~ to be equal to the ~~one~~ value indicated by the receiver. ~~The transmitter may use any value of G (sub-carrier grouping, see clause 7.1.4.2.4 of [ITU-T G.9960]) that satisfies the following conditions: $G(t_i) \geq G_{min}$, and $G(t_{i+1}) \geq G(t_i)$, where $G(t_i)$ denotes the value of G at arbitrary time t_i, and $t_i < t_{i+1}$. If the transmitter uses $G > G_{min}$, the new BAT (B') shall be formed by decimating the old BAT (B) by taking the minimum BAT entry from the original group of sub-carriers. That is, the new bit allocation entry for sub-carrier i, $B'_i = \min\{B_i\}$ where $i = G \times j, G \times j+1, \ldots, G \times j+G-1$, and $j = 0, 1, \ldots, (N/G)-1$.~~

~~If the receiver has requested one or more PROBE frames in CM_CE_Initiation.req message, then the transmitter shall send a CM_CE_Initiation.cnf message over the first PROBE frame (i.e., CMPL of CM_CE_Initiation.cnf message is carried in PRB_CE_CNF field of PROBE frame as described in clause 7.1.2.3.2.7.6 of [ITU-T G.9960]). This PROBE frame shall contain the PROBE symbols as requested in CM_CE_Initiation.req message. If the receiver has not requested PROBE frames,~~

~~t~~The transmitter shall send the ~~CM_CE_Initiation.cnf message using a CTMG frame. The transmitter shall send CM_~~CE_Initiation.cnf message within 100 ms~~ee~~ after it receives the ~~CM_~~CE_Initiation.req message. If the transmitter needs to request the bandwidth for PROBE frame transmission, the transmitter shall send the CE_Initiation.cnf message within 200 ms.

42) Clause 8.11.2.1, Channel estimation request (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.11.2.1 "Channel estimation request" (from ITU-T G.9961 Corrigendum 1) as follows:

8.11.2.1 Channel estimation request

The transmitter triggers the channel estimation process by sending the receiver ~~CM_~~CE_Request.ind message. ~~This message shall be carried using a CTMG frame.~~

The transmitter ~~may~~can specify the channel estimation window (CE_STIME ~~&~~and CE_ETIME). In this case the receiver shall use the same channel estimation window as the transmitter requested in CE_Initiation.req message. Otherwise, the receiver can determine the channel estimation window at its own discretion.

The receiver shall respond to a CE_Request.ind message from the transmitter within 100 ms with either a CE_Initiation.req message or a CE_ParamUpdate.req message.

If the transmitter does not receive either the CE_Initiation.req or the CE_ParamUpdate.req messages within 200 ms after the CE_Request.ind message is sent, it may resend the channel estimation request message.

43) Clause 8.11.3.2, Partial BAT update (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.11.3.2 "Partial BAT update" (from ITU-T G.9961 Corrigendum 1) as follows:

8.11.3.24 ~~Channel adaptation via p~~Partial BAT update

The transmitter and receiver that communicate with each other by establishing a common runtime BAT may update a portion of the BAT at any time during its usage. The receiver may initiate the partial BAT update (PBU) by sending PBU information in the management message.

The process of partial BAT update is described as follows:

1) At any time during communication, the receiver may send the PBU request for ~~the~~any valid BAT ~~currently~~ used by the transmitter. The PBU request contains the new valid BAT_ID (N_BAT_ID), old BAT_ID (O_BAT_ID) associated with the BAT to be updated, and bit allocation changes (see clause 8.11.3.2.14~~.1.1~~).
2) Upon reception of the PBU request, the transmitter shall update the BAT associated with the O_BAT_ID, and assign N_BAT_ID to the updated BAT and reply with the PBU confirmation. After receiving the first ~~MSG~~ frame carrying payload using the N_BAT_ID, the receiver shall consider O_BAT_ID ~~i~~as invalid (see clause 8.11.5).

…

44) Clause 8.11.4, Channel estimation using PROBE frames (from ITUT G.9961 Corrigendum 1)

Revise the text of clause 8.11.4 "Channel estimation using PROBE frames" (from ITU-T G.9961 Corrigendum 1) as follows:

8.11.4 Channel estimation using PROBE frames

The receiver can request the transmitter for PROBE frame transmission at any time after registration without going through the channel estimation initiation process.

To request PROBE frames, the receiver may use CE_ProbeRequest.ind messages or the ACK_CE_CTRL field in the PFH of an ACK frame (see clause 7.1.2.3.2.3.8 of [ITU-T G.9960]). Upon reception of a request for PROBE frame transmission, the transmitter should transmit PROBE frames as soon as possible.

If the receiver requests a PROBE frame through a specific management message, the transmitter shall transmit the PROBE frame using parameters selected by the receiver, that is, the parameters selected in the latest request for PROBE frame transmission (CE_ProbeRequest.ind) or channel estimation initiation (CE_Initiation.req).

If the receiver requests a PROBE frame through an ACK frame, the transmitter shall use the default PROBE frame. The transmitter shall use the default PROBE frame for all ACK frame-based requests for PROBE frame transmission by the receiver. In this case, the transmitter may use an entire MAC cycle to transmit PROBE frames, regardless of a particular channel estimation window associated with the BAT_ID under channel estimation.

The parameters for the default PROBE frame are determined by the receiver through the CE_Initiation.req message as described in clause 8.11.1.1. Alternatively, they can be updated by setting a bit in the CE_ProbeRequest.ind message as described in Table 8-102.

When a transmitter receives a request for PROBE frame transmission from a receiver while handling previous requests for PROBE frame transmission from the same receiver, it should ignore the new request if the requested parameters are the same as the old ones, regardless of the value of the BAT_ID under estimation.

NOTE – The transmitter should try to cover as much of the channel estimation window as possible when generating PROBE frames.

When the receiver requests a PROBE frame via ACK frames, it may request multiple times by sending multiple ACK frames by setting ACK_CE_CTRL until it receives the PROBE frame. The transmitter should ignore new requests for PROBE frame transmission coming from the receiver in order to avoid unnecessary PROBE transmissions.

After PROBE transmissions, the receiver may send the outcome of channel estimation to the transmitter in case it is needed, using an unsolicited CE_ParamUpdate.req (clause 8.11.3.1) or a partial BAT update (clause 8.11.3.2).

A PROBE frame should be considered as having an MPDU priority equal to 7.

45) Clause 8.11.7.3, Format of CE_ParamUpdate.req

Revise the text of clause 8.11.7.3 "Format of CE_ParamUpdate.req" as follows:

8.11.7.3 Format of CE_ParamUpdate.req~~ind~~

The format of the MMPL of the CE_ParamUpdate.req~~ind~~ message shall be as shown in Table 8-93.

Table 8-93 – Format of the MMPL of the CE_ParamUpdate.req~~ind~~ message

Field	Octet	Bits	Description
New BAT ID	0	[4:0]	This field indicates the BAT_ID associated with a new BAT (CE_BAT_ID). It shall be formatted as shown in Table 7-55 of [ITU-T G.9960]

Table 8-93 – Format of the MMPL of the CE_ParamUpdate.req~~ind~~ message

Field	Octet	Bits	Description
Bandplan ID		[7:5]	This field indicates the type of bandplan based on which the subsequent BAT entry is defined. It shall be formatted as shown in Table 7-10 of [ITU-T G.9960]
~~Minimum g~~Group ID	1	[2:0]	This field indicates the ~~minimum~~ GRP_ID (CE_GRP~~_MIN~~) associated with the new BAT (*G*)~~, and determined during at the channel estimation~~ initiation ~~confirmation~~. It shall be formatted as shown in Table 7-13 of [ITU-T G.9960]
Reserved		[7:3]	Reserved by ITU-T (Note 1)
VALID_BAT_ID	2 to 4	[23:0]	This field contains a bitmap indicating which runtime BATs are valid (including the New BAT ID) for this node (SID) when receiving from the destination node (DID). Each bit is associated with one runtime BAT. The LSB of the VALID_BAT_ID shall be set to one if runtime BAT 8 is valid. The MSB of the VALID_BAT_ID shall be set to one if runtime BAT 31 is valid
NUM_TX_AVAIL_BATS	5	[4:0]	This field contains the number of runtime BATs, assuming *G*=1, that this node (SID) can support when transmitting to the destination node (DID). Valid values are from 0 to 24
Reserved		[7:5]	Reserved by ITU-T (Note 1)
New block size	6~~2~~	[1:0]	This field indicates the proposed BLKSZ associated with the new BAT. It shall be formatted as shown in Table 7-7 of [ITU-T G.9960] (Note 2)
New FEC rate		[4:2]	This field indicates the proposed FEC_RATE associated with the new BAT. It shall be formatted as shown in Table 7-12 of [ITU-T G.9960] (Note 3)
New GI		[7:5]	This field indicates the proposed GI_ID associated with the new BAT. It shall be formatted as shown in Table 7-14 of [ITU-T G.9960] (Note 4)
New PSD ceiling	7~~3~~	[4:0]	This field is the value of APSDC-M in the PHY-frame header associated with the new BAT. This field shall be formatted as shown in clause 7.1.2.3.2.2.11 of [ITU-T G.9960]
NUM_VALID_DUR		[7:5]	This field indicates the number of valid durations specified for the new BAT (*V*). The valid range of values for this field is from 0 (*V*=1) to 7 (*V*=8) (Note 5)
CE_STIME_1	8~~4~~	[7:0]	This field indicates the start time of the first duration in which the new BAT is valid. It shall be formatted as shown in Table 8-98
CE_ETIME_1	9~~5~~	[7:0]	This field indicates the end time of the first duration in which the new BAT is valid. It shall be formatted as shown in Table 8-99
…	…	…	…
CE_STIME_v	2V+6~~2~~	[7:0]	This field indicates the start time of the last duration in which the new BAT is valid. It shall be formatted as shown in Table 8-98

Table 8-93 – Format of the MMPL of the CE_ParamUpdate.req~~ind~~ message

Field	Octet	Bits	Description
CE_ETIMEv	2V+7~~3~~	[7:0]	This field indicates the end time of the last duration in which the new BAT is valid. It shall be formatted as shown in Table 8-99
$TIDX_{MIN}$	(2V+8~~4~~) to (2V+10~~6~~)	[11:0]	12-bit unsigned integer indicating the lowest sub-carrier index to which non-zero bits are assigned. It shall be an integer multiple of *G* (Note 6)
$TIDX_{MAX}$		[23:12]	12-bit unsigned integer indicating the highest sub-carrier index to which non-zero bits are assigned. It shall be an integer multiple of *G* (Note 6) and if bit-loading grouping is used (*G*>1) shall meet: $TIDX_{MAX}$+G-1 ≤ StopSubCarrier, where StopSubCarrier is specified in Table 8-16.6 (Bandplan Info Capability Value field). Let W denote the number of BAT entries, which is ($TIDX_{MAX}$ – $TIDX_{MIN}$) / *G* + 1. Let Z denote the smallest integer larger than or equal to W/2
B_1	2V+11~~7~~	[3:0]	4-bit unsigned integer indicating the number of bits assigned to sub-carrier indices $TIDX_{MIN}$ to $TIDX_{MIN}$ + *G* – 1 (Note 6)
		[7:4]	4-bit unsigned integer indicating the number of bits assigned to sub-carrier indices $TIDX_{MIN}$ + *G* to $TIDX_{MIN}$ + 2*G* – 1 (Notes 6, 7, 8)
…	…	…	…
B_Z	2V+10~~6~~ + Z	[3:0]	4-bit unsigned integer indicating the number of bits assigned to sub-carrier indices $TIDX_{MAX}$ – *G* to $TIDX_{MAX}$ – 1 if W is even, or to sub-carrier indices $TIDX_{MAX}$ to $TIDX_{MAX}$ + *G* – 1 if W is odd (Notes 6, 7)
		[7:4]	4-bit unsigned integer indicating the number of bits assigned to sub-carrier indices $TIDX_{MAX}$ to $TIDX_{MAX}$ + *G* – 1 if W is even (Notes 6, 9)

NOTE 1 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.

NOTE 2 – The transmitter shall use the proposed block size or larger block size for a new connection. Once the block size is selected for a connection, it shall not be changed throughout the lifetime of the connection (clause 8.1.3.2).

NOTE 3 – The transmitter shall use the proposed FEC rate or lower FEC rate.

NOTE 4 – The transmitter shall use the proposed GI or longer GI value.

NOTE 5 – A new BAT shall only be used over specified non-overlapping durations (up to 8) within a MAC cycle, defined by CE_STIME_i and CE_ETIME_i.

NOTE 6 – Sub-carrier index represents the physical index (clause 7.1.4.1 of [ITU-T G.9960]). All BAT entries outside [$TIDX_{MIN}$, $TIDX_{MAX}$ + *G* – 1] shall be considered as unloaded.

NOTE 7 – If a sub-carrier is not loaded, the field shall be set to zero.

NOTE 8 – If W = 1, this field shall be set to zero.

NOTE 9 – If W is an odd number, this field shall be set to zero.

46) Clause 8.11.7.12, Format of CE_Initiation.req, and clause 8.11.7.13, Format of CE_Initiation.cnf (both taken from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.11.7.12 "Format of CE_Initiation.req" and clause 8.11.7.13 "Format of CE_Initiation.cnf" (both taken from ITU-T G.9961 Corrigendum 1) as follows:

8.11.7.128.2 Format of ~~CM_~~CE_Initiation.req

The format of the ~~C~~MMPL of the ~~CM_~~CE_Initiation.req message shall be as shown in Table 8-100.

Table 8-100 – Format of the ~~C~~MMPL of the ~~CM_~~CE_Initiation.req message

Field	Octet	Bits	Description
CE_BAT_ID	0	[4:0]	This field indicates the BAT_ID associated with the runtime BAT to be created by channel estimation. It shall be formatted as shown in Table 7-55 of [ITU-T G.9960]
CE_GRP~~_MIN~~		[7:5]	This field indicates the ~~minimum~~ value of sub-carrier grouping. It shall be formatted as shown in Table 7-13 of [ITU-T G.9960]
CE_STIME	1	[7:0]	This field indicates the time at which the transmitter can start PROBE frame transmissions, and it shall be coded as shown in Table 8-98
CE_ETIME	2	[7:0]	This field indicates the time at which the transmitter shall end PROBE frame transmissions, and it shall be coded as shown in Table 8-99
CE_PRB_RQST	3	[0]	This field shall be set to one if the receiver wants PROBE frames along with channel estimation initiation confirmation. It shall be set to zero otherwise
Reserved		[7:1]	Reserved by ITU-T (Note)
CE_PRB_PARM	4 to 6	[23:0]	This field specifies a set of parameters for PROBE frame. It shall be coded as shown in Table 8-102. This field shall be set to 000000_{16} if CE_PRB_RQST is set to zero
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

8.11.7.138.3 Format of ~~CM_~~CE_Initiation.cnf

The format of the ~~C~~MMPL of the ~~CM_~~CE_Initiation.cnf message ~~shall be as~~is shown in Table 8-101.

Table 8-101 – Format of the ~~C~~MMPL of the ~~CM_~~CE_Initiation.cnf message

Field	Octet	Bits	Description
CE_BAT_ID	0	[4:0]	This field indicates the BAT_ID associated with the runtime BAT to be created by channel estimation. It shall be formatted as shown in Table 7-55 of [ITU-T G.9960]
CE_GRP~~_MIN~~		[7:5]	This field indicates the ~~minimum~~ value of sub-carrier grouping. It shall be formatted as shown in Table 7-13 of [ITU-T G.9960]
~~CE_CNF_TYPE~~	~~1~~	~~[0]~~	~~This field indicates the type of channel estimation confirmation. It shall be set to one if channel estimation initiation is granted or set to zero otherwise.~~
~~CE_CNF_CODE~~		~~[3:1]~~	~~This field indicates the reason for channel estimation rejection. 001_2: CE_BAT_ID is invalid.~~

Table 8-101 – Format of the ~~C~~MMPL of the ~~CM_~~CE_Initiation.cnf message

Field	Octet	Bits	Description
			~~010_2: Bandwidth for PROBE frame transmission is not available.~~ ~~000_2, 011_2 to 111_2: Reserved by ITU-T.~~ ~~If CE_CNF_TYPE is set to one, this field shall be set to 000_2.~~
~~Reserved~~		~~[7:4]~~	~~Reserved by ITU-T (Note).~~
NUM_AVAIL_BATS		[4:0]	This field contains the number of available runtime BATs, assuming G = 1, that this node (SID) can support when transmitting to the destination node (DID). It excludes the BAT associated with the CE_BAT_ID. Valid values are from 0 to 23
Request Status		[7:5]	0 – Channel estimation initiation is confirmed 1 – Rejected (CE_BAT_ID is valid and currently in use)~~.~~ 2 – Rejected (Bandwidth for PROBE frame transmission is not available) 3 – Rejected (Bandwidth request for probe frame transmission is pending) 4 – Rejected (Channel estimation window is currently not available) 5 to 7 – Reserved by ITU-T (Note)
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

47) Clause 8.12.7, Reset of a unicast connection with acknowledgements

Revise the text of clause 8.12.7 "Reset of a unicast connection with acknowledgements" as follows:

8.12.7 Reset of a unicast connection with acknowledgements

Reset of a connection shall only be initiated by the transmitter. ~~For the connections that require reset of their transmission and reception windows~~If a reset is received (see clause 8.9.5.3~~.1~~), the nodes shall follow the procedure described in clause 8.12.1.1 with CNN_MNGMT set to 0011_2 for a management connection, and the procedure described in clause 8.12.1.2 with CNN_MNGMT set to 0111_2 for a data connection.

48) New clause 8.12.9, Multicast data connection

Add text for new clause 8.12.9 "Multicast data connection" (clause 8.16 is found in ITU-T G.9961 Corrigendum 1) as follows:

8.12.9 Multicast data connection

For multicast data connections, see clause 8.16 – PHY multicast binding protocol.

49) Clause 8.16.1, Initialization of a PHY multicast group (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.16.1 "Initialization of a PHY multicast group" (from ITU G.9961 Corrigendum 1) as follows:

8.16.1 Initialization of a PHY ~~M~~multicast ~~G~~group ~~for a new Multicast Stream~~

A transmitting node of a PHY multicast stream may initiate the PHY multicast binding protocol when it needs to transmit the same data to several nodes directly (in the same hop).~~upon detecting~~

~~the presence of a multicast source (e.g. when IGMP Query or multicast traffic is transmitted by the multicast source) when there are nodes that requested to receive the multicast stream (e.g. via IGMP join message).~~

~~If the transmitter~~When a node initiates the multicast binding protocol, it shall compute the common BATs to be used for the PHY multicast ~~stream~~group based on the BATs (see clause 8.11) reported by the receiver nodes in the group~~that requested to receive this multicast stream. The transmitter shall then determine the number of multicast groups and the assignment of receivers to each multicast group.~~ The transmitter shall also consider its own StartSubCarrier and StopSubCarrier and that of the receivers of the PHY multicast group. More specifically, the range of sub-carriers of the BAT sent in the MC_GrpInfoUpdate.ind message shall be within the intersection of the sub-carrier ranges determined by the node's StartSubCarrier and StopSubCarrier of both the transmitter and all receivers in the PHY multicast group (also see clause 8.18.5).

BATs to be used for multicast transmission shall not include values of 5, 7, 9 or 11 bits.

When Mc-ACK is used for a PHY multicast group, the transmitter shall assign~~s~~ receivers to the Mc-ACK/NACK slots. The acknowledgement protocol state machine for PHY multicast transmission shall be initialized as specified in clause 8.9.5.4.

NOTE 1 – The actual method for deciding on the number of PHY multicast groups and the BATs used for each group and the assignment of nodes to the Mc-ACK slots is beyond the scope of this Recommendation.

The transmitter shall then send MC_GrpInfoUpdate.ind message to all the nodes that will be part of the PHY multicast group including information about the BATs to be used within the PHY multicast group and the receiver nodes that are members of the group~~for each created multicast group~~. Upon reception of a MC_GrpInfoUpdate.ind message, each receiver that appears as a receiver of a PHY multicast group shall confirm the message by sending an MC_GrpInfoUpdate.cnf message to the transmitter.

In case the MC_GrpInfoUpdate.cnf message is not received from all of the receiving devices within T_{MCST} the transmitter shall retransmit the request until N_{MCST} retries are exhausted.

The transmitter may control whether flow-control is enabled or not for a PHY multicast group by setting the appropriate value of the FlowControlInd field in the MC_GrpInfoUpdate.ind message. The decision as to whether flow control should be enabled or not is beyond the scope of this Recommendation.

NOTE 2 – Flow-control may be disabled if Mc-ACK slots have not been allocated to all members of the PHY multicast group.

When flow-control is not used on a PHY multicast group a transmitter shall advertise the recommended receive buffer size in the MC_GrpInfoUpdate.ind message. The initial recommended receive buffer size (ACK_RX_CONF_WINDOW_SIZE) for a PHY multicast group shall be specified by the transmitter to have a maximum value (set in the MinRxBufSize field in Table 8-107). Upon reception of the MC_GrpInfoUpdate.ind message receivers shall respond by specifying their available receive buffer sizes (ACK_RX_CONF_WINDOW_SIZE) in the MC_GrpInfoUpdate.cnf message. The transmitter shall collect all the receive buffer sizes advertised by all PHY multicast group members and shall adjust the recommended receive buffer size advertised in the MC_GrpInfoUpdate.ind message. The adjusted value of the MinRxBufSize field (see MinRxBufSize in Table 8-107) in the MC_GrpInfoUpdate.ind message shall be set to the minimum of the receive buffer size of all members of the PHY multicast group. Upon reception of the adjusted MC_GrpInfoUpdate.ind message receivers may reduce the size of their receive buffers to the specified value. The new receive buffer size used by the receiver shall be reported in the corresponding MC_GrpInfoUpdate.cnf message.

NOTE 3 – Based on the advertised receive buffer sizes of members of the PHY multicast group the transmitter may decide to reassign PHY multicast group members to different groups.

When flow-control is not used the value of FLCTRL specified in the ACK~~, BACK and BMSG~~ frames shall be set to the value advertised by the receiver in the last MC_GrpInfoUpdate.cnf message.

When flow control is used the recommended receive buffer size specified in the MC_GrpInfoUpdate.ind message shall be ignored by the receiving nodes that are assigned an Mc-ACK slot. The initial recommended receive buffer size (ACK_RX_CONF_WINDOW_SIZE) for a PHY multicast group shall be specified by the transmitter to have a maximum value (set in the MinRxBufSize field in Table 8-107). The receiving nodes that are not assigned a Mc-ACK slot shall respond by specifying their available buffer sizes (ACK_RX_CONF_WINDOW_SIZE) in the MC_GrpInfoUpdate.cnf message. The transmitter shall collect the receive buffer sizes advertised by all receiving nodes that are not assigned an Mc-ACK slot and shall adjust the recommended receive buffer size advertised in the MC_GrpInfoUpdate.ind message. The adjusted value of the MinRxBufSize field (see MinRxBufSize in Table 8-107) in the MC_GrpInfoUpdate.ind message shall be set to the minimum of the receive buffer size of those receiving nodes of the PHY multicast group. Upon reception of the adjusted MC_GrpInfoUpdate.ind message these receivers may reduce the size of their receive buffers to the specified value. The new receive buffer size used by these receivers shall be reported in the corresponding MC_GrpInfoUpdate.cnf message. The transmitter shall limit the number of LPDUs transmitted within each PHY frame according to the transmit window corresponding to this group, to the minimum of the receive buffer size indicated in the MC_GrpInfoUpdate.cnf ~~-~~message by the receivers that are not assigned an Mc-ACK slot and the values indicated in the FLCTRL field by the receivers that are assigned Mc-ACK slots~~receiver~~.

Before the multicast binding is completed for a new PHY multicast ~~stream~~group the transmitter may send the multicast stream traffic using the BROADCAST_ID as DID, or by making unicast transmissions to the multicast receivers.

During initialization of a PHY multicast group or when a change in the membership of nodes of an existing PHY multicast group occurs the transmitter may use broadcast DID when sending the protocol messages. The reserved MAC address 01-19-A7-52-76-96 shall be used as the DA in the LCDU delivering the MC_GrpInfoUpdate.ind message. The DestinationNode of the LLC frame corresponding to the LCDU delivering the MC_GrpInfoUpdate.ind message shall be set to zero.

50) Clause 8.16.2, Maintenance of multicast binding information (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.16.2 "Maintenance of multicast binding information" (from ITU-T G.9961 Corrigendum 1) as follows:

8.16.2 Maintenance of multicast binding information

The transmitter shall send MC_GrpInfoUpdate.ind message as specified in this clause to update receivers of a PHY multicast group when there is a change in BATs, or in the membership of receiver nodes, or in Mc-ACK slot assignment ~~occurs~~.

Changes in the Mc-ACK slots assignments shall take effect only when the number of Mc-ACK slots following a multicast transmission changes, as reflected in the NUM_MCACK_SLOTS field of the PHY-frame header. The transmitter shall not indicate a different number of Mc-ACK slots in the NUM_MCACK_SLOTS field until all receivers assigned to acknowledge have confirmed their status in MC_GrpInfoUpdate.ind message by sending an MC_GrpInfoUpdate.cnf message. The transmitter shall not change the Mc-ACK slot assignment for an existing node if the number of Mc-ACK slots remains same.

A receiver that was assigned a Mc-ACK slot of a PHY multicast group associated with a multicast stream shall continue acknowledging in its assigned slot until its assignment for this Mc-ACK slot is terminated by an MC_GrpInfoUpdate.ind message. ~~If the receiver is no longer interested in that~~

~~multicast stream while it has an assigned Mc-ACK slot, it shall set the FACK field to 111 (see clause 7.1.2.3.2.3.9.1.5) and ACKI field to all ones (see clause 7.1.2.3.2.3.9.1.7) to indicate to the transmitter that it is no longer interested in receiving the multicast stream, and its ACKI field shall be ignored by the transmitter.~~

The transmitter of a PHY multicast group may remove any receiver that is a member of the PHY multicast group at any time by sending a MC_GrpRemove.req message to that receiver. The receiver shall send the MC_GrpRemove.cnf message to the transmitter, confirming that it is no longer a member of the multicast group. If the receiver is assigned an Mc-ACK slot of the PHY multicast group, it shall continue acknowledging in its assigned slot until its assignment for this Mc-ACK slot is terminated by an MC_GrpInfoUpdate.ind message. From the time of receiving MC_GrpRemove.req until the assignment of its Mc-ACK slot is terminated by an MC_GrpInfoUpdate.ind message, the receiver shall set the FACK field to 111_2 (see clause 7.1.2.3.2.3.9.1.5 of [ITU-T G.9960]).

The transmitter may split an existing PHY multicast ~~stream~~ group into several PHY multicast groups, for example, when new receivers with very different BAT join. The transmitter shall assign a new multicast DID to each of the newly created PHY multicast groups and shall send MC_GrpInfoUpdate.ind, which includes the information describing the new PHY multicast groups, to all nodes associated with that PHY multicast ~~stream~~group, using either separate unicast DIDs, broadcast DID or other multicast group DIDs.

~~Splitting of an existing multicast group or moving of receivers from one multicast group to another is for further study.~~

The transmitter shall follow the actions described in clause 8.16.1 each time it sends MC_GrpInfoUpdate.ind for informing on new PHY multicast groups or for updating existing PHY multicast group information.

The transmitter shall allocate a new BAT ID for a PHY multicast group when a change is required in any of the active BAT IDs of the PHY multicast group.

The MC_GrpInfoUpdate.ind message sent by the transmitter shall include the list of all BAT_IDs that are to be active in the PHY multicast group ~~or groups~~ (inside the McstGroupInfo field, see Table 8-107). This consists of those BAT_IDs that are to be retained and new BAT_IDs to be added. BAT_IDs to be removed shall be excluded from the list. New BAT_IDs in this list are accompanied ~~with~~ by BATInfo fields (see Table 8-109). The transmitter shall not start using the new BATs until all the PHY multicast group receivers have confirmed the change. Once ~~a new BAT is used for transmission by~~ the transmitter actually uses a new BAT in transmission, the receivers of the PHY multicast group shall invalidate the old BATs assigned to that PHY multicast group that were excluded from the list in the McstGroupInfo field of the last received MC_GrpInfoUpdate.ind message.

In case the transmitter detects a change in the PHY multicast ~~binding~~group information while awaiting confirmation from the receivers, it shall restart the procedure generating full binding information and retransmitting MC_GrpInfoUpdate.ind with a higher sequence number.

51) Clause 8.17.1, DLL multicast stream establishment (from ITU-T G.9961 Corrigendum 1)

Revise the text of clause 8.17.1 "DLL multicast stream establishment" (from ITU-T G.9961 Corrigendum 1) as follows:

A~~n ITU-T G.9960~~ node that determines that it has to transmit a multicast stream to client nodes in the domain, shall establish a path to each one of the client nodes. The source node that generates the DLL multicast stream shall first allocate an MSID that together with the DEVICE_ID of the source node, shall uniquely identify the DLL multicast stream. Valid values of MSID are from 1 to 250.

The source node shall also initialize the Transaction ID for that DLL multicast stream to zero. The source node shall increment the Transaction ID for each new DLL multicast path it establishes for that DLL multicast stream.

…

52) New clause 8.18, Inter-bandplan interoperability

Add the new clause 8.18 "Inter-bandplan interoperability" as follows:

8.18 Inter-bandplan interoperability

ITU-T G.9960/1 specifies, for each of the supported mediums, transceivers capable of operating with different bandplans. Transceivers of different bandplans, specified for the mediums that the domains work on, may be used in the same domain and shall be capable of interoperating with transceivers of other bandplans, specified for the same medium. This requirement does not apply to transceivers working on disjoint frequency bands since they cannot interoperate. This clause specifies the means provided by this Recommendation by which transceivers of different bandplans can interoperate, by either specifying these means or referencing other clauses of the ITU-T G.9960/1 Recommendations which specifies them.

8.18.1 Bandplan-related information

For allowing nodes of different bandplans to communicate, the following parameters are use in this Recommendation:

- Bandplan-related capabilities of a node:
  - The node's maximal reported bandplan: This is the maximal bandwidth that a node supports, and is reported by the node during its registration. This maximal bandplan cannot be changed during the time the node is registered to the domain. A node is allowed to register to a domain only if its bandplan is within the range indicated by the Minimal bandplan and Maximal bandplan allowed by the domain (see Table 8-77).
  - StartSubCarrier and StopSubCarrier: The first and last sub-carrier indexes supported by the node. The StartSubCarrier and StopSubCarrier are within the node's maximal reported bandplan. The bandwidth determined by the StartSubCarrier and StopSubCarrier shall be equal to or lower than the bandwidth associated with the node's maximal reported bandplan. If it is lower, it shall be lower than the bandwidth of this maximal bandplan by no more than 15% of the non-masked sub-carriers. The StartSubCarrier and StopSubCarrier cannot be changed while the node is registered to the domain.

  The bandplan related information (i.e., the node's maximal reported bandplan and StartSubCarrier and StopSubCarrier) of each node in the domain is made available to all nodes in the domain, as specified in clause 8.18.3.
- Configured minimal and maximal bandplans for the domain: Values configured by a service provider (or user) to determine the minimal and maximal bandplan allowed in the domain. See Table 8-77 and clause 8.18.2.
- Bandplan-related information for payloads using pre-defined BATs: The BNDPL (bandplan identifier) field in the PFH of the MSG PHY frame and the MAP PHY frame is used to identify the bandplan in which this frame's payload was transmitted (whenever the payload is transmitted using pre-defined BATs). The bandplan indicated by the BNDPL field may be lower than the node's maximal reported bandplan indicated by the node during registration and it may be lower than the minimal bandplan configured for the domain. For MSG PHY frames transmitted by a node, this (intermediate) bandplan may be different

from one frame to another (even if the destination node is the same node). The BNDPL field is not relevant for payloads which are using runtime BATs.

- Bandplan-related information for payloads using runtime BATs: For payload transmissions using runtime BATs, the CE_ParamUpdate.req and the MC_GrpInfoUpdate.ind messages include the following set of relevant parameters for each BAT_ID associated with a specific runtime BAT:
  - Bandplan ID: The bandplan of the specific BAT associated with the BAT_ID.
  - $TIDX_{MIN}$ and $TIDX_{MAX}$ of the specific BAT associated with the BAT_ID.

NOTE – Given $TIDX_{MIN}$ and $TIDX_{MAX}$, the Bandplan ID information is redundant.

8.18.2 Configuring the minimal and maximal bandplan for the domain

A service provider (or user) configuring the home network may configure the minimal and maximal bandplans allowed in the domain. These parameters are specified in [ITU-T G.9962] (see clauses 7.4.9 and 7.4.10 of [ITU-T G.9962]) and are published in the MAP's PSD-related domain info sub-field (see Table 8-77). The default values for the minimal and maximal configured bandplans shall be set to the lowest and highest bandplan specified in the Recommendation for the specific medium, respectively (for example, for powerline, the default minimum bandplan is 25 MHz, and the default maximum bandplan is 100 MHz).

A node trying to join the domain shall only try to join when its bandplan is within the range set by the minimal and maximal configured bandplans. The DM shall reject registration requests from nodes which indicate (in the ADM_NodeRegister.req and the TM_NodeTopologyChange.ind messages) that their maximal reported bandplan is outside the minimal and maximal bandplans range configured for the domain.

A node may transmit any type of PHY frame (when the payload is either transmitted with a pre-defined or runtime BAT) in a bandplan which is lower than the minimal bandplan configured for the domain.

8.18.3 Conveying bandplan-related information to all nodes in the domain

In order to allow inter-bandplan operation over a domain, every node communicating with another node within the domain needs to know the bandplan-related information (the node's maximal reported bandplan, StartSubCarrier and StopSubCarrier) of that node. In order to achieve that, the Recommendation specifies the following mechanisms:

1) **Registration**: The registering node reports its maximal supported bandplan (referred to as the "node's maximal reported bandplan") and the StartSubCarrier and StopSubCarrier in the ADM_NodeRegister.req message.

2) **Conveying the routing information from the DM to all nodes of the domain**: The DM reports the maximal reported bandplan and StartSubCarrier and StopSubCarrier of all nodes of the domain in the TM_DomainRoutingChange.ind message.

3) **Topology information report of a node to the DM**: A node reports its maximal bandplan and its StartSubCarrier and StopSubCarrier whenever it sends its TM_NodeTopologyChange.ind message to the DM.

The bandplan-related information (the node's maximal reported bandplan and StartSubCarrier and StopSubCarrier) is carried in the above-mentioned messages in the "Bandplan Info Capability Value" field, which is part of the NodeVersionTLVs.

8.18.4 Transmissions of MAPs to guarantee inter-bandplan interoperability

To facilitate operation of domains with devices of different bandplans, the MAP PHY frames are transmitted using transmission parameters as specified in clause 7.1.2.3.2.1.10 of [ITU-T G.9960] and clause 8.8.2 of [ITU-T G.9961].

8.18.5 Inter-bandplan payload transmissions

In order for two nodes to communicate with each other using runtime BATs, the following rules shall apply:

- A receiver performing channel estimation with a transmitter shall consider its own bandplan information (namely the StartSubCarrier and StopSubCarrier) and that of the transmitter. More specifically, the range of sub-carriers of the BAT sent in the CE_ParamUpdate.req message shall be within the intersection of the sub-carrier ranges determined by the StartSubCarrier and StopSubCarrier of both the receiver and transmitter.
- A transmitter sending BAT information to receivers belonging to a PHY multicast group, shall consider its own bandplan information (namely the StartSubCarrier and StopSubCarrier) and that of the receivers of the PHY multicast group. More specifically, the range of sub-carriers of the BAT sent in the MC_GrpInfoUpdate.ind message shall be within the intersection of the sub-carrier ranges determined by the StartSubCarrier and StopSubCarrier of both the transmitter and all receivers in the PHY multicast group.
- For MAP transmission see clause 8.18.4. For other transmissions using pre-defined BATs the transmitter may choose either to:
  - Transmit using the lowest bandplan between the receiver and transmitter bandplans or,
  - Transmit using its own bandplan using a sufficient repetition factor (e.g., a node of bandplan 100 MHz can transmit a frame using 100 MHz bandplan with repetition factor of 2, to be received by a node of 50 MHz bandplan).

NOTE – Whenever the transmitter is of a higher bandplan than that of the receiver, transmissions outside the frequency band supported by the receiver (i.e., transmitting power on sub-carriers higher than the receiver's StopSubCarrier) might cause degradation of the receiver's performance in some cases depending on the receiver implementation. It is therefore recommended, if the transmitter's implementation allows it, to not output any power on the sub-carriers outside the frequency band supported by the receiver (i.e., on sub-carriers beyond the receiver's StopSubCarrier).

53) New clause 8.19, Version control and capabilities exchange

Add text for new clause 8.19 "Version control and capabilities exchange" as follows:

8.19 Version control and capabilities exchange

Each node that enters the domain shall inform its domain master about the version of the ITU-T Recommendations that it implements and the capabilities it can support. It shall include at least one node versioning TLV (see Table 8-16.1) in the ADM_NodeRegistrRequest.req message. The first node versioning TLV included in the message shall correspond to an ITU Versioning Type TLV (see Table 8-16.2). In addition, if a node's capabilities change (i.e., any of the node's versioning TLVs change) during its operation, it shall inform its DM via the TM_NodeTopologyChange.ind message, by including the relevant Node Versioning TLVs.

The DM shall inform the rest of the nodes in its domain whenever it receives new versioning information from a node by using the TM_DomainRoutingChange.ind message.

The versioning dependencies between a Recommendation of the ITU-T G.996x family and other Recommendations of that family shall be as described in Annex V "Versioning dependencies" of each Recommendation.

54) Clause 9.1.1.3, Input variables

Revise the text of clause 9.1.1.3 "Input variables" as follows:

9.1.1.3 Input variables

The input variables to support CCM encryption are:

– counter blocks (Ctr_n);

– n~~N~~once block (B_0);

– associated data blocks (~~B_0,~~ B_1, and B_2);

– payload blocks (B_3 to B_r);

– encryption key.

The 16-byte counter blocks Ctr_0, Ctr_1…, Ctr_m shall have the format presented in Table 9-2. Each block shall comprise a 1-byte flag, a 13-byte nonce, and a 2-byte counter block number (in the range from 0 to m). All bytes of the counter block shall be formatted MSB first: the first bit of the byte 0 is the MSB (bit 7) and the last bit of the byte 15 is the LSB (bit 0). The counter block number shall be represented as a 16-bit binary integer where the LSB is the LSB of byte 15.

Table 9-2 – Format of the Ctr blocks

Byte number	**0**	**1, 2 …, 13**	**14, 15**
Contents	Flags (Note)	Nonce	Counter block number
NOTE – The content of the Flags byte is: bits [7:6] – reserved by ITU-T for NIST, shall be set to 00_2. bits [5:3] – shall be set to 000_2. bits [2:0] – shall be set to 001_2.			

The 13-byte nonce shall be constructed as presented in Table 9-3. The MSB of byte 0 of the nonce in Table 9-3 shall be mapped to the MSB of byte 1 of the Ctr block. The value and format of the frame number (FN) shall be as specified in clause 9.1.2 (Table 9-6). The LSB of the FN shall be mapped to the LSB of byte 12 of the nonce, and byte 7 of the nonce shall be set to 00_{16}. The source MAC address of the APDU or LCDU shall have a standard IEEE 802.3 format where the MSB shall be mapped to the MSB of byte 1 of the nonce. All bytes of the nonce shall be formatted MSB first: the first bit of the byte 0 is MSB (bit 7) and the last bit of the byte 12 is LSB (bit 0).

Table 9-3 – Format of the nonce

Byte number	**0**	**1 – 6**	**7**	**~~7~~8 – 12**
Contents	Flags (Note)	Source MAC address	00_{16}	Frame number (FN)
NOTE – The content of the Flags byte is: Bits [7:3] – the same bits of Byte 0 of the CCMP header. Bits [2:0] – reserved by ITU-T. All reserved bits of the Flags byte shall be set to zero.				

The value of the nonce (for the given key) shall never be the same for different encrypted payloads, and shall always be the same for identical encrypted payloads (e.g., when APDU or LCDU is retransmitted or relayed). The encryption key shall be changed promptly to avoid repetition of the nonce (see clause 9.1.2.3).

The ~~associated data blocks B_0 B_2 shall each be~~ 16-byte~~s long.~~ nonce ~~B~~block B_0 shall have a format as presented in Table 9-4. The length of the encrypted payload in octets (*Plen*) shall be represented as a 16-bit unsigned integer with the LSB mapped to the LSB of byte 15 of B_0.

Table 9-4 – Format of block B_0

Byte number	0	1, 2…, 13	14, 15
Content	Flags (Note)	Nonce	Length of the payload (*Plen*)
NOTE – The content of the Flags byte is: Bit [7] – Reserved by ITU-T for NIST, shall be set to zero. Bit [6] – Shall be set to one. Bits [5:3] – Shall indicate the length of the MIC encoded as: 001_2 – 4-byte MIC. 011_2 – 8-byte MIC. 111_2 – 16-byte MIC. All other values are reserved by ITU-T. Bits [2:0] – Shall be set to 001_2			

The two 16-byte associated data ~~B~~blocks B_1~~,~~ and B_2 shall have a format as presented in Table 9-5. Byte 0 is the first byte and byte 15 is the last byte.

Table 9-5 – Format of blocks B_1, B_2

Block	Bytes	Contents (Note 1)
B_1	0 and 1	Length of associated data in bytes (*Alen*), expressed as an unsigned integer (Note 2).
	2 and 3	Reserved by ITU-T (Note 3~~2~~).
	4 to 9	Destination MAC address.
	10 to 15	Source MAC address.
B_2	0 to 4~~3~~	Portion of LFH excluding bytes containing TTL and TSMP fields (Note 4~~3~~).
	4~~5~~ to (3~~4~~ + *V*)	Additional unencrypted field. APDU (EAPC): *TG* bytes of ~~EtherType/~~VLAN TAGs plus 2 bytes of MAC client length/type (*V* = *TG* + 2, see Figure A.1). LCDU: 2 bytes of EtherType (*V* = 2, or equivalent).
	(4~~5~~ + *V*) to 15	~~Remainder of the associated data (~~Zero padding as specified in [NIST 800-38C]~~Reserved by ITU-T, Note 2)~~.
NOTE 1 – All fields are mapped so that the most significant byte of the value associated with a particular field is mapped onto the byte with the smaller sequential number. NOTE 2 – For APDU (EAPC), *Alen* shall include byte 2 of B_1 to byte 6+*TG* of B_2 (21+*TG* bytes). For LCDU, *Alen* shall include byte 2 of B_1 to byte 6 of B_2 (21 bytes). NOTE 3~~2~~ – Bits that are reserved by ITU-T shall be set to zero~~ by the transmitter and ignored by the receiver~~. NOTE 4~~3~~ – Byte 0 to byte 4~~3~~ of B_2 shall correspond to byte 0 to byte 4~~3~~ of LFH, respectively (Table 8-1). The bit corresponding to the CCMPI field of LFH shall be set to 1.		

All bytes of the nonce and the associated data blocks shall be formatted MSB first: the first bit of the byte 0 is MSB (bit 7) and the last bit of the byte 15 is LSB (bit 0).

Payload blocks (B_3 to B_r) are 16-byte long and shall contain bytes of the APDU or LCDU to be encrypted (see clause 9.1.2.2, encrypted part of APDU or LCDU). The APDU or LCDU bytes shall be mapped to payload blocks in sequential order, so that the first byte of the APDU or LCDU to be encrypted is mapped to byte 0 of B_3, the second byte of the payload is mapped to byte 1 of B_3, the 17-th byte of the APDU or LCDU is mapped to byte 0 of B_4, and so on. If the last byte of the

payload does not fall on byte 15 of B_r, the payload shall be padded to fill the last block by appending zero bytes (00_{16}). All bytes of the payload blocks shall be formatted MSB first: the first bit of byte 0 of block B_3 is the MSB (bit 7) and the last bit of byte 15 of block B_r is LSB (bit 0).

The encryption key is 128 bits long and shall be generated and assigned as described in clause 9.2.

55) Clause 9.1.2.3, CCMP header

Revise the text of clause 9.1.2.3 "CCMP header" as follows:

The CCMP header consists of six bytes and shall have a format as presented in Table 9-6. It carries the encryption key identification number (key-ID), the type of the encryption key, the length of the MIC, and the security frame number (FN). These ~~three~~ four parameters are necessary for decryption.

The length of the MIC shall be selected according to the procedure defined in clause 9.2.3.

Table 9-6 – CCMP header format

Field	Octet	Bits	Description
CCMP header	0	[2:0]	Length of the MIC encoded as: 001 – 4-byte MIC. 011 – 8-byte MIC. 111 – 16-byte MIC. All other values are reserved by ITU-T.
		[~~5:~~3]	Reserved by ITU-T (Note).
		[5:4]	The type of encryption key: 00 – NN key or NMK. 10 – DB key. 01 – NSC key. 11 – Reserved by ITU-T.
		[~~7:~~6]	Encryption key ID, formatted as an unsigned binary integer.
		[7]	Reserved by ITU-T (Note).
	1 to 5	[39:0]	40-bit FN, formatted as an unsigned binary integer.
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

The key-ID identifies the used encryption key among those assigned to the communicating nodes during the AKM procedure, as described in clause 9.2.5.2. Keys assigned for communication with different peers may have the same key-IDs. The keyID shall be formatted as a 1-bit unsigned binary integer. ~~The range~~Valid values of the key-ID ~~is from~~are 0 ~~to 3~~and 1. ~~The format of the key ID is a 2-bit unsigned binary integer.~~

The FN is a serial number of the encrypted LLC frame and shall be represented as a 40-bit unsigned binary integer. The FN shall be set to one when a new encryption key is established and increased by one with every encrypted LLC frame passed using this key. FN shall never be repeated for the same value of the key: the key shall be changed prior to FN reaching its maximum value. FN expiration is used as a trigger for key~~s~~ update (see clause 9.2.4).

In order to allow some time for the FN update procedure before the FN repeats itself, whenever an FN covers 95% of its maximum value, the key update procedure for the corresponding key shall be started. This would apply to all the keys, namely NSC, NN, NMK, DB and multicast keys. In the case of NMK, DB and multicast keys it is possible that multiple nodes realize that the keys need to be updated at the same time, so a random delay in the range 0 to 5s is added between when a node realizes that it needs the key to be updated and when it communicates this to the SC. The node shall

communicate to the SC that the key needs to be updated only if the key is not updated by the time its random delay timer expires.

For the DB, the NMK and multicast keys the FN is initialized to a value that is composed of the node's DEVICE_ID in the most significant byte of the FN field and zeros in all other bytes of the FN field. This will ensure that a node will repeat an FN value already used by another node for the same key with a very low probability.

NOTE – On the receive side, the FN may not appear to be sequential, if the order in which packets are encrypted and transmitted is different.

56) Clause 9.2.1, Authentication and key management procedures

Revise the text of clause 9.2.1 "Authentication and key management procedures" as follows:

9.2.1 Overview

Authentication and key management (AKM) defines a set of procedures allowing a node to join a secure domain and to operate in it with point-to-point and point-to-multipoint security. AKM includes the following main procedures:

– Aauthentication to the domain in secure mode;

– Eestablishing point-to-point encryption keys for unicast communication;

– Establishing point-to-multipoint encryption keys for multicast communication;

– Pperiodic re-authentication and updating point-to-point and point-to-multipoint encryption keys.

To set a node for secure operation, it shall be provided with a password. The node password shall comply with the characteristics presented in clause 9.2.2.1. Passwords shall never be communicated, even if encrypted. A particular way to establish a node password is vendor discretionary and beyond the scope of this Recommendation.

Prior to authentication to the domain in secure mode, the node shall first register with the domain master using the admission procedure described in clause 8.6.1. The domain master shall indicate to the registering node that the domain operates in secure mode by setting the security field of the ADM_DmRegistrResponse.cnf message to "Secure". A registered node ~~can further~~shall then apply for authentication to operate in secure mode. Authentication shall be performed as described in clause 9.2.2. A node that is not authenticated shall not attempt to communicate with other nodes of the secure domain. In a domain operating in secure mode, all the communications within the domain between authenticated nodes (except for MAPs) shall be encrypted with the appropriate key.

NOTE – A node that is not yet authenticated can communicate without encryption with the DM and the SC (either directly or via proxy node) for the purpose of registration and authentication respectively.

An authenticated node can establish encryption keys for secure unicast, multicast, and broadcast communications inside the domain~~ and communications to nodes of other secure domains~~. Point-to-point encryption keys shall be established using the procedures described in clause 9.2.3.

The procedures described in clause 9.2.4 shall be used for periodical re-authentication and updates of encryption keys.

A flowchart of AKM procedures is presented in Figure 9-4.

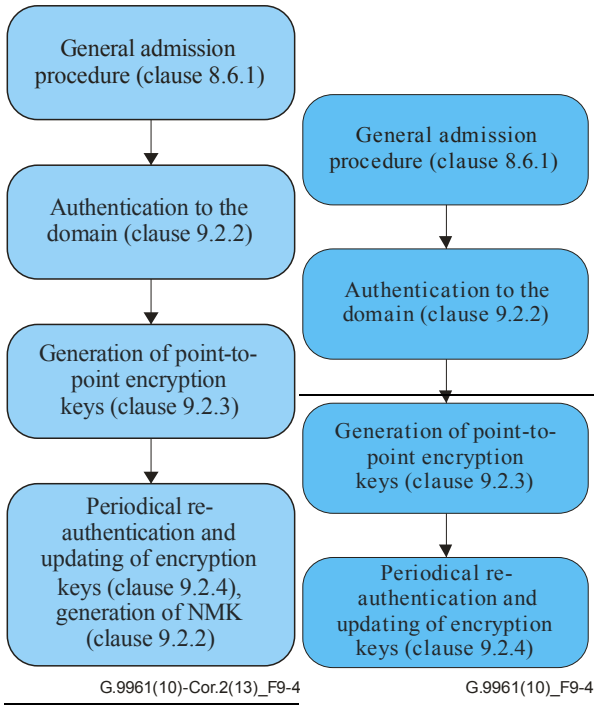


Figure 9-4 – Flowchart of AKM

AKM procedures in the domain are managed by the security controller (SC), which may be an additional function of the domain master or an endpoint node.

When the SC is implemented by an endpoint node, the DM shall start authentication as soon as possible, after this endpoint node has completed its registration with the DM. In this case, the authentication of this endpoint node is done internally.

A DM in a secure domain that is not functioning as the SC is allowed to transmit MAPs, registration confirmation messages to the SC and authentication requests to the SC, before being authenticated. In a secure domain, if a node sends a registration request when the DM is not yet authenticated, the DM shall reject the registration request by registration response with status 'DM not authenticated' (see Table 8-15).

If the DM and SC are not co-located then the REGID of the SC shall be configured to the DM and the DM shall only allow registration of the SC until it is authenticated by the SC.

The SC can be configured to use a single encryption key per domain/network, to use the network membership key (NMK) or to use a key per connection (NN). NMK is granted to the node during its authentication, as specified in clause 9.2.2.1. In case of using NMK, the AKM procedures intended for generating point-to-point encryption keys (clause 9.2.3) are skipped.

57) Clause 9.2.2, Authentication to the domain

Revise the text of clause 9.2.2 "Authentication to the domain" as follows:

9.2.2 Authentication to the domain

For operation in secure mode, a registered node shall authenticate itself to the security controller (SC) as described in this clause. A node that is not authenticated by the SC shall not attempt to communicate (both transmit and receive) with any other node in the secure domain, until it ~~gets~~is authenticated (except for the purpose of registration and authentication) ~~operating in secure mode~~. After authentication, the node can be a part of the domain operating in secure mode.

NOTE – Authentication of the devices joining the domain with a remote facility (e.g., a broadband service provider) requires a trusted channel between the remote facility and the user or between the remote facility and the SC. Set up of this channel and related communication protocols is beyond the scope of this Recommendation. In this case, it is assumed that a remote authenticator, as necessary, may perform some SC functions and it controls operation of the SC.

58) Clause 9.2.2.1, Authentication

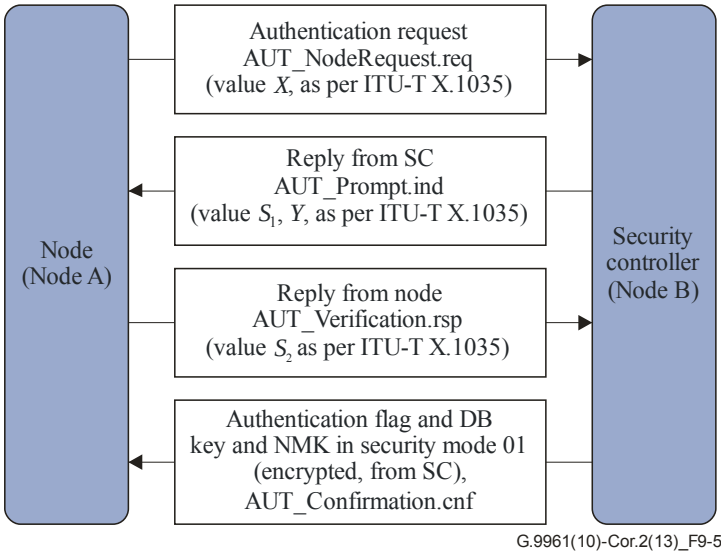
Revise the text of clause 9.2.2.1 "Authentication" as follows:

9.2.2.1 Authentication

Authentication to the SC shall use the Password-Authenticated Key Exchange (PAK) protocol defined in [ITU-T X.1035] with protocol parameters specified in clause 9.2.2.2. The procedure is described in Figure 9-5. It assumes two nodes, called Supplicant (Node A in Figure 9-5, the node requesting authentication) and Authenticator (Node B in Figure 9-5, the SC), which both share the node password PW. The Supplicant shall initiate a Diffie-Hellman handshake with the Authenticator specified in [ITU-T X.1035]. The handshake results that the Supplicant and the Authenticator co-generate a Node-to-SC (NSC) encryption key, ~~K,~~ which shall only be used for encryption of secure communications between the node and the SC.

NOTE 1 – The NSC key is used only for communication with the SC function of the node. For secure communications with other clients associated with a node containing the SC function, NMK, DB, or NN keys are used, as defined in clause 9.2.3.

NOTE 2 – The PAK protocol, with very high probability, returns a new encryption key after each run.



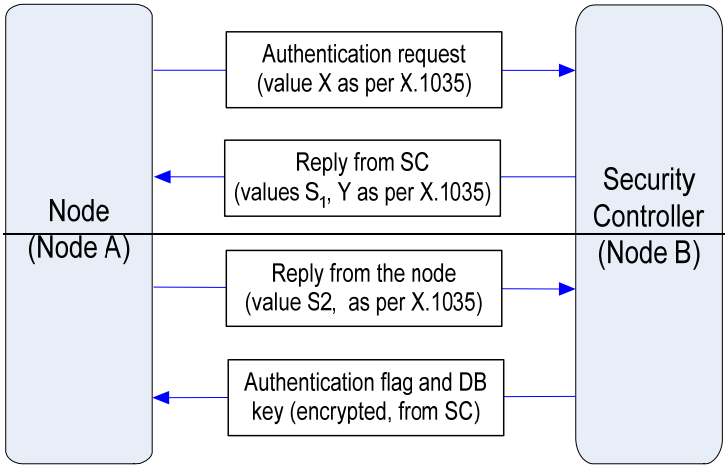


Figure 9-5 – PAK handshake procedure

The procedure shall include the following steps presented in Figure 9-5. The format of the authentication messages supporting the procedure shall be as described in clause 9.2.5.1.

1) The Supplicant shall initiate~~s~~ the authentication procedure with the SC by sending to the SC an authentication request (AUT_~~NodeRequest~~NodeAuthentication.req) message which includes the REGID ~~name~~ of the node, the REGID~~name~~ of the SC as specified in the registration response, and the node password known to the SC (values *A*, *B*, *PW*, respectively, as per [ITU-T X.1035]), hashed into value of *X*~~,~~ [~~5~~ITU-T X.1035]. The values of *A*, *B*, and *PW* shall be as defined in Table 9-7. The AUT_~~NodeRequest~~NodeAuthentication.req message shall be sent unencrypted.

2) The SC shall verify~~ies~~ the received value of *X* and reply~~ies~~ to the Supplicant with an authentication prompt (AUT_Promp.ind) message, including values S_1 and *Y*, as per [ITU-T X.1035]. The message shall be sent unencrypted within 800 ms after reception of the AUT_Node~~Request~~Authentication.req message. If the SC determines that the value of *X* is invalid, it shall reply with AUT_Promp.ind status set to one.

   NOTE 3 – The SC identifies the node providing the value of X using the node MAC address (SA of the LCDU carrying the AUT_~~NodeRequest~~NodeAuthentication.req message).

3) The node shall verify~~ies~~ the prompt, compute~~s~~ the value S_2, as per [ITU-T X.1035], and send~~s~~ it to the SC in the authentication prompt verification (AUT_Verification.rsp~~es~~) message. The message shall be~~is~~ sent unencrypted within 800 ms after reception of the AUT_Promp.ind. If the node determines that the value of S_1 or the value of *Y* is invalid, it shall reply with AUT_Verification.rsp with the status set to one.

4) Using the exchanged variables S_1 and S_2 both nodes shall compute independently the 128-bit NSC encryption key (value of K as per [ITU-T X.1035]).

5) The SC shall send~~s~~ to the Supplicant the authentication confirmation (AUT_Confirmation.cnf) message, which includes the confirmation flag, NMK, and the in-domain broadcast (DB) encryption key. The AUT_Confirmation.cnf message shall be sent encrypted by NSC within 800 ms after reception of the AUT_Verification.rsp message.

9.2.2.1.1 Authentication failure

The node shall consider the authentication process failed in the following conditions:

- The node does not receive the AUT_Promp.ind within 1 s after it sent AUT_NodeAuthentication.req message; or
- The node receives the failure indication in the AUT_Promp.ind message; or

- The node does not receive the AUT_Confirmation.cnf message within 1 s after it sent AUT_Verfication.rsp; or
- The node receives the failure indication in the AUT_Confirmation.cnf message.

If the SC does not receive AUT_Verification.rsp message within 1 s after it sent AUT_Promp.ind message, it shall abort the authentication process.

~~If any one of the steps fails, SC shall send AUT_Confirmation.cnf with a confirmation flag set off. In case the confirmation flag is off, or the node can't decrypt the AUT_Confirmation.cnf, or it didn't receive AUT_Confirmation.cnf during 200 ms after it sent the value S_2, the node shall consider~~In case the authentication process failed,. ~~T~~the Supplicant may start re-authentication in time period greater than 1 s from the instan~~ee~~t it detects the failure, and shall not transmit any data ~~from the time it received AUT_Confirmation.cnf~~ until it starts re-authentication. After four~~4~~ unsuccessful re-authentication attempts, the SC shall request the domain master by sending the SC_DMRes.req message to resign the node (Supplicant) from the domain using forced resignation, as described in clause 8.6.1.1.3.2. The domain master, upon receiving the SC_DMRes.req message, shall reply with an SC_DMRes.cnf message within 100 ms. If the SC does not receive an SC_DMRes.cnf message within 200 ms, it shall retry sending the SC_DMRes.req message.

9.2.2.1.2 Successful authentication

The node whose authentication was confirmed is allowed to broadcast and receive broadcast messages from other secure nodes of the domain using the DB encryption key. If the domain is configured to operate in point-to-point mode, the node ~~and~~ can request from the SC point-to-point encryption keys to communicate with other nodes operating in secure mode as described in clause 9.2.3.

If the SC is configured to operate with NMK, it shall send the NMK and DB encryption key to the authenticated node in an AUT_Confirmation.cnf message (see clause 9.2.5.1.4). The authenticated node is allowed to broadcast and receive broadcast messages using the DB encryption key and may communicate with other nodes using the NMK encryption key in this mode.

If the SC is configured to operate with NN key, it shall send only the DB encryption key to the authenticated node in AUT_Confirmation.cnf message.

9.2.2.1.3~~1~~ Authentication via proxy

If a node cannot communicate with the SC directly, it shall authenticate itself with the SC using other nodes as relays. In a secure domain, a node is not allowed to send its topology update messages and cannot read topology updates sent by other nodes prior to authentication. Thus, the registration proxy may be used as the first relay between the SC and the node.

~~The node starts authentication by sending AUT_NodeRequest.req message encapsulated into an LCDU where the destination address is the MAC address of the SC, and uses the DID in the PHY frame carrying AUT_NodeRequest.req equal to the DEVICE_ID of the first relay (which is its registration proxy).~~

~~After the node has transmitted AUT_NodeRequest.req, it shall wait for an AUT_Promp.ind message (expected to come from the node that was used as the first relay). After AUT_Promp.ind is received, the node shall reply by AUT_Verification.res using again the same considerations as when it sends AUT_NodeRequest.req, and wait for AUT_Confirmation.cnf to complete the authentication process.~~

~~Messages AUT_Promp.ind and AUT_Confirmation.cnf are sent by the SC encapsulated in LCDUs with the destination address equal to the MAC address of the supplicant..~~

~~NOTE – To avoid multiple authentication attempts, a node may delay the start of authentication procedure after it gets registered to accommodate topology update initiated by the registration proxy (see clause 8.6.1.2).~~

The authenticating node shall start authentication by sending an AUT_NodeAuthentication.req message with the ProxyAuth field set to 1 and the addressing fields as defined in Table 9-6.1.

Table 9-6.1 – Addressing fields of the AUT_NodeAuthentication.req from authenticating node to proxy node

Field	Value
DA of the LCDU carrying the message	REGID of the SC
SA of the LCDU carrying the message	REGID of the authenticating node
OriginatingNode of the LLC frame carrying the LCDU	DEVICE_ID of the authenticating node
DestinationNode of the LLC frame carrying the LCDU	DEVICE_ID of the proxy node
SID of the PHY frame carrying the message	DEVICE_ID of the authenticating node
DID of the PHY frame carrying the message	DEVICE_ID of the proxy node

Upon receiving the AUT_NodeAuthentication.req message the proxy node shall relay the AUT_NodeAuthentication.req message to the SC by setting the addressing fields as defined in Table 9-6.2.

Table 9-6.2 – Addressing fields of the AUT_NodeAuthentication.req from proxy to the SC

Field	Value
DA of the LCDU carrying the message	REGID of the SC
SA of the LCDU carrying the message	REGID of the authenticating node
OriginatingNode of the LLC frame carrying the LCDU	DEVICE_ID of the proxy node
DestinationNode of the LLC frame carrying the LCDU	DEVICE_ID of the SC
SID of the PHY frame carrying the message	DEVICE_ID of the proxy node
DID of the PHY frame carrying the message	DEVICE_ID of the SC or the next relay node towards the SC in case where the proxy does not have a direct link with the SC

Upon receiving the AUT_NodeAuthentication.req message the SC shall detect that the received AUT_NodeAuthentication.req message was sent by a proxy node. It shall extract the REGID of the authenticating node from the SA of the LCDU carrying the AUT_NodeAuthentication.req message to identify the authenticating node. The SC shall validate the AUT_NodeAuthentication.req message as described in the regular authentication procedure (see clause 9.2.2.1) and build accordingly the AUT_Promp.ind message with the ProxyAuth set to 1. The SC shall set the addressing fields as defined in Table 9-6.3.

The AUT_Promp.ind message shall be sent unencrypted within 800 ms after receipt of the AUT_NodeAuthentication.req message. If the SC determines that the value of *X* is invalid, it shall reply with an AUT_Promp.ind message with the status set to one.

Table 9-6.3 – Addressing fields of the AUT_Promp.ind from SC to the proxy node

Field	Value
DA of the LCDU carrying the message	REGID of the proxy node
SA of the LCDU carrying the message	REGID of the SC
OriginatingNode of the LLC frame carrying the LCDU	DEVICE_ID of the SC
DestinationNode of the LLC frame carrying the LCDU	DEVICE_ID of the proxy node
SID of the PHY frame carrying the message	DEVICE_ID of the SC
DID of the PHY frame carrying the message	DEVICE_ID of the proxy node or the next relay node towards the proxy in case where the SC has not direct link with the proxy

Upon receiving the AUT_Promp.ind message the proxy node shall then unicast the received AUT_Promp.ind message to the authenticating node using the addressing scheme shown in Table 9-6.4.

Table 9-6.4 – Addressing fields of the AUT_Promp.ind from the proxy node to the authenticating node

Field	Value
DA of the LCDU carrying the message	REGID of the authenticating node
SA of the LCDU carrying the message	REGID of the proxy node
OriginatingNode of the LLC frame carrying the LCDU	DEVICE_ID of the proxy node
DestinationNode of the LLC frame carrying the LCDU	DEVICE_ID of the authenticating node
SID of the PHY frame carrying the message	DEVICE_ID of the proxy node
DID of the PHY frame carrying the message	DEVICE_ID of the authenticating node

Upon receiving the AUT_Promp.ind message, the authenticating node shall verify the prompt, compute the value S_2 as per [ITU-T X.1035], and send to the SC in the AUT_Verification.rsp message by again using the proxy node as the first relay towards the SC with the addressing scheme defined in Table 9-6.1 and wait 1 second for the AUT_Confirmation.cnf message to complete the authentication process.

Upon receiving the AUT_Verification.rsp message, the proxy node shall forward the received message to the SC with the addressing scheme defined in Table 9-6.2.

Upon receiving the AUT_Verification.rsp message, the SC shall verify that the authenticating node is authenticated. If the SC concludes that the authenticating node is authenticated, it shall inform the DM by sending an AUT_NodeAuthenticated.req message that the authenticating node has been authenticated and may be joined to the secured domain. The AUT_NodeAuthenticated.req message is encrypted by the NSC key that it shares with the DM. The AUT_NodeAuthenticated.req message includes the DEVICE_IDs of the authenticating node and the proxy node. The DM shall confirm receiving the AUT_NodeAuthenticated.req message by replying with an AUT_NodeAuthenticated.cnf message and shall then update the routing table to include the authenticating node, and indicate that all routes to the authenticating node go via the proxy node. The updated routing table is then advertised by the DM by broadcasting the updated TM_DomainRoutingChange.ind message. After the SC receives the updated TM_DomainRoutingChange.ind message that includes the authenticating node, it shall send the AUT_Confirmation.cnf message to the authenticating node via the proxy node using the normal

unicast routing procedure described in clause 8.5.7. The addressing fields of the AUT_Confirmation.cnf message from the SC to the authenticating node are shown in Table 9-6.5.

Table 9-6.5 – Addressing fields of the AUT_Confirmation.cnf from SC to the authenticating node

Field	Value
DA of the LCDU carrying the message	REGID of the authenticating node
SA of the LCDU carrying the message	REGID of the SC
OriginatingNode of the LLC frame carrying the LCDU	DEVICE_ID of the SC
DestinationNode of the LLC frame carrying the LCDU	DEVICE_ID of the authenticating node
SID of the PHY frame carrying the message	DEVICE_ID of the SC
DID of the PHY frame carrying the message	DEVICE_ID of the proxy node or the next relay node towards the proxy node in case the SC does not have direct link to the proxy node

The failure of a node's authentication and related actions by the node and the SC are as described in clause 9.2.2.1.1.

The steps that the authenticating node takes after its successful authentication to become part of a secure domain are as described in clause 9.2.2.1.2.

59) Clause 9.2.2.2, The PAK protocol parameters

Revise the text of clause 9.2.2.2 "The PAK protocol parameters" as follows:

9.2.2.2 The PAK protocol parameters

The PAK parameters used for node authentication shall comply with the requirements listed in Table 9-7. ~~The detailed requirements for selection of the values for PW are for further study.~~

Table 9-7 – ITU-T X.1035 – PAK parameters

ITU-T X.1035 parameter	Description	Length (bits)	Notes
~~Node name (*A*, *B*)~~	Node identifiers of the supplicant and authenticator~~Value of the unicast DEVICE_ID granted to the node~~	48	Clause 9.2.2.2.1
~~Password (*PW*)~~	~~User~~ Node password of the ~~authenticated node~~supplicant	96~~ bits (12 ASCII characters)~~	Clause 9.2.2.2.2
p	Diffie-Hellman ~~constant (~~prime ~~value)~~	1024	Clause 9.2.2.2.3~~recommended by X.1035~~
g	Diffie-Hellman generator	~~160~~8	Clause 9.2.2.2.4~~recommended by X.1035 (value from TIA-683-D~~

Table 9-7 – ITU-T X.1035 – PAK parameters

ITU-T X.1035 parameter	Description	Length (bits)	Notes
R_A, R_B	Secret exponents of the supplicant and authenticator	384	Clause 9.2.2.2.5~~recommended by X.1035~~
H_1	Hash functions of SHA-256 type	1152	Clause 9.2.2.2.6~~recommended by X.1035~~
H_2		1152	Clause 9.2.2.2.6~~recommended by X.1035~~
H_3, H_4, H_5		128	Clause 9.2.2.2.6~~recommended by X.1035~~
K	NSC key	128	Clause 9.2.2.2.7

9.2.2.2.1 Node identifier

The parameters *A* and *B* defined in [ITU-T X.1035] represent the identifiers of the supplicant and the authenticator of the PAK protocol, respectively. *A* shall be set to the 48-bit MAC address of the supplicant. *B* shall be set to the 48-bit MAC address of the security controller.

9.2.2.2.2 Node password

The parameter *PW* defined in [ITU-T X.1035] represents the secret password shared by the supplicant and the authenticator. The size of *PW* shall be 96 bits. Generation of the node password is out of scope of this Recommendation.

9.2.2.2.3 Diffie-Hellman prime

The parameter *p* defined in [ITU-T X.1035] represents the Diffie-Hellman prime, which is a 1024-bit predefined constant. It shall be set to the following number (MSB first):

```
p =FFFF FFFF FFFF FFFF C90F DAA2 2168 C234 C4C6 628B 80DC 1CD1
   2902 4E08 8A67 CC74 020B BEA6 3B13 9B22 514A 0879 8E34 04DD
   EF95 19B3 CD3A 431B 302B 0A6D F25F 1437 4FE1 356D 6D51 C245
   E485 B576 625E 7EC6 F44C 42E9 A637 ED6B 0BFF 5CB6 F406 B7ED
   EE38 6BFB 5A89 9FA5 AE9F 2411 7C4B 1FE6 4928 6651 ECE6 5381
   FFFF FFFF FFFF FFFF16
```

9.2.2.2.4 Diffie-Hellman generator

The parameter *g* defined in [ITU-T X.1035] represents the Diffie-Hellman generator, which is an 8-bit predefined constant. It shall be set to the following number (MSB first):

```
g =0D16
```

9.2.2.2.5 Secret exponents

The parameters R_A and R_B defined in [ITU-T X.1035] represent the secret exponents selected by the supplicant and the authenticator of the PAK protocol, respectively. They shall be selected randomly as follows:

– The number generated shall be 384 bits in length.

– The number generated shall not be less than 4.

– The number generated shall have a uniform statistical distribution over its range [4, $2^{384}-1$].

9.2.2.2.6 Hash functions

The parameters, H_1, H_2, H_3, H_4 and H_5 defined in [ITU-T X.1035] represent the hashing functions used by the supplicant and authenticator in various stages of PAK protocol. They shall be defined as follows:

$H_1(u_1)$ = SHA-256($00\ 00\ 00\ 01_{16} \mid 00\ 00\ 00\ 01_{16} \mid u_1$) |
SHA-256($00\ 00\ 00\ 01_{16} \mid 00\ 00\ 00\ 02_{16} \mid u_1$) |
SHA-256($00\ 00\ 00\ 01_{16} \mid 00\ 00\ 00\ 03_{16} \mid u_1$) |
SHA-256($00000001_{16} \mid 00\ 00\ 00\ 04_{16} \mid u_1$) |
TRC(SHA-256($00\ 00\ 00\ 01_{16} \mid 00\ 00\ 00\ 05_{16} \mid u_1$), 128);

$H_2(u_2)$ = SHA-256($00\ 00\ 00\ 02_{16} \mid 00\ 00\ 00\ 01_{16} \mid u_2$) |
SHA-256($00\ 00\ 00\ 02_{16} \mid 00\ 00\ 00\ 02_{16} \mid u_2$) |
SHA-256($00\ 00\ 00\ 02_{16} \mid 00\ 00\ 00\ 03_{16} \mid u_2$) |
SHA-256($00\ 00\ 00\ 02_{16} \mid 00\ 00\ 00\ 04_{16} \mid u_2$) |
TRC(SHA-256($00\ 00\ 00\ 02_{16} \mid 00\ 00\ 00\ 05_{16} \mid u_2$), 128);

$H_3(u_3)$ = TRC(SHA-256($00\ 00\ 00\ 03_{16} \mid u_3$), 128);

$H_4(u_4)$ = TRC(SHA-256($00\ 00\ 00\ 04_{16} \mid u_4$), 128); and

$H_5(u_5)$ = TRC(SHA-256($00\ 00\ 00\ 05_{16} \mid u_5$), 128).

SHA-256 is defined in [NIST FIPS 180-3]. $x \mid y$ denotes the concatenation of strings x and y (MSB first) as defined in [ITU-T X.1035]. 00000001_{16} denotes a 32-bit constant in a hexadecimal form. TRC(z, a) denotes the first a MSBs of z (i.e., truncated string). u_1 and u_2 denote the 192-bit inputs to $H_1(\cdot)$ and $H_2()$, respectively. u_3, u_4, and u_5 denote the 3264-bit inputs to $H_3(\cdot)$, $H_4(\cdot)$, and $H_5(\cdot)$, respectively.

9.2.2.2.7 NSC key

The parameter K defined in [ITU-T X.1035] represents the 128-bit NSC key, which is the output of the PAK protocol generated independently by the supplicant and the authenticator.

60) Clause 9.2.3.1, Generation of point-to-point and point-to-multipoint encryption keys

Revise the text of clause 9.2.3.1 "Generation of point-to-point and point-to-multipoint encryption keys" as follows:

9.2.3.1 Generation of point-to-point encryption keys

The procedure to establish NN keys shall include the following steps, also presented in Figure 9-6. The format of the messages supporting the described procedure is defined in clause 9.2.5.2.

1) The supplicant shall send~~s~~ an ~~communication request~~ (AKM_KeyRequest.req~~)~~ message to the SC which includes the DEVICE_ID(s) ~~(or MULTICAST_ID in case of multicast)~~ of the addressee node(s) it intends to communicate with. The message shall be encrypted with NSC of the supplicant. In the case of requesting point-to-multipoint keys, the supplicant shall set the Multicast Stream Identifier to the MSID of the associated DLL multicast group. In the case of requesting point-to-point keys, the Multicast Stream Identifier field shall be set to 0.

2) Upon reception of the AKM_KeyRequest.req message, ~~T~~the SC shall accept~~s~~ the request and shall generate~~s~~ a pair of NN keys (NN_{SA} to be used for supplicant towards the addressee(s), and NN_{AS} to be used by each addressee towards the supplicant) if at least one of the addressees is authenticated. Keys shall not be generated if none of the addressees in the supplicant request are authenticated. In the case of multicast keys the NN_{SA} and NN_{AS} keys shall be same.

3) The SC shall send~~s~~ the generated pair of NN keys to each of the authenticated addressees using the AKM_NewKey.req message; no key shall be ~~generated for~~sent to addressees that are not authenticated. The AKM_NewKey.req message shall be encrypted using the NSC key of the addressee. The addressee shall acknowledge the AKM_NewKey.req message by sending an AKM_NewKey.cnf~~AKM_KeyAck.cnf~~ message to the SC. In case no AKM_NewKey.cnf message~~AKM_KeyAck.cnf~~ is received from a particular addressee during the time period of ~~2~~100 ms, the SC ~~may~~ shall retransmit the message up to four times, and shall remove the addressee from the list if no AKM_NewKey.cnf~~AKM_KeyAck.cnf~~ message arrives after the last attempt or an AKM_NewKey.cnf~~AKM_KeyAck.cnf~~ message ~~brings~~ is received with a rejection code (NACK).

4) After receiving confirmation messages from all the addressees or expiration of all attempts, ~~T~~the SC shall reply~~ies~~ to the supplicant with the ~~confirmation (~~AKM_KeyConfirmation.req~~)~~ message, which includes the generated pair of NN keys and DEVICE_ID(s) of the addressee(s) that acknowledged ~~reception~~ receipt of the AKM_NewKey.req message without a rejection code in the AKM_NewKey.cnf message. The AKM_KeyConfirmation.req message shall be encrypted using the NSC key of the supplicant.

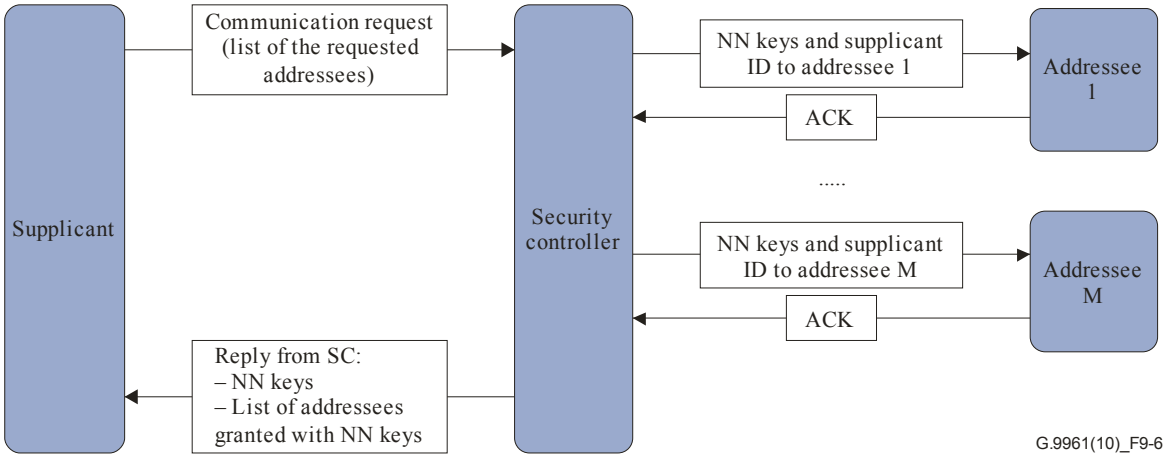


Figure 9-6 – Procedure for NN key generation for unicast (M=1) and multicast (M > 1)

Upon receiving the AKM_KeyConfirmation.req message, the supplicant shall send an AKM_NewKey.ind message to the addressee(s) indicating that new NN keys are established. The AKM_NewKey.ind message shall be sent encrypted using the new NN key.

If the supplicant does not receive the reply from the SC (AKM_KeyConfirmation.req message) ~~during~~ within 5~~1~~ seconds, it shall consider the procedure failed and may re-start it again at the first opportunity. The maximum number of attempts shall be four. After four unsuccessful attempts, the supplicant shall resign from the network (since it is improperly configured) using the resignation procedure defined in clause 8.6.1.1.3.

In case a supplicant intends to join an additional addressee to the existing multicast group, the following steps shall be taken:

1) The supplicant shall sends to the SC an AKM_AddClient.reqrequest (AKM_KeyAddRequest.req) message that includes the NN keys already established for the multicast group and the DEVICE_ID of the addressee node it intends to join. The message shall be encrypted with NSC of the supplicant.

2) Upon reception of the AKM_AddClient.req message, Tthe SC shallaccepts the request, checks whether the addressee is authenticated, and shall sends the NN keys supplied by the supplicant to the authenticated addressee using the AKM_NewKey.req message, encrypted using the NSC key of the addressee. The addressee shall acknowledge the AKM_NewKey.req message by sending an AKM_NewKey.cnfAKM_KeyAck.cnf message to the SC. In case no AKM_NewKey.cnf messageAKM_KeyAck.cnf is received from the addressee during the time period of 200 ms, the SC may shall retransmit the message up to four times, and shall remove the addressee from the list if no AKM_NewKey.cnf messageAKM_KeyAck.cnf arrives after the last attempt or if the AKM_NewKey.cnf messageAKM_KeyAck.cnf brings a rejection code (NACK).

3) After receiving the AKM_NewKey.cnf message, Tthe SC replies to the supplicant with the confirmation (AKM_KeyConfirmation.req) message that includes the pair of NN keys and DEVICE_ID of the addressee, if it acknowledged reception of the AKM_NewKey.req message (. iIf no addressee name is communicated in the AKM_KeyConfirmation.req message, the addressee is not joined added to the group). The AKM_KeyConfirmation.req message shall be encrypted using the NSC key of the supplicant.

4) Upon receiving the AKM_KeyConfirmation.req message, the supplicant shall transmit an AKM_NewKey.ind message (encrypted using the NN key) to the Addressee indicating that the NN key is confirmed.

61) Clause 9.2.4, Updating and termination of encryption keys

Revise the text of clause 9.2.4 "Updating and termination of encryption keys" as follows:

9.2.4 Updating and termination of encryption keys

From time to time the SC may shall initiate a routine update of encryption keys. The frequency of routine updates is vendor discretionary, although but the period interval of between the updates shall be much :

- lLonger than the duration of the procedure to establish the corresponding key30 minutes and not exceed 1 hour for NMK but shall not exceed 24 hours.
- Longer than 1 hour and not exceed 6 hours for NN/DB keys.

In addition, the key shall be updated to prevent repetition of FN for the same key (see clause 9.1.2.3). In case the SC suspects a security breach, it may update the security keys immediately.

A transmitting node shall not use an old key to encrypt APDUs that arrived at the A-interface, or LCDUs that were generated after the key was updated.

9.2.4.1 Updating of NSC and NN keys

When an SC determines that the NSC key should be updated (due to routine update), The key updating procedure shall be initiated by the SC. To initiate the procedure, the SCit shall send an key update request AKM_KeyUpdate.req message to the node. The AKM_KeyUpdate.req message shall indicate the NSC key update request and 'routine update' request reason (see Table 9-19). The node receiving the AKM_KeyUpdate.req message shall then initiate an authentication procedure with the SC, as described in clause 9.2.2. The 'Re-authentication flag' in the AUT_NodeAuthentication.req message used to initiate the authentication shall be set to 1, to signal that the request is for re-authentication (see Table 9-8).that initiated generation of the key(s) to be updated. The node receiving the AKM_KeyUpdate.req message shall reply to the SC by:

- ~~initiating an authentication procedure with the SC, as described in clause 9.2.2, if AKM_KeyUpdate.req message indicates NSC key update;~~
- ~~initiating a point-to-point key generation procedure with the relevant addressees, as described in clause 9.2.3, if AKM_KeyUpdate.req message indicates the NN keys update.~~

If the SC does not receive the reply from the requested node in a time period of 200 ms, it shall repeat the request. If after four attempts the node does not start the process to re-establish the key, the SC shall terminate the NSC key associated with this node, and initiate forced resignation of the node from the domain using the procedures described in clause 8.6.1.1.3.2 (by sending to the domain master the SC_DMRes.req message). The resigned node can further request to be admitted back using the standard admission procedure described in clause 8.6.1.

9.2.4.2 Updating of NN keys

When a node detects FN expiration for one of its point-to-point/point-to-multipoint keys, it shall initiate a point-to-point/point-to-multipoint key generation procedure to establish NN keys with the relevant addressees, as described in clause 9.2.3. The 'Request Reason' field in the AKM_KeyRequest.req message used to initiate the key generation shall be set to 'key update due to FN expiration' (see Table 9-12).

NOTE – Setting the 'Request Reason field to 'key update due to FN expiration' can help the SC to refrain from multiple key updates in a point-to-multipoint scenario when multiple nodes detect FN expiration at the same time and send multiple AKM_KeyRequest.req messages to the SC to update the same key.

When an SC detects that a point-to-point/point-to-multipoint key should be updated (due to routine update), it shall send a key update request AKM_KeyUpdate.req message to the node that initiated generation of the key(s) to be updated. The AKM_KeyUpdate.req message shall indicate NN keys update request and 'routine update' request reason (see Table 9-19). The node shall then initiate a point-to-point/point-to-multipoint key generation procedure as described in the previous paragraph.

9.2.4.3~~2~~ Termination of NSC and NN keys

The SC shall terminate all NSC keys associated with a node upon node resignation from the domain, as indicated in the TM_DomainRoutingChange.ind message. The node shall terminate NN keys if the node-supplicant for these keys resigns from the domain or its re-registration is unsuccessful. Old values of NSC and NN keys shall be terminated after the corresponding key update procedures.

The NSC and NN keys associated with a node shall not be terminated and are not required to be updated after a successful re-registration or re-authentication of the node.

The domain master may resign any node from the domain based on security considerations using the forced resignation procedure described in clause 8.6.1.1.3.2. The SC shall use the SC_DMRes.req message to request resignation of the node from the domain.

9.2.4.4 Updating of DB and NMK keys

Whenever the NMK key or DB key has expired, the SC shall update the DB keys and NMK, and communicate the updated keys to all authenticated nodes in the domain, by unicasting the AKM_DomainKeyUpdate.ind message. This message shall always be sent encrypted with the NSC key of the corresponding destination node. The SC shall send the request AKM_DomainKeyUpdate.req message to the domain master. The domain master shall confirm with confirmation AKM_DomainKeyUpdate.cnf message. If the SC does not receive the confirmation AKM_DomainKeyUpdate.cnf message within 800 ms it shall resend the AKM_DomainKeyUpdate.req message to the domain master.

Upon receiving the AKM_DomainKeyUpdate.req message, the domain master shall advertise that the updated NMK or DB key is going to take effect starting from the MAC cycle that is specified in the UpdateMacCycle field in the auxiliary NMK_DB_update sub-field in the MAP. A node that

concludes, according to the advertisement in the MAP, that it did not receive the updated key shall request the updated key from the SC by sending the request AKM_KeyUpdate.req message.

Any authenticated node can also detect FN expiration of the NMK or DB and request an updated set of DB or NMK by sending to the SC an AKM_KeyUpdate.req message. The 'Request Reason' field in the AKM_KeyRequest.req message used to initiate the key generation shall be set to 'key update due to FN expiration' (see Table 9-12). The SC shall reply with an AKM_DomainKeyUpdate.ind message with security mode set to 01 (NMK and DB keys) within 200 ms.

62) Clause 9.2.5, Messages supporting AKM procedures

Revise the text of clause 9.2.5 "Messages supporting AKM procedures" as follows:

9.2.5 Messages supporting AKM procedures

9.2.5.1 Authentication messages

9.2.5.1.1 ~~Authentication request message (~~Format of AUT_~~NodeRequest~~NodeAuthentication.req~~)~~

The AUT_~~NodeRequest~~NodeAuthentication.req message is a unicast management message intended to be used for authentication request only. The format of the MMPL of the AUT_~~NodeRequest~~NodeAuthentication.req message shall be as shown in Table 9-8.

Table 9-8 – Format of the MMPL of the AUT_~~NodeRequest~~NodeAuthentication.req message

Field	Octet	Bits	Description
Value of *X*	0 to 271	[2175:0]	Value of *X* as per [ITU-T X.1035].
Re-authentication flag	272	[0]	Shall be set to zero for first authentication request and to one if the request is for re-authentication.
Attempt number		[2:1]	Shall be set to 00_2 for the initial request and incremented for every next attempt.
ProxyAuth		[3]	Proxy authentication flag; shall be set to one for authentication through proxy and zero otherwise.
Reserved		[7:~~4~~3]	Reserved by ITU-T (Note 1).
ProxyDevID	273	[7:0]	Device ID of the authentication proxy (Note 2).
NOTE 1 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver. NOTE 2 – This field shall be set to zero by the transmitter and ignored by the receiver when the ProxyReg field is set to zero.			

9.2.5.1.2 ~~Authentication Prompt message (~~Format of AUT_Promp.ind~~)~~

The AUT_Promp.ind ~~_PRM~~ message is a unicast management message intended to be used for communication of the prompt computed by the Authenticator. The format of the MMPL of the AUT_Promp.ind ~~_PRM~~ message shall be as shown in Table 9-9.

Table 9-9 – Format of the MMPL of the AUT_Promp.ind message

Field	Octet	Bits	Description
Value S_1	0 to 15	[127:0]	Value of S_1 as per [ITU-T X.1035]
Value Y	16 to ~~159~~287	[~~1151~~2175:0]	Value of Y as per [ITU-T X.1035]
Status	~~160~~288	[0]	Shall be set to zero if the X-value was accepted ~~and~~ or one~~1~~ otherwise
Reserved		[7:1]	Reserved by ITU-T (Note)
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

9.2.5.1.3 ~~Authentication Prompt Verification message (~~Format of AUT_Verification.rspes~~)~~

The AUT_Verification.rspes message is a unicast management message is intended to communicate to the Authenticator the variables computed for prompt verification by the Supplicant. The format of the MMPL of the AUT_Verification.rspes message shall be as shown in Table 9-10.

Table 9-10 – Format of the MMPL of the AUT_Verification.rspes message

Field	Octet	Bits	Description
Value S_2	0 to 15	[127:0]	Value of S_2 as per [ITU-T X.1035]
Status	16	[0]	Shall be set to zero if both the S_1-value and Y-value were accepted ~~and~~ or one~~1~~ otherwise
Reserved		[7:1]	Reserved by ITU-T (Note)
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

9.2.5.1.4 ~~Authentication Confirmation message (~~Format of AUT_Confirmation.cnf~~)~~

The AUT_Confirmation.cnf message is a unicast management message intended to communicate confirmation of authentication from the Authenticator to the Supplicant, and grant the Supplicant the DB key and the NMK. The format of the MMPL of the AUT_Confirmation.cnf message shall be as shown in Table 9-11.

Table 9-11 – Format of the MMPL of the AUT_Confirmation.cnf message

Field	Octet	Bits	Description
Security mode	0	[1:0]	00_2 – point-to-point (Note 1) 01_2 – single key per domain (NMK) (Note 1) 10_2, 11_2 – reserved by ITU-T (Note 2)
Confirmation flag		[3:2]	~~Shall be set to 11_2 if authenticated and any other value for "authentication fails":~~ 00_2 – Success~~reason undefined~~ 01_2 – Failure~~reason (reserved)~~ 10_2, 11_2 – Reserved by ITU-T (Note 2)~~ reason (reserved)~~
DB Key Present		[4]	0 if DB Key is not present 1 if DB Key is present
NMK Key Present		[5]	0 if NMK Key is not present 1 if NMK Key is present

Table 9-11 – Format of the MMPL of the AUT_Confirmation.cnf message

Field	Octet	Bits	Description
DB Key ID		[6]	The current DB key ID to use for encryption. This field shall be ignored if DB Key Present is set to 0.
NMK Key ID		[7]	The current NMK key ID to use for encryption. This field shall be ignored if NMK Key Present is set to 0.
~~Reserved~~		~~[7:4]~~	~~Reserved by ITU-T (NOTE)~~
DB0 key	~~1 to 16~~variable	[127:0]	Encryption key for broadcast communications with keyId 0. This field only exists if DB Key Present is set to 1
DB1 key	variable	[127:0]	Encryption key for broadcast communications with keyId 1. This field only exists if DB Key Present is set to 1
NMK0	variable	[127:0]	NMK with keyID 0, if security mode is set to 01_2. This field shall be skipped if security mode is set to 00_2. This field only exists if NMK Present is set to 1
NMK1	~~17to 32~~variable	[127:0]	NMK with keyID 1, if security mode is set to 01_2. This field shall ~~not~~ be ~~present~~ skipped if security mode is set to 00_2. This field only exists if NMK Present is set to 1
NOTE 1 – For the first key exchange with the security controller, if the security mode is a point-to-point, DB Key Present shall be set to one, and NMK Key Present shall be set to zero; and if the security mode is a single key per domain, both DB Key Present and NMK Key Present shall be set to one. NOTE 2 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

9.2.5.1.5 Format of AUT_NodeAuthenticated.req

The AUT_NodeAuthenticated.req message is a unicast management message intended to inform the DM that the specified authenticating node is authenticated by the SC and the DM shall include it in the routing table. The format of the MMPL of the AUT_NodeAuthenticated.req message shall be as shown in Table 9-11bis~~2~~.

Table 9-11bis~~2~~ – Format of the MMPL of the AUT_NodeAuthenticated.req message

Field	Octet	Bits	Description
SC	0	[7:0]	DEVICE ID of the SC
AUTH	1	[7:0]	DEVICE ID of the authenticated node
Proxy	2	[7:0]	DEVICE ID of the proxy node

9.2.5.1.6 Format of AUT_NodeAuthenticated.cnf

The AUT_NodeAuthenticated.cnf message is a unicast management message intended to inform the SC that the DM received successfully the AUT_NodeAuthenticated.req message. The format of the MMPL of the AUT_NodeAuthenticated.cnf message shall be as shown in Table 9-11ter~~3~~.

Table 9-11ter~~3~~ – Format of the MMPL of the AUT_NodeAuthenticated.cnf message

Field	Octet	Bits	Description
AUTH	0	[7:0]	DEVICE ID of the authenticated node
Proxy	1	[7:0]	DEVICE ID of the proxy node

63) Clause 9.2.5.2, Pair-wise authentication messages

Revise the text of clause 9.2.5.2 "Pair-wise authentication messages" as follows:

9.2.5.2 Pair-wise authentication messages

9.2.5.2.1 ~~Communication request message (~~Format of AKM_KeyRequest.req~~)~~

The AKM_KeyRequest.req message is a unicast management message intended to be used for communication request by the supplicant only. It is limited to 248 addressees. The format of the MMPL of the AKM_KeyRequest.req message shall be as shown in Table 9-12.

Table 9-12 – Format of the MMPL of the AKM_KeyRequest.req message

Field	Octet	Bits	Description
Number of Addressees	0	[7:0]	Number of addressees N (1 for unicast transmission and up to 248 for multicast transmission).
Multicast stream identifier	1	[7:0]	Shall be set to the multicast stream identifier (MSID) for multicast keys. Otherwise it shall be set to 0.
Addressee name	2~~1~~	[7:0]	First addressee unicast DEVICE_ID.
Addressee name	3~~2~~	[7:0]	Second addressee unicast DEVICE_ID.
…	…	…	…
Addressee name	N+1	[7:0]	N-th addressee unicast DEVICE_ID.
Attempt number	N+2~~1~~	[1:0]	Shall be set to 00_2 for the initial request and incremented for every next attempt.
KeyID		[2]	Set to zero to request key~~s~~ with ID = 0~~,~~ ~~1~~ ~~and~~ or set to one to request key~~s~~ with ID = 1~~2~~ ~~3~~.
Request Reason		[4:3]	00 for first key generation, 01 for key update due to FN expiration, 10-11 are reserved by ITU-T
Reserved		[7:5~~3~~]	Reserved by ITU-T (Note).
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

9.2.5.2.1.1 ~~Add-a-node Request message (~~Format of AKM_~~Key~~Add~~Client~~Request~~.req~~)~~

The AKM_AddClient.req~~AKM_KeyAddRequest.req~~ message is a unicast management message intended to be used for joining a node to a multicast group originated by the supplicant only. It is limited to one addressee. The MMPL of the AKM_AddClient.req~~AKM_KeyAddRequest.req~~ message ~~shall be as~~is presented in Table 9-13.

Table 9-13 – Format of the MMPL of the AKM_AddClient.req~~AKM_KeyAddRequest.req~~ message ~~format~~

Field	Octet	Bits	Description
Addressee name	0	[7:0]	The addressee unicast DEVICE_ID.
Multicast stream identifier	1	[7:0]	Shall be set to the multicast stream identifier (MSID) for multicast keys.
NN_{SA} key 0/1~~2~~	~~1~~2 to 1~~76~~	[127:0]	Encryption key for Supplicant-to-Addressee direction with ID=0 if KeyID=0 and with ID=1~~2~~ if KeyID=1
~~NN_{AS} key 0/2~~	~~17 to 32~~	~~[127:0]~~	~~Encryption key for Addressee-to-Supplicant direction with ID=0 if KeyID=0 and with ID=2 if KeyID=1~~
~~NN_{SA} key 1/3~~	~~33 to 48~~	~~[127:0]~~	~~Encryption key for supplicant-to-addressee direction with ID=1 if KeyID=0 and with ID=3 if KeyID=1~~
~~NN_{AS} key 1/3~~	~~49 to 64~~	~~[127:0]~~	~~Encryption key for addressee-to-supplicant direction with ID=1 if KeyID=0 and with ID=3 if KeyID=1~~
Attempt number	~~65~~18	[1:0]	Shall be set to 00_2 for the initial request and incremented for every next attempt.
KeyID		[2]	Set to zero for key with ID = 0 and set to one for key with ID = 1
Reserved		[7:3~~2~~]	Reserved by ITU-T (Note).
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

9.2.5.2.2 ~~Key communication message (~~Format of AKM_NewKey.req~~)~~

The AKM_NewKey.req message is a unicast management message intended to be used for communication of the NN key from the SC to the addressee only. The format of the MMPL of the AKM_NewKey.req message shall be as shown in Table 9-14.

Table 9-14 – Format of the MMPL of the AKM_NewKey.req message

Field	Octet	Bits	Description
S~~s~~upplicant name	0	[7:0]	Supplicant's unicast DEVICE_ID.
Number of keys	1	[1:0]	Number of keys provided by the SC represented as an unsigned integer minus 1.
Reserved		[7:2]	Reserved by ITU-T (Note).
Multicast stream identifier	2	[7:0]	Shall be set to the multicast stream identifier (MSID) for multicast keys. Otherwise it shall be set to 0
NN_{SA} key 0/1~~2~~	~~2~~3 to 1~~87~~	[127:0]	Encryption key for supplicant-to-addressee direction with ID=0 if KeyID=0 and with ID=1~~2~~ if KeyID=1
NN_{AS} key 0/1~~2~~	1~~98~~ to 3~~43~~	[127:0]	Encryption key for addressee-to-supplicant direction with ID=0 if KeyID=0 and with ID=1~~2~~ if KeyID=1
~~NN_{SA} key 1/3~~	~~34 to 49~~	~~[127:0]~~	~~Encryption key for supplicant-to-addressee direction with ID=1 if KeyID=0 and with ID=3 if KeyID=1~~
~~NN_{AS} key 1/3~~	~~50 to 65~~	~~[127:0]~~	~~Encryption key for addressee-to-supplicant direction with ID=1 if KeyID=0 and with ID=3 if KeyID=1~~
KeyID	35	[0]	Set to zero for key with ID = 0 and set to one for key with ID = 1
Reserved		[7:1]	Reserved by ITU-T (Note)

Table 9-14 – Format of the MMPL of the AKM_NewKey.req message

Field	Octet	Bits	Description
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

9.2.5.2.3 ~~Communication acknowledgement message (~~Format of AKM_New~~Key~~~~Ack~~.cnf~~)~~

The AKM_NewKey.cnf~~COM _ACK~~ message is a unicast management message intended to be used to confirm delivery of the new encryption key to the SC or to reject the communication request. The format of the MMPL of the AKM_NewKey.cnf~~AKM_KeyAck.cnf~~ message shall be as shown in Table 9-15.

Table 9-15 – Format of the MMPL of the AKM_NewKey.cnf~~AKM_KeyAck.cnf~~ message

Field	Octet	Bits	Description
Supplicant	0	[7:0]	Device ID of the supplicant associated with this key
Multicast Stream Identifier	1	[7:0]	Shall be set to the multicast stream identifier (MSID) for multicast keys. Otherwise it shall be set to 0
ACK	2~~0~~	[1:0]	00 – If the addressee successfully ~~decoded~~ received the new encryption key ~~01 – f the ddressee is incapable to decoded the new encryption key.~~ 01~~0~~ – If the addressee ~~decoded~~ successfully received the new encryption key, but denies communication with supplicant (NACK) 10, 11 – Reserved by ITU-T
Reserved		[7:2]	Reserved by ITU-T (Note)
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

9.2.5.2.4 ~~Confirmation message (~~Format of AKM_KeyConfirmation.req~~)~~

The AKM_KeyConfirmation.req message is a unicast management message intended to communicate the NN key with the actual list of addressees or the NMK from the SC to the supplicant only. The format of the MMPL of the AKM_KeyConfirmation.req message ~~shall be as~~is shown in Table 9-16.

Table 9-16 – Format of the MMPL of the AKM_KeyConfirmation.req message

Field	Octet	Bits	Description
Security mode	0	[1:0]	00 – Point-to-Point. 01 – Single key per domain (NMK). 10~~,~~ – Update of the DB keys. 11 – Reserved by ITU-T.
KeyID		[2]	Set to zero for key with ID = 0 ~~and~~or set to one for key with ID = 1
Reserved		[7:3~~2~~]	Reserved by ITU-T (Note 1).
DB 0/1	1 to 16	[127:0]	DB key with ID=0 if KeyID=0 ~~and~~or with ID=1 if KeyID=1.
NMK 0/1~~2~~	17 to 32~~16~~	[127:0]	NMK with ID=0 if KeyID=0 ~~and~~ or with ID=1~~2~~ if KeyID=1, if security mode is 01. This field shall be

Table 9-16 – Format of the MMPL of the AKM_KeyConfirmation.req message

Field	Octet	Bits	Description
			skipped if security mode is 00. All of the fields describing NN keys shall be skipped if security mode is 01.
NN_{SA} key 0/1~~2~~	~~17~~33 to 48~~32~~	[127:0]	Encryption key for supplicant-to-addressee direction with ID=0 if KeyID=0 ~~and~~ or with ID=1~~2~~ if KeyID=1
NN_{AS} key 0/1~~2~~	~~33~~49 to 64~~48~~	[127:0]	Encryption key for addressee-to-supplicant direction with ID=0 if KeyID=0 ~~and~~ or with ID=1~~2~~ if KeyID=1
~~NN_{SA} key 1/3~~	~~49 to 64~~	~~[127:0]~~	~~Encryption key for supplicant-to-addressee direction with ID=1 if KeyID=0 and with ID=3 if KeyID=1~~
~~NN_{AS} key 1/3~~	~~65 to 80~~	~~[127:0]~~	~~Encryption key for addressee-to-supplicant direction with ID=1 if KeyID=0 and with ID=3 if KeyID=1~~
Number of addressees	65~~81~~	[7:0]	Number of addressees N (1 for unicast transmission and up to 248 for multi-cast transmission) (Note 2).
Addressee name	66~~82~~	[7:0]	First addressee unicast DEVICE_ID.
Addressee name	67~~83~~	[7:0]	Second addressee unicast DEVICE_ID.
…	…	…	…
Addressee name	~~81~~65 + N	[7:0]	N-th addressee unicast DEVICE_ID.
NOTE 1 – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver. NOTE 2 – I~~n case~~f no addressee is authenticated, the list is empty and the field shall be set to zero.			

9.2.5.2.5 ~~Resignation request message (~~Format of SC_DMRes.req~~)~~

The SC_DMRes.req message is a unicast management message sent by the SC to the domain master and is intended to inform the domain master that a particular node(s) has to be forced out of the domain due to authentication failure. This message is invalid if the SC and the domain master functions are performed by the same node. The MMPL of the SC_DMRes.req message shall be as presented in Table 9-17.

Table 9-17 – Format of the MMPL of the SC_DMRes.req message ~~format~~

Field	Octet	Bits	Description
Number entries	0	[7:0]	Indicates the number of nodes (n) in the following list, represented as an unsigned integer.
Entry 1	1	[7:0]	DEVICE_ID of the first node that is requested to be expelled from the domain.
…	…	…	…
Entry n	n	[7:0]	DEVICE_ID of the last node that is requested to be expelled from the domain.

9.2.5.2.6 ~~Confirmation of resignation message (~~Format of SC_DMRes.cnf~~)~~

The SC_DMRes.cnf message is a unicast management message sent by the domain master to the SC to confirm the receipt of the SC_DMRes.req message from~~resignation of the nodes requested by~~ the SC. The MMPL of the SC_DMRes.cnf message shall be as presented in Table 9-18.

Table 9-18 – Format of the MMPL of the SC_DMRes.cnf message ~~format~~

Field	Octet	Bits	Description
Number entries	0	[7:0]	Indicates the number of nodes (n) in the following list, represented as an unsigned integer.
Entry 1	1	[7:0]	DEVICE_ID of the first node ~~requested to be~~ expelled from the domain.
	~~2~~	~~[7:0]~~	~~Status code~~
…	…	…	…
Entry n	~~(2×n) 1~~	[7:0]	DEVICE_ID of the last node requested to be expelled from the domain.
	~~(2×n)~~	~~[7:0]~~	~~Status code~~

9.2.5.2.7 Format of AKM_NewKey.ind

The AKM_NewKey.ind message is a unicast management message that shall only be used to inform the addressee that the supplicant received the NN key and communication using this new NN key is available. The MMPL of the AKM_NewKey.ind message shall be empty.

64) Clause 9.2.5.3, Key updating messages

Revise the text of clause 9.2.5.3 "Key updating messages" as follows:

9.2.5.3 Key updating messages

9.2.5.3.1 ~~Re-authentication and key update request (~~Format of AKM_KeyUpdate.req~~)~~

The AKM_KeyUpdate.req message is a unicast management message intended to be used for node re-authentication and update of the:

- NSC key, or ~~and~~
- NN keys or NMK, or
- DB key ~~only~~.

The format of the MMPL of the AKM_KeyUpdate.req message shall be as shown in Table 9-19.

Table 9-19 – Format of the MMPL of the AKM_KeyUpdate.req message

Field	Octet	Bits	Description
Supplicant	0	[7:0]	Device ID of the supplicant associated with this key. This field shall be set to FF_{16} if NSC, DB or NMK is updated.
Multicast Stream Identifier	1	[7:0]	Shall be set to the multicast stream identifier (MSID) for multicast keys. Otherwise it shall be set to 00_{16}
Type of the key	2~~0~~	[1:0]	00 for NSC, 01 for NN or for NMK. 10 for DB, 11 ~~are~~ is reserved by ITU-T
KeyID		[2]	Set to 0 to request keys with ID = 0 or set to 1 to request keys with ID = 1
Request reason		[4:3]	00 for FN expiration, 01 for routine update, 10-11 are reserved by ITU-T
Reserved		[7:5~~2~~]	Reserved by ITU-T (Note)

Table 9-19 – Format of the MMPL of the AKM_KeyUpdate.req message

Field	Octet	Bits	Description
Authenticator	~~3~~1	[7:0]	This field shall be set to the DEVICE_ID of the node requesting the key update.~~Shall be set to FF_{16} if NSC is to be updated and set to DEVICE_ID of one of the addressees if NN is to be updated (both for unicast and multicast).~~
Attempt number	~~4~~2 and ~~5~~3	[1:0]	Shall be set to 00_2 for the initial request and incremented for every next attempt
Reserved		[3:2]	Reserved by ITU-T (Note)
Last update		[15:4]	Indicates time from the last successful update in minutes. Special value FFF_{16} indicates any period longer than 4095 minutes
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

9.2.5.3.2 Format of AKM_DomainKeyUpdate.ind

The AKM_DomainKeyUpdate.ind message is a unicast management message intended to be used for indicating the update of either the DB key or the NMK. The format of the MMPL of the AKM_DomainKeyUpdate.ind message shall be as shown in Table 9-20.

Table 9-20 – Format of the MMPL of the AKM_DomainKeyUpdate.ind message

Field	Octet	Bits	Description
Key Type	0	[0]	0 if DB Key is present 1 if NMK Key is present
Key ID		[1]	The key ID of the updated key.
Reserved		[7:2]	Reserved by ITU-T (Note)
Transaction_ID	1	[7:0]	Transaction identification for this key update. For each update of NMK or DB key. The Transaction_ID value is incremented by one until 255 and then wraparound to 0.
DB NMK_key	2 to ~~-~~17	[127:0]	Encryption key of the DB or NMK key according to the value set to Key_Type field.
NOTE – Bits that are reserved by ITU-T shall be set to zero by the transmitter and ignored by the receiver.			

9.2.5.3.3 Format of AKM_DomainKeyUpdate.req message

The AKM_DomainKeyUpdate.req message is a unicast management message the SC sends to the domain master to update the NMK key or the DB key. The format of the MMPL of the AKM_DomainKeyUpdate.req message shall be as shown in Table 9-20.

9.2.5.3.4 Format of AKM_DomainKeyUpdate.cnf message

The DM shall confirm receiving the AKM_DomainKeyUpdate.req by sending to the SC the AKM_DomainKeyUpdate.cnf message. The format of the MMPL of the AKM_DomainKeyUpdate.cnf message shall be as shown in Table 9-21.

Table 9-21 – AKM_DomainKeyUpdate.cnf message format

Field	Octet	Bits	Description
SC	0	[7:0]	Device ID of the SC node that sent the AKM_DomainKeyUpdate.req message.
DM	1	[7:0]	Device ID of the Domain Master that sends this message.
Transaction_ID	2	[7:0]	The transaction identification that was specified in the confirmed AKM_DomainKeyUpdate.req message.

65) Clause A.1.1, Frame conversion

Revise the text of clause A.1.1 "Frame conversion" as follows:

A.1.1 Frame conversion

The incoming set of primitives (AIF_DATA.REQ) and the outgoing set of primitives (AIF_DATA.IND) at the A-interface of EAPC represent a sequence of Ethernet frames, each defined as a set of IEEE 802.1 primitives of M_UNITDATA.request and M_UNITDATA.indication, respectively, Table A.1.

Table A.1 – A-interface primitives description

AIF_DATA.REQ (AE → EAPC)	AIF_DATA.IND (EAPC → AE)
M_UNITDATA.request (frame_type, destination_address, source_address, mac_service_data_unit, user_priority, access_priority, frame_check_sequence)	M_UNITDATA.indication (frame_type, destination_address, source_address, mac_service_data_unit, user_priority, frame_check_sequence)

All unit-signal primitives specified in Table A.1 shall be interpreted in terms of clause 6.4 of [IEEE 802.1D]. Note that primitives frame_type, user_priority, access_priority, and frame_check_sequence, may not be provided by the AE, and primitives frame_type and user_priority may not be requested by the AE.

NOTE 1 – Clause 6.5.1 of [IEEE 802.1D] suggests that the "access priority" primitive be ignored and the frame_type primitive be set to user_data_type for 802.3 MAC frames.

NOTE 2 – The M_UNITDATA.request description in [IEEE 802.1Q] differs from that in [IEEE 802.1D] as it omits the frame_type and access_priority parameters. The frame_type is not required in [IEEE 802.1Q] as the receipt of a frame other than a user data frame does not cause a data indication, nor are such frames transmitted by the medium independent bridge functions. The mapping of M_UNITDATA.request to particular access methods specified in [IEEE 802.1Q] includes derivation of the access_priority parameter (for those media that require it) from the user_priority parameter.

NOTE 3 – The EM_UNITDATA.request and EM_UNITDATA.indication description in [IEEE 802.1ad] includes more QoS related primitives, such as drop_eligible and others. These primitives, similarly to those defined in clause 6.6.1 of [IEEE 802.1Q] should be accommodated in the corresponding tags fields of the APDU as described in Table A.1.

If the frame_check_sequence primitive is provided by the AE, the incoming M_UNITDATA.request primitives (described in Table A.1 for AIF_DATA.REQ) shall be verified to be error free by computing their FCS as defined in clause 6.5.1 of [IEEE 802.1D]. If the computed FCS does not match the received value of the frame_check_sequence, the incoming primitive shall be discarded. If the frame_check_sequence primitive is not provided by the AE, the APC shall compute the FCS of the incoming M_UNITDATA.request primitives as defined in clause 3.28 of [IEEE 802.3].

Error-free primitives described in Table A.1 for AIF_DATA.REQ shall be converted into the APDU format presented in Figure A.1. The same APDU format shall be used for in-band management messages sourced by the local DLL management entity for the remote AE.

	~~LSB MSB~~
6 octets	Destination address
6 octets	Source address
~~0~~ TG octets	~~EtherType/~~VLAN TAGs
2 octets	MAC client length/type
Application dependent	Service data unit (APDU payload)
4 octets	Frame check sequence (FCS)

Figure A.1 – APDU format (TX and RX)

All fields shall have the same content as the corresponding fields of the MAC frame defined in [IEEE 802.3], including various embedded tags mapped into the VLAN TAGs field. Mapping of the unit-data primitives, including embedded tags, into all these APDU fields shall comply with the [IEEE 802.3] or relevant IEEE bridging standard, such as [IEEE 802.1D], [IEEE 802.1Q], etc. The VLAN TAGs field shall only be present (i.e., TG > 0) for:

- Single-tagged MAC frames according to [IEEE 802.1Q] (8100_{16}, VLAN-tagged frames, TG = 4) or
- Single-tagged MAC frames according to [IEEE 802.1ad] ($88A8_{16}$, provider bridging, TG = 4) or
- Double-tagged MAC frames according to [IEEE 802.1ad] ($88A8_{16}$ for the 4-byte outer tag, followed by 8100_{16} for the 4-byte inner tag, TG = 8).

NOTE 4– [IEEE 802.1ad] is an amendment to [IEEE 802.1Q].

NOTE 5 – Usage of tags 9100_{16}, 9200_{16} and 9300_{16} has been deprecated by IEEE.

Otherwise, TG shall be set to zero, and the 2 octets after the source address are considered as MAC client length/type field. If AE provides neither frame_type, nor access_priority or user_priority primitives, the ~~EtherType/~~VLAN TAGs field of the APDU shall be zero octets long.

The unencrypted part of the APDU shall include all bytes starting from the first byte of the APDU and ending at the last byte of the "MAC client length/type" field of the APDU. The length of the unencrypted part of the ~~M~~APDU depends on the length *TG* of the ~~EtherType/~~VLAN TAGs field of the APDU (see clause 9.1.2.2).

The FCS of APDU shall be used only if MIC is not used as a part of the encryption scheme (see clause 9.1.1); otherwise, the FCS shall be stripped off and not communicated through the domain.

NOTE ~~4~~ 6 – Since the FCS is stripped off and reconstructed by the remote APC in the case MIC is included, verification of the incoming M_UNITDATA.request primitives to be error free is essential in order to avoid the creation and propagation of frames with undetectable errors.

Bits of APDU shall be transmitted starting from the first octet of the destination address. The ~~LSB~~ least significant bit of each octet shall be transmitted first. The most significant octet of each field shall be transmitted first.

The order of outgoing APDUs at the x1 reference point associated with a particular destination and particular user priority shall be the same as the order of incoming unit-data of these same user priority and destination. No re-ordering inside the same user priority group for the same destination is allowed.

The M_UNITDATA.indication primitives shall be derived from the APDUs received from the LLC across the x1 reference point as defined in clause 6.4.1 of [IEEE 802.1D], with the following additional rules:

– The user_priority primitive shall be derived from the TAGs field for all embedded tags as defined in clauses 6.6.1 and 9 of [IEEE 802.1Q]; if TAGs field is of zero length, the user_priority primitive shall be set to zero.
– The frame_check_sequence primitive, if FCS is not a part of APDU, shall be computed as defined in clause 3.28 of [IEEE 802.3].
– The frame_check_sequence primitive, if FCS is a part of APDU, shall be verified as defined in clause 6.5.1 of [IEEE 802.1D]. APDUs that did not pass verification shall be discarded.

The same rules shall also be used to derive the M_UNITDATA.indication primitives for the in-band management messages sourced by the DLL management entity for the local AE.

In-band management data units generated by the DLL management entity shall follow the LCDU format defined in clause 8.1.3.4.

66) New Annex V, Versioning dependencies of ITU-T G.9961

Add new Annex V "Versioning dependencies of ITU-T G.9961" as follows:

Annex V

Versioning dependencies of ITU-T G.9961

(This annex forms an integral part of this Recommendation.)

For details on the versioning mechanism, see clause 8.19.

The versioning dependencies between this Recommendation and other Recommendations of the ITU-T G.996x family is described in Table V.1. The number indicated in the following table represents the minimum amendment that is compatible with the Recommendation described in this document.

Table V.1 – Versioning dependencies of ITU G.9961

ITU-T G.9960	ITU-T G.9961	ITU-T G.9962	ITU-T G.9963	ITU-T G.9964
0	N/A	X	X	0

NOTE – The following values apply to this table:

- A value of 0 indicates the base document of a Recommendation.
- A value of X indicates that this Recommendation is not dependent on the indicated Recommendation.
- A value of N/A indicates this Recommendation.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	**Transmission systems and media, digital systems and networks**
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Terminals and subjective and objective assessment methods
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects and next-generation networks
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2014