

**Draft Amendment to STANDARD [for]
Information Technology - Telecommunications and
Information Exchange Between Systems -
LAN/MAN Specific Requirements -**

**Part 11: Wireless Medium Access Control (MAC)
and Physical Layer (PHY) specifications:**

**Medium Access Control (MAC) Quality of Service
(QoS) Enhancements**

Sponsored by the
IEEE 802 Committee
of the
IEEE Computer Society

Copyright © 2003 by the Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street
New York, NY 10017, USA
All rights reserved.

This is an unapproved draft of a proposed IEEE Standard, subject to change. Permission is hereby granted for IEEE Standards Committee participants to reproduce this document for purposes of IEEE standardization activities. If this document is to be submitted to ISO or IEC, notification shall be given to the IEEE Copyright Administrator. Permission is also granted for member bodies and technical committees of ISO and IEC to reproduce this document for purposes of developing a national position. Other entities seeking permission to reproduce this document for standardization or other activities, or to reproduce portions of this document for these or other uses, must contact the IEEE Standards Department for the appropriate license. Use of information contained in this unapproved draft is at your own risk.

IEEE Standards Department
Copyright and Permissions
445 Hoes Lane, P.O. Box 1331
Piscataway, NJ 08855-1331, USA

Introduction

(This introduction is not part of IEEE P802.11e, Draft Amendment to STANDARD [for] Information Technology-Telecommunications and Information Exchange Between Systems -LAN/MAN Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: Medium Access Control (MAC) Quality of Service (QoS) Enhancements)

At the time this standard was submitted to Sponsor Ballot, the working group had the following membership:

Stuart J. Kerry, *Chair*

Al Petrick and Harry Worstell, *Vice-chairs*

Tim Godfrey, *Secretary*

John Fakatselis, *Tge Chair*

Duncan Kitchin, *Tge Vice-chair*

David Hunter and Tim Godfrey, *Tge Secretaries*

Srinivas Kandala and Michael Fischer, *Tge Editors*

<insert full membership list here>

Major contributions were received from the following individuals:

<insert the names of the contributors>

The following persons were on the balloting committee:
(To be provided by IEEE editor at time of publication.)

1 Contents

2	Introduction.....	ii
3	Contents	iii
4	1. Overview.....	1
5	1.2 Purpose	1
6	2. Normative references.....	1
7	3. Definitions	2
8	4. Abbreviations and acronyms	5
9	5. General description.....	6
10	5.1 General description of the architecture.....	6
11	5.1.1 How wireless LAN systems are different	6
12	5.1.1.2 ~~The m~~ Media impact ~~the on~~ design and performance	6
13	5.1.1.4 Interaction with other IEEE 802 [®] layers.....	6
14	5.2 Components of the IEEE 802.11 [®] architecture	6
15	5.2.5 QBSS: The quality of service network.....	6
16	5.3 Logical service interfaces.....	7
17	5.3.1 Station service (SS).....	7
18	5.3.2 Distribution system service (DSS).....	7
19	5.4 Overview of the services	8
20	5.4.1 Distribution of messages within a DS.....	8
21	5.4.1.3 QoS traffic scheduling.....	8
22	5.4.2 Services that support the distribution service	8
23	5.4.2.2 Association.....	8
24	5.4.4 Traffic differentiation and QoS support.....	8
25	5.4.5 Support for higher-layer timer synchronization.....	8
26	5.5 Relationships between services	9
27	5.6 Differences between ESS and IBSS LANs	9
28	5.7 Message information contents that support the services.....	9
29	5.7.1 Data	9
30	5.7.2 Association.....	10
31	5.7.3 Reassociation.....	10
32	6. MAC service definition	11
33	6.1 Overview of MAC services	11
34	6.1.1 ~~Asynchronous d~~ Data service.....	11
35	6.1.1.1 Determination of UP	11
36	6.1.1.2 Interpretation of priority parameter in MAC service primitives	12
37	6.1.1.3 Access Categories	12
38	6.1.1.4 Interpretation of service class parameter in MAC service primitives in a	
39	QSTA	13
40	6.1.3 MSDU ordering.....	13
41	6.1.4 MAC Data service architecture (Informative)	14
42	6.2 Detailed service specification.....	15
43	6.2.1 MAC data services	15
44	6.2.1.1 MA-UNITDATA.request.....	15
45	6.2.1.2 MA-UNITDATA.indication	16

1	6.2.1.3 MA-UNITDATA-STATUS.indication	16
2	7. Frame formats.....	18
3	7.1 MAC frame formats	18
4	7.1.1 Conventions	18
5	7.1.2 General frame format.....	19
6	7.1.3 Frame fields	19
7	7.1.3.1 Frame Control field	19
8	7.1.3.2 Duration/ID field	22
9	7.1.3.4 Sequence Control field	22
10	7.1.3.5 QoS Control field	23
11	7.1.3.6 Frame Body field.....	26
12	7.1.3.7 FCS field.....	26
13	7.1.4 Duration/ID field in Data and Management frames	26
14	7.2 Format of individual frame types.....	27
15	7.2.1 Control frames	27
16	7.2.1.1 Request To Send (RTS) frame format.....	27
17	7.2.1.2 Clear To Send (CTS) frame format	27
18	7.2.1.3 Acknowledgement (ACK) frame format.....	28
19	7.2.1.7 Block Acknowledgement Request (BlockAckReq) frame format	28
20	7.2.1.8 Block Acknowledgement (BlockAck) frame format.....	29
21	7.2.2 Data frames	30
22	7.2.3 Management frames.....	31
23	7.2.3.1 Beacon frame format	33
24	7.2.3.4 Association Request frame format	33
25	7.2.3.5 Association Response frame format	33
26	7.2.3.6 Reassociation Request frame format	34
27	7.2.3.7 Reassociation Response frame format.....	34
28	7.2.3.9 Probe Response frame format	34
29	7.3 Management frame body components	35
30	7.3.1 Fixed fields	35
31	7.3.1.4 Capability Information field.....	35
32	7.3.1.7 Reason Code field.....	38
33	7.3.1.9 Status Code field.....	39
34	7.3.1.11 Action field.....	39
35	7.3.1.12 Dialog Token field.....	39
36	7.3.1.13 DLP Timeout Value	40
37	7.3.1.14 Block Ack Parameter Set field	40
38	7.3.1.15 Block Ack Timeout Value.....	40
39	7.3.1.16 DELBA Parameter Set field	41
40	7.3.2 Information elements	41
41	7.3.2.13 QBSS Load element	42
42	7.3.2.14 EDCA Parameter Set element	42
43	7.3.2.15 Traffic Specification (TSPEC) element.....	44
44	7.3.2.16 Traffic Classification (TCLAS) Element	48
45	7.3.2.17 TS Delay	50
46	7.3.2.18 TCLAS Processing	50

1	7.3.2.19 Schedule Element.....	51
2	7.3.2.20 QoS Capability element.....	51
3	7.3.2.21 QoS Action Element	52
4	7.4 Action frame format details.....	52
5	7.4.2 QoS action frame details	52
6	7.4.2.1 ADDTS request frame format.....	53
7	7.4.2.2 ADDTS response frame format	53
8	7.4.2.3 DELTS frame format	54
9	7.4.2.4 Schedule frame body format.....	54
10	7.4.3 Direct Link Protocol (DLP) action frame details.....	55
11	7.4.3.1 DLP request	55
12	7.4.3.2 DLP response	56
13	7.4.3.3 DLP teardown	56
14	7.4.4 Block Ack action frame details	57
15	7.4.4.1 ADDBA request.....	57
16	7.4.4.2 ADDBA response	57
17	7.4.4.3 DELBA	58
18	7.5 Frame usage.....	59
19	9. MAC sublayer functional description.....	61
20	9.1 MAC architecture	61
21	9.1.3 Hybrid coordination function (HCF)	62
22	9.1.3.1 HCF contention-based channel access (EDCA).....	62
23	9.1.3.2 HCF controlled channel access (HCCA).....	63
24	9.1.4 Coexistence of DCF, ~~and PCF~~ and HCF.....	64
25	9.1.5 Fragmentation/defragmentation overview	64
26	9.1.6 MAC Data Service	65
27	9.2 DCF	65
28	9.2.3 Interframe space (IFS).....	65
29	9.2.3.4 Arbitration IFS (AIFS).....	66
30	9.2.3.5 EIFS	66
31	9.2.9 Duplicate detection and recovery.....	67
32	9.6 Multirate support	67
33	9.7 MSDU transmission requirements	68
34	9.8 Operation across regulatory domains	69
35	9.9 HCF	69
36	9.9.1 HCF contention-based channel access (EDCA)	69
37	9.9.1.1 Reference Implementation (Informative)	69
38	9.9.1.2 EDCA TXOPs.....	70
39	9.9.1.3 Obtaining an EDCA TXOP	70
40	9.9.1.4 Obtaining a Continuation of EDCA TXOP	72
41	9.9.1.5 EDCA Backoff Procedure.....	73
42	9.9.1.6 Retransmit Procedures	74
43	9.9.2 HCF controlled channel access	74
44	9.9.2.1 HCF controlled channel access procedure.....	75
45	9.9.2.2 TXOP structure and timing.....	77
46	9.9.2.3 HCF controlled channel access transfer rules.....	79

1	9.9.3 Admission Control at the HC.....	80
2	9.9.3.1 Contention-based Admission Control Procedures	81
3	9.9.3.2 Controlled-access Admission Control.....	82
4	9.10 Block acknowledgment.....	84
5	9.10.1 Introduction (Informative).....	84
6	9.10.2 Set up and modification of the Block Ack parameters	85
7	9.10.3 Data and acknowledgement transfer.....	85
8	9.10.4 Receive buffer operation.....	88
9	9.10.5 Tear down of the Block Ack mechanism.....	88
10	9.10.6 Error recovery upon a peer failure.....	88
11	9.11 No acknowledgement.....	88
12	9.12 Frame exchange sequences	89
13	10. Layer management	98
14	10.3 MLME SAP Interface	98
15	10.3.2 Scan.....	98
16	10.3.2.2 MLME-SCAN.confirm	98
17	10.3.6 Associate.....	98
18	10.3.6.1 MLME-ASSOCIATE.request	98
19	10.3.6.2 MLME-ASSOCIATE.confirm	99
20	10.3.6.3 MLME-ASSOCIATE.indication.....	99
21	10.3.7 Reassociate.....	100
22	10.3.7.1 MLME-REASSOCIATE.request	100
23	10.3.7.2 MLME-REASSOCIATE.confirm	100
24	10.3.7.3 MLME-REASSOCIATE.indication.....	101
25	10.3.10 Start.....	101
26	10.3.10.1 MLME-START.request.....	101
27	10.3.11 TS management interface	102
28	10.3.11.1 MLME-ADDTS.request.....	102
29	10.3.11.2 MLME-ADDTS.confirm.....	103
30	10.3.11.3 MLME-ADDTS.indication	105
31	10.3.11.4 MLME-ADDTS.response	106
32	10.3.11.5 MLME-DELTS.request.....	108
33	10.3.11.6 MLME-DELTS.confirm.....	108
34	10.3.11.7 MLME-DELTS.indication	109
35	10.3.12 Management of Direct Links	110
36	10.3.12.1 MLME-DLP.request.....	110
37	10.3.12.2 MLME-DLP.confirm.....	111
38	10.3.12.3 MLME-DLP.indication	112
39	10.3.12.4 MLME-DLPteardown.request	113
40	10.3.12.5 MLME-DLPteardown.confirm	113
41	10.3.12.6 MLME-DLPteardown.indication	114
42	10.3.13 Higher Layer Synchronization Support.....	114
43	10.3.13.1 MLME-HL-SYNC.request.....	114
44	10.3.13.2 MLME-HL-SYNC.confirm.....	115
45	10.3.13.3 MLME-HL-SYNC.indication	115
46	10.3.14 Block Ack	116

1	10.3.14.1 MLME-ADDBA.request	116
2	10.3.14.2 MLME-ADDBA.confirm	117
3	10.3.14.3 MLME-ADDBA.indication	118
4	10.3.14.4 MLME-ADDBA.response	119
5	10.3.14.5 MLME-DELBA.request	120
6	10.3.14.6 MLME-DELBA.confirm	121
7	10.3.14.7 MLME-DELBA.indication	121
8	10.3.15 Schedule Element Management	122
9	10.3.15.1 MLME-SCHEDULE.request	122
10	10.3.15.2 MLME-SCHEDULE.confirm	123
11	10.3.15.3 MLME-SCHEDULE.indication	123
12	11. MAC sublayer management	125
13	11.2 Power Management	125
14	11.2.1 Power management in an infrastructure network	125
15	11.2.1.4 Power management with APSD	125
16	11.2.1.4.5 AP operation during CP	126
17	11.2.1.6.7 Receive operation for STAs in PS mode during the CP	128
18	11.2.1.8 Receive operation for STAs in PS mode during the CFP	129
19	11.2.1.9 Receive operation for non-AP QSTAs using APSD	129
20	11.2.1.10 STAs operating in the Active mode	129
21	11.2.1.11 AP aging function	129
22	11.2.2 Power management in an IBSS	129
23	11.4 Traffic Stream operation	129
24	11.4.1 Introduction (Informative)	129
25	11.4.2 TSPEC Construction (informative)	130
26	11.4.3 TS lifecycle	130
27	11.4.4 TS setup	131
28	11.4.5 Failed TS setup	132
29	11.4.6 Data transfer	133
30	11.4.7 TS deletion	134
31	11.4.8 TS timeout	134
32	11.4.9 TS suspension	135
33	11.4.10 TS Reinstatement	135
34	11.5 Block Ack operation	135
35	11.5.1 Set up and modification of the Block Ack parameters	135
36	11.5.1.1 Procedure at the originator	136
37	11.5.1.2 Procedure at the recipient	136
38	11.5.2 Tear down of the Block Ack mechanism	136
39	11.5.2.1 Procedure at the initiator of the Block Ack teardown	137
40	11.5.2.2 Procedure at the recipient of the DELBA frame	137
41	11.5.3 Error recovery upon a peer failure	137
42	11.6 Higher-layer timer synchronization	138
43	11.6.1 Introduction (Informative)	138
44	11.6.2 Procedure at the QSTA	139
45	11.7 DLP operation	139
46	11.7.1 DLP set up	140

1	11.7.1.1 Set up procedure at the QSTA.....	140
2	11.7.1.2 Set up procedure at the QAP	141
3	11.7.2 Data transfer after DLP set up	142
4	11.7.3 DLP Teardown.....	142
5	11.7.3.1 Teardown procedure at the QSTA.....	142
6	11.7.3.2 Teardown procedure at the AP	142
7	11.7.4 Error recovery upon a peer failure.....	142
8	Annex A.....	144
9	A.4.3 IUT configuration	144
10	A.4.x QoS Base Functionality	144
11	A.4.y QoS Enhanced Distributed Channel Access	145
12	A.4.z QoS HCF Controlled Channel Access	145
13	Annex C.....	146
14	Annex D.....	147
15	Annex D.....	147
16	Annex E.....	160
17	E.1 General.....	160
18	Annex H.....	161
19	H.1 Example use of TSPEC for admission control.....	161
20	H.2 Recommended Practices for Contention-based Admission Control	162
21	H.2.1 Use of Admission Control Mandatory (ACM) Subfield.....	162
22	H.2.2 Deriving Medium Time	162
23	H.3 Guidelines and reference design for example scheduler and admission control unit	
24		162
25	H.3.1 Guidelines for Deriving Service Schedule Parameters.....	162
26	H.3.2 TSPEC Construction (informative).....	163
27	H.3.3 Reference Design for an example Scheduler and Admission Control Unit.....	165
28	H.3.3.1 An example scheduler	165
29	H.3.3.2 An Admission Control Unit.....	167
30		

Draft Amendment to STANDARD FOR Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements -

Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications:

Medium Access Control (MAC) Enhancements for Quality of Service (QoS)

[This amendment is based on the current edition of IEEE Std 802.11[®], 1999 (Reaff 2003) Edition and IEEE Std 802.11g[®], 2003 and IEEE Std 802.11h[®], 2003]

NOTE—The editing instructions contained in this amendment define how to merge the material contained herein into the existing base standard to form the new comprehensive standard as created by the addition of IEEE Std 802.11-1999 (Reaff 2003).

The editing instructions are shown in ***bold italic***. Three editing instructions are used: change, delete, and insert. ***Change*** is used to make small corrections in existing text or tables. The editing instruction specifies the location of the change and describes what is being changed either by using strikethrough (to remove old material) or underscore (to add new material). ***Delete*** removes existing material. ***Insert*** adds new material with-out disturbing the existing material. Insertions may require renumbering. If so, renumbering instructions are given in the editing instruction.

1. Overview

1.2 Purpose

Insert the following text at the end of 1.2, as part of the indented list:

- Defines the MAC procedures to support LAN applications with Quality of Service (QoS) requirements, including the transport of voice, audio and video over IEEE 802.11[®] wireless LANs.

2. Normative references

Insert the following citation at the appropriate location in clause 2:

ISO/IEC 15802-3: 1998, Information Technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Common specifications – Part 3: Media Access Control (MAC) Bridges.

3. Definitions

Change the definition of coordination function in 3.13 as follows:

3.13 coordination function: The logical function that determines when a station operating within a basic service set (BSS) is permitted to transmit and may be able to receive protocol data units (PDUs) via the wireless medium (WM). The coordination function within a BSS may have one hybrid coordination function (HCF) and/or may have one point coordination function (PCF) and will have one distributed coordination function (DCF). A QoS basic service set (QBSS) will have one DCF and one HCF.

Change the term “coordination function pollable” to “contention free pollable” and change its definition in 3.14 as follows:

~~**3.14 coordination function**~~**contention free (CF) pollable:** A station that is able to (1) respond to a ~~coordination function~~**contention free (CF)** poll with a data frame, if such a frame is queued and able to be generated; and (2) ~~interpret acknowledgments in frames sent to or from a point coordinator.~~

Insert the following new definitions at the appropriate locations in clause 3, renumbering as necessary:

3.51 access category (AC): A label for the common set of enhanced distributed channel access (EDCA) parameters that are used by a QSTA for the contention of the channel for the transmission of MSDUs with certain priorities.

3.52 aggregated schedule: The aggregation of delivery and/or poll schedules by the QAP for a particular QSTA into a single service period.

3.53 channel access function: A logical function in a QoS station (QSTA) that determines, using enhanced distributed channel access (EDCA), when a frame in the transmit queue with the associated AC is permitted to be transmitted via the wireless medium.

3.54 contention free period (CFP): The time period during operation of a point coordination function (PCF) when the right to transmit is assigned to stations solely by a point coordinator (PC), allowing frame exchanges to occur without contention between members of that BSS for the wireless medium (WM).

3.55 contention period (CP): The time period outside of the CFP in a point-coordinated BSS. In a BSS where there is no PC, this corresponds to the entire time of operation of the BSS.

3.56 controlled access phase (CAP): A time period when the HC maintains control of the medium, after sensing the channel to be idle for a PIFS duration and gaining access to the medium. It may span multiple consecutive transmission opportunities (TXOPs) and may contain polled TXOPs.

3.57 direct link: A unidirectional link from one non-AP QSTA to another non-AP QSTA operating in the same infrastructure QBSS that does not pass through a QAP. Once a direct link has been set up, all frames between the two non-AP QSTAs are exchanged directly.

3.58 downlink: A unidirectional link from an AP to a non-AP STA.

3.59 enhanced distributed channel access (EDCA): The prioritized CSMA/CA access mechanism used by QSTAs in a QBSS. This access mechanism is also used by the QAP and operates concurrently with HCCA.

3.60 fragmentation: The process of partitioning a MAC service data unit (MSDU) or MAC management protocol data unit (MMPDU) into a sequence of smaller MAC protocol data units (MPDUs) prior to transmission. The process of recombining a set of fragment MPDUs into an MSDU or MMPDU is known as defragmentation.

3.61 hidden station: A station whose transmissions cannot be detected using carrier sense by a second station but whose transmissions interfere with transmissions from the second station to a third station.

1 **3.62 hybrid coordination function (HCF):** A coordination function that combines and enhances aspects of
2 the contention-based and contention-free access methods to provide QoS stations (QSTAs) with prioritized
3 and parameterized QoS access to the wireless medium (WM), while continuing to support non-QoS STAs for
4 best-effort transfer. The HCF includes the functionality provided by both EDCA and HCCA. The HCF is
5 compatible with the distributed coordination function (DCF) and may optionally contain the point
6 coordination function (PCF). It supports a uniform set of frame formats and exchange sequences that QSTAs
7 may use during both the contention period (CP) and the contention free period (CFP).

8 **3.63 hybrid coordinator (HC):** A type of coordinator - defined as part of the QoS facility - that implements
9 the frame exchange sequences and MSDU handling rules defined by the hybrid coordination function. The
10 HC operates during both the CP and CFP. The HC performs bandwidth management including the allocation
11 of TXOPs to QSTAs. The HC is collocated with a QAP.

12 **3.64 HCF controlled channel access (HCCA):** The channel access mechanism utilized by the HC to
13 coordinate contention free media use by QSTAs for downlink, uplink and direct link transmissions.

14 **3.65 link:** In the context of an IEEE 802.11[®] MAC entity, a physical path consisting of exactly one traversal
15 of the wireless medium that is used to transfer an MSDU between two STAs.

16 **3.66 non-AP QSTA:** A STA that supports the QoS facility, but is not an AP.

17 **3.67 non-QoS AP (nQAP):** An AP that does not support the QoS facility

18 **3.68 non-QoS BSS (nQBSS):** A BSS that does not support the QoS facility.

19 **3.69 non-QoS STA:** A STA that does not support the QoS facility.

20 **3.70 piggyback:** The overloading of a frame of type Data with an acknowledgement of a previously received
21 MAC Protocol Data Unit (MPDU) and/or a poll to the STA to which the frame is directed.

22 **3.71 QoS access point (QAP):** An AP that supports the QoS facility specified in this standard. The functions
23 of a QAP are a superset of the functions of an nQAP, and thus a QAP is able to function as an nQAP to non-
24 QoS STAs.

25 **3.72 QoS basic service set (QBSS):** A basic service set (BSS) that provides the QoS facility. An
26 infrastructure QBSS contains a QAP.

27 **3.73 QoS facility:** The set of enhanced functions, channel access rules, frame formats, frame exchange
28 sequences and managed objects to provide parameterized and prioritized QoS.

29 **3.74 QoS independent basic service set (QIBSS):** An independent basic service set (IBSS) in which one or
30 more of its STAs support the QoS facility.

31 **3.75 QoS station (QSTA):** A station (STA) that implements the QoS facility. A QSTA acts as a non-QoS
32 STA when associated in an nQBSS.

33 **3.76 service period (SP):** A service period is a contiguous time during which of one or more downlink
34 frames are transmitted to a QSTA and/or one or more TXOPs are granted to the same QSTA. Service periods
35 start at fixed intervals of time.

36 **3.77 service interval (SI):** The interval between two successive SPs.

37 **3.78 traffic category (TC):** A label for those MSDUs that have a distinct user priority (UP), as viewed by
38 higher-layer entities, relative to other MSDUs provided for delivery over the same link. Traffic categories are
39 only meaningful to MAC entities that support QoS within the MAC data service. These MAC entities

1 determine the UP for MSDUs belonging to a particular traffic category using the priority value provided with
2 those MSDUs at the MAC_SAP.

3 **3.79 traffic classification (TCLAS):** A traffic classification specifies certain parameter values to identify the
4 MSDUs belonging to a particular traffic stream (TS). The classification process, performed above the
5 MAC_SAP at a QSTA, uses the parameter values of the traffic classification for a given TS to examine each
6 incoming MSDU and determine whether this MSDU belongs to that TS.

7 **3.80 traffic identifier (TID):** Any of the identifiers usable by higher-layer entities to distinguish MSDUs to
8 MAC entities that support quality of service (QoS) within the MAC data service. There are 16 possible TID
9 values, 8 identify TCs and the other 8 identify parameterized TSs.

10 **3.81 traffic specification (TSPEC):** The QoS characteristics of a data flow to and from a non-AP QSTA.

11 **3.82 traffic stream (TS):** A traffic stream is a set of MSDUs to be delivered subject to the QoS parameter
12 values provided to the MAC in a particular TSPEC. Traffic streams are only meaningful to MAC entities that
13 support QoS within the MAC data service. These MAC entities determine the TSPEC applicable for delivery
14 of MSDUs belonging to a particular traffic stream using the TSID value provided with those MSDUs at the
15 MAC_SAP.

16 **3.83 traffic stream identifier (TSID):** Any of the identifiers usable by higher-layer entities to distinguish
17 MSDUs to MAC entities for parameterized quality of service (QoS) (i.e., the TS with a particular TSPEC)
18 within the MAC data service.

19 **3.84 transmission opportunity (TXOP):** An interval of time when a particular QSTA has the right to initiate
20 frame exchange sequences onto the wireless medium (WM). A TXOP is defined by a starting time and a
21 maximum duration. The TXOP is either obtained by the QSTA by successfully contending for the channel or
22 is assigned by the HC.

23 **3.85 TXOP holder:** A TXOP holder is a QSTA that has either been granted a TXOP by the HC or
24 successfully contended for a TXOP.

25 **3.86 uplink:** A unidirectional link from a non-AP STA to an AP.

26 **3.87 user priority (UP):** A value associated with an MSDU that indicates how the MSDU should be
27 handled.

4. Abbreviations and acronyms

Delete the acronym "CID" from clause 4

~~CID~~ ~~connection identifier~~

Insert the following new acronyms at appropriate locations in clause 4:

AC	access category
ACI	AC index
ACM	admission control mandatory
ADDBA	add block ack
ADDTS	add traffic stream
AIFS	arbitration inter frame space
AIFSN	arbitration inter frame space number
APSD	automatic power-save delivery
BA	block acknowledgement
BAR	block acknowledgement request
CA	collision avoidance
CAP	controlled access phase
CSMA	carrier sense multiple access
DELBA	delete block ack
DELTS	delete traffic stream
DLP	direct link protocol
DSCP	differentiated services code point
EDCA	enhanced distributed channel access
EOSP	end of service period
HCCA	HCF controlled channel access
HC	hybrid coordinator
HCF	hybrid coordination function
nQAP	non-QoS access point
nQBSS	non-QoS BSS
QAP	QoS access point
QBSS	quality of service basic service set
QIBSS	quality of service independent basic service set
QoS	quality of service
QSTA	QoS station
SI	service interval
SP	service period
TC	traffic category
TCLAS	traffic classification
TID	traffic identifier
TS	traffic stream
TSID	traffic stream identifier
TSPEC	traffic specification
TXOP	transmission opportunity
UP	user priority

5. General description

5.1 General description of the architecture

5.1.1 How wireless LAN systems are different

5.1.1.2 ~~The m~~Media impact ~~the on~~ design and performance

Change item b) of the indented list as follows:

- b) Are potentially unprotected from ~~outside other~~ signals that may be sharing the medium.

Insert the following text at the end of the indented list:

- g) May experience interference from logically disjoint 802.11 networks operating in adjacent or overlapping areas.

Insert the following text at the end of the final paragraph:

When providing QoS services it should be understood that the MAC endeavors to provide QoS "service guarantees" within the limitations of the medium properties identified above. That is, particularly in unlicensed spectrum, true guarantees often are not possible. However gradations of service are always possible, and in sufficiently controlled environments, QoS guarantees can truly be made.

5.1.1.4 Interaction with other IEEE 802[®] layers

Insert the following paragraph at the end of 5.1.1.4:

When used to support applications with quality of service requirements, each IEEE 802.11[®] LAN provide a link within an end-to-end QoS environment that may be established between, and managed by, higher layer entities. To handle QoS traffic in a manner comparable to other IEEE 802 LANs, the IEEE 802.11[®] QoS facility requires the IEEE 802.11[®] MAC sublayers to incorporate functionality that is not traditional for MAC sublayers. In addition, it may be necessary for certain higher layer management entities to be "WLAN aware" at least to the extent of understanding that the available bandwidth and other QoS characteristics of a WLAN are subject to frequent, and sometimes substantial, dynamic changes due to causes other than traffic load and are outside the direct control of network management entities.

5.2 Components of the IEEE 802.11[®] architecture

Insert the following subclause after 5.2.4:

5.2.5 QBSS: The quality of service network

The IEEE 802.11[®] QoS facility provides MAC enhancements to support LAN applications with quality of service (QoS) requirements. The QoS enhancements are available to QoS stations (QSTAs) associated with a QoS access point (QAP) in a QoS BSS (QBSS). A subset of the QoS enhancements is available for use between QSTAs that are members of the same QoS IBSS (QIBSS). Because a QSTA implements a superset of STA functionality, as defined in IEEE Std. 802.11[®], 1999 edition (Reaffirm 2003), the QSTA may associate with an nQAP in an nQBSS, to provide non-QoS MAC data service in cases where there is no QBSS with which to associate.

The enhancements that distinguish QSTAs from non-QoS STAs and QAPs from nQAPs are collectively termed the QoS facility. The quantity of certain, QoS-specific, mechanisms may vary among QoS implementations, as well as between QSTAs and QAPs, over ranges specified in subsequent clauses. All service primitives, frame formats, coordination function and frame exchange rules, and management interface functions except for the Block Acknowledgement function, Direct Link Protocol and Automatic Power-save

Delivery (APSD) are part of the core QoS facilities. A QSTA or QAP must implement those core QoS facilities necessary for its QoS functions to interoperate with other QSTAs in the QBSS. Functions such as the Block Acknowledgement, DLP and APSD are separate from the core QoS facilities, and the presence of these functions is indicated by QSTAs separately from the core QoS facilities. A comprehensive statement on mandatory and optional functionalities is available in Annex A.

This standard provides two mechanisms for the support of applications with QoS requirements.

The first mechanism, designated the EDCA, delivers traffic based on differentiating UPs. This differentiation is achieved through varying the amount of time a station senses the channel to be idle before backoff or transmission, the length of the contention window to be used for the backoff or the duration a station may transmit after it acquires channel access. These transmissions may also be subject to certain channel access restrictions in the form of an admission control. Details of this mechanism are provided in clause 9.9.1.

The second mechanism, designated the HCCA, allows for the reservation of transmission opportunities (TXOPs) with the HC. A non-AP QSTA based on its requirements requests the HC for TXOPs – both for its own transmissions as well as transmissions from the QAP to itself.¹ The request is initiated by the Station Management Entity (SME) of the non-AP QSTA. The HC, which is collocated at the QAP, either accepts or rejects the request based on an admission control policy. If the request is accepted, the HC schedules TXOPs for both the QAP and the non-AP QSTA. For transmissions from the non-AP QSTA, the HC polls the non-AP QSTA based on the parameters supplied by the non-AP QSTA at the time of its request. For transmissions to the non-AP QSTA, the QAP directly obtains TXOPs from the collocated HC and delivers the queued frames to the non-AP QSTA, again based on the parameters supplied by the non-AP QSTA. Details of the mechanism are provided in clauses 9.9.2 and 11.4. This mechanism is expected to be used for applications such as voice and video, which may need periodic service from the HC. If the application constraints dictate the use of this mechanism, the application initiates this mechanism using the management service primitives.

Non-QoS STAs may associate in a QBSS, if allowed to associate by the QAP. All directed frames that are sent to non-QoS STAs by a QAP shall not use the frame formats associated with the QoS facility.

5.3 Logical service interfaces

Insert the following items at the end of the list:

- j) Higher-layer timer synchronization (QoS facility only)
- k) QoS traffic scheduling (QoS facility only)

5.3.1 Station service (SS)

Insert the following items at the end of the list:

- e) Higher-layer timer synchronization (QoS facility only)
- f) QoS traffic scheduling (QoS facility only)

5.3.2 Distribution system service (DSS)

Insert the following item at the end of the list:

- f) QoS traffic scheduling (QoS facility only)

¹ In the case of downlink traffic streams.

5.4 Overview of the services

Change the first paragraph as follows:

There are ~~nine~~many services specified by IEEE 802.11[®]. Six of the services are used to support MSDU delivery between STAs. Three of the services are used to control IEEE 802.11[®] LAN access and confidentiality. One of the services provides support for LAN applications with QoS requirements. Another of the services provides support for higher-layer timer synchronization.

5.4.1 Distribution of messages within a DS

Insert the following subclause after 5.4.1.2:

5.4.1.3 QoS traffic scheduling

QoS traffic scheduling provides intra-QBSS QoS frame transfers under the hybrid coordination function (HCF), using either contention-based or controlled channel access. At each TXOP, a traffic scheduling entity at the QSTA selects a frame for transmission, from the set of frames at the heads of a plurality of traffic queues, based on requested UP and/or parameter values in the traffic specification for the requested MSDU. Additional information is available in clause 9.9.3.

5.4.2 Services that support the distribution service

5.4.2.2 Association

Change the fifth paragraph of 5.4.2.2 as follows:

A STA learns what APs are present and what operational capabilities are available from each of those APs, and then requests establishment of an association with an AP of appropriate capabilities by invoking the association service. For details of how a station learns about what APs are present, see 11.1.3.

Insert the following subclause after the end of 5.4.3.3:

5.4.4 Traffic differentiation and QoS support

IEEE 802.11[®] uses a shared medium and provides differentiated control of access to the medium to handle data transfers with QoS requirements. The QoS facility (per MSDU traffic class and TSPEC negotiation) allows an IEEE 802.11[®] LAN to become part of a larger network providing end-to-end QoS delivery, or to function as an independent network providing transport on a per-link basis with specified QoS commitments.

5.4.5 Support for higher-layer timer synchronization

Some applications, for example, the transport and rendering of audio or video streams, require synchronized timers shared among different STAs. Greater accuracy (in terms of jitter bounds) or finer timer granularity than that provided by a BSS TSF may be an additional requirement. In support of such applications, this standard defines a MAC service that enables layers above the MAC to accurately synchronize application-dependent timers shared among QSTAs. The service is usable by more than one application at a time.

Although the timer synchronization methods and accuracy requirements are application-dependent and are beyond the scope of this standard, they rely on an indication from each QSTA's MAC that is provided essentially simultaneously, via multicast, to the QSTAs. The MAC accomplishes this by indicating the occurrence of the end of the last symbol of particular data frames; the data frames of interest are identified by their MAC header Address 1 field when it contains a group address previously registered with the MAC. The last symbol is observed on the air simultaneously by QSTAs within a BSS while the delay between the observation and the delivery of the indication is known within a MAC by design (and communicated to the application by implementation-dependent means). The common reference point in time provided by the end of last symbol indication is the essential building-block upon which a variety of application-dependent timer synchronization methods may be based.

5.5 Relationships between services

Change the definition of item vi in class 1 management frames as follows:

2) Management frames

vi) Spectrum Management Action.

Change the definition of class 1 data frames as follows:

3) Data frames

i) Data: Data frames between STAs in an IBSS with frame control (FC) bits "To DS" and "From DS" both false.

Change the definition of class 1 and 3 frames as follows:

c) Class 3 frames (if and only if associated; allowed only from within State 3):

1) Data frames

i) Data subtypes: Data frames allowed. That is, the "To DS" and/or "From DS" FC bits may be set to true to utilize DSSs.

ii) QoS data subtypes allowed to/from non-AP QSTA(s) that are associated with QAP(s).

iii) Data frames between QSTAs in a QBSS with FC bits "To DS" and "From DS" both false.

2) Management frames

i) Deauthentication: Deauthentication notification when in State 3 implies disassociation as well, changing the STA's state from 3 to 1. The station shall become authenticated again prior to another association.

ii) QoS, DLP and Block Ack Action

3) Control frames

i) PS-Poll

ii) Block Ack (BlockAck)

iii) Block Ack Request (BlockAckReq)

5.6 Differences between ESS and IBSS LANs

Change the final paragraph in this clause as follows:

The services that apply to an IBSS are the SSs. A QIBSS supports operation under the HCF using TXOPs gained through the EDCA mechanism. The parameters that control differentiation of traffic classes using EDCA are fixed. A QIBSS has no HC and does not support polled TXOP operation and setting up of TSPEC.

5.7 Message information contents that support the services

5.7.1 Data

Change the subclause as shown below:

For a STA to send data to another STA, it sends ~~one of the following~~ data messages, ~~as~~ shown below:

Data Messages

- Message type: Data
- Message sub-type: Data
- Information items:

- IEEE source address of message
- IEEE destination address of message
- BSS identification (BSS ID)
- Direction of message: From QSTA or non-QoS STA to non-QoS STA

Insert the QoS Data message at the end of 5.7.1 (with appropriate leader characters):

QoS Data Messages

- Message type: Data
- Message sub-type: QoS Data
- Information items:
 - IEEE source address of message
 - IEEE destination address of message
 - BSS ID
 - Traffic Identifier
- Direction of message: From QSTA to QSTA

5.7.2 Association

Insert the following text under "Association request" just after entry for "ESSID" (with appropriate leader character):

- Requester's capabilities

Insert the following text under "Association response" just after entry beginning "If the association..."(with appropriate leader character):

- Responder's capabilities

5.7.3 Reassociation

Insert the following text under "Reassociation request" just after entry for "ESSID" (with appropriate leader character):

- Requester's capabilities

Insert the following text under "Reassociation response" just after entry beginning "If the reassociation..."(with appropriate leader character):

- Responder's capabilities

6. MAC service definition

6.1 Overview of MAC services

Change the title and the existing paragraph in 6.1.1 as follows:

6.1.1 Asynchronous-~~d~~Data service

This service provides peer LLC entities with the ability to exchange MSDUs. To support this service, the local MAC uses the underlying PHY-level services to transport an MSDU to a peer MAC entity, where it will be delivered to the peer LLC. Such asynchronous MSDU transport is performed on a ~~best effort~~ connectionless basis. By default, MSDU transport is on a best effort basis. However, the QoS facility uses a traffic identifier to specify differentiated services on a per-MSDU basis. The QoS facility also permits more synchronous behavior to be supported on a connection oriented basis using TSPECs. There are no guarantees that the submitted MSDU will be delivered successfully. Broadcast and multicast transport is part of the ~~asynchronous~~ data service provided by the MAC. Due to the characteristics of the WM, broadcast and multicast MSDUs may experience a lower quality of service, compared to that of unicast MSDUs. All STAs support the asynchronous data service, but only QSTAs in a QBSS differentiate their MSDU delivery according to the designated traffic category or traffic stream of individual MSDUs.

Because operation of certain functions of the MAC may cause reordering of some MSDUs, as discussed in more detail below, in non-QoS STAs, there are two service classes within the ~~asynchronous~~ data service. By selecting the desired service class, each LLC entity initiating the transfer of MSDUs is able to control whether MAC entities are or are not allowed to reorder those MSDUs.

For unicast MSDUs sent or received by QSTAs, there are two service classes, designated as QoSAck and QoSNoAck, and all MSDUs may be reordered. The service classes QoSAck and QoSNoAck are used to signal if the MSDU is to be transmitted using the MAC-level acknowledgement or not. For broadcast and multicast MSDUs sent or received by a non-AP QSTA there is an additional service class, QoSLocalMulticast. The QoSLocalMulticast service class causes broadcast and multicast MSDUs to be delivered directly from a non-AP QSTA without being relayed by a QAP. Conversely, broadcast and multicast MSDUs using the QoS class are sent as unicast MPDUs to the QAP and the QAP transmits these MPDUs as broadcast/multicast MPDUs within the QBSS and delivers them as well to the QAP's bridging layer.

Insert the following paragraph at the end of 6.1.1:

In QSTAs either associated in a QBSS or having membership in a QIBSS, the MAC uses a set of rules that tends to cause higher UP MSDUs in a BSS to be sent before lower UP MSDUs in the BSS. The MAC sublayer entities determine the UPs for MSDUs based on the TID values provided with those MSDUs. If a TSPEC has been provided for a traffic stream, via the MAC sublayer management entity, the MAC attempts to deliver MSDUs belonging to that traffic stream in accordance with the QoS parameter values contained in the TSPEC. In a QBSS with some QSTAs supporting the QoS facility and others not supporting the QoS facility, in delivering an MSDU delivery to a non-QoS STA, the QSTA uses the access category corresponding to the UP of the MSDU.

Insert the following subclauses between 6.1.1 and 6.1.2:

6.1.1.1 Determination of UP

The QoS facility supports 8 priority values, referred to as user priorities (UPs). The values an UP may take are the integer values from 0 to 7 and are identical to the IEEE 802.1D priority tags. An MSDU with a particular UP is said to belong to a traffic category (TC) with that UP. The UP is provided with each MSDU at the medium access control service access point (MAC_SAP), either directly, in the UP parameter, or indirectly, in a traffic specification (TSPEC) designated by the UP parameter.

6.1.1.1.1 Determination of UP of received frames at the QAP sent by other STAs in the BSS

The received unicast frames at the QAP may be:

- a) Non-QoS subtypes, in which case the QAP shall assign to them a priority of Contention if they are received during the CP, or ContentionFree, if they are received during the CFP.
- b) QoS subtypes, in which case the QAP shall infer the UP value from the TID in the QoS control field directly for TID values between 0 and 7. For TID values between 8 and 15, the QAP shall extract the UP value in the user priority subfield of the TSInfo field in the associated TSPEC or from the User priority field in the associated TCLAS element, as applicable.

QAPs deliver the user priority with the received MSDUs to the DS.

6.1.1.2 Interpretation of priority parameter in MAC service primitives

The value of the priority parameter in the MAC service primitives (see subclause 6.2) may be a non-integer value of either Contention or Contention Free, or may be any integer value in the range 0 through 15.

When the priority parameter has an integer value, it is used in the traffic identifier (TID) fields that appear in certain frames that are used to deliver, and to control the delivery of QoS data across the WM.

Priority parameter and TID field values 0 through 7 are interpreted as UPs for the MSDUs. Outgoing MSDUs with UP values 0 through 7 are handled by MAC entities at QSTAs in accordance with the UP.

Priority parameter and TID field values 8 through 15 specify TIDs that are also traffic stream identifiers (TSIDs), and select the TSPEC for the TS designated by the TID. Outgoing MSDUs with priority parameter values 8 through 15 are handled by MAC entities at QSTAs in accordance with the UP value determined from the user priority subfield as well as other parameter values in the selected TSPEC. Selection of a TSPEC for which the MLME has not provided QoS parameter values is equivalent to using a TSPEC with a user priority subfield value equal to the TID value minus 8 and the current, local default values for all other QoS parameters.

The non-integer values of the priority parameter are allowed at all non-QoS STAs. The integer values of the priority parameter (i.e., TID) are supported only at QSTAs that are either associated in a QBSS or members of a QIBSS. A range of 0 through 15 is supported by QSTAs associated in a QBSS; where as a range of 0 through 7 is supported by QSTAs that are members of a QIBSS. If a QSTA is associated in an nQBSS, the QSTA is functioning as a non-QoS STA, so the priority value is always Contention or ContentionFree.

At QSTAs associated in a QBSS, MSDUs with priority of Contention are considered equivalent to MSDUs with TID 0, and those with priority of ContentionFree are delivered using the Contention-free delivery, if a point coordinator (PC) is present in the QAP. If a PC is not present, a QSTA should attempt to deliver traffic with ContentionFree priority using a polled service by initiating a TSPEC with default parameters and a UP of zero and Access Policy set to HCCA. If a TSPEC cannot be setup by the QSTA, traffic marked ContentionFree shall be delivered using an UP of 0. At QSTAs associated in an nQBSS, all MSDUs with an integer priority are considered equivalent to MSDUs with priority Contention.

If a QSTA is associated in a QBSS, the MSDUs it receives in QoS data frames are reported with the TID value contained in the MAC header of that frame; and the MSDUs it receives in non-QoS data frames are reported to LLC with priority of Contention if they are received during the CP, or ContentionFree, if they are received during the CFP.

6.1.1.3 Access Categories

When a QSTA uses contention-based access to transmit a frame, the access is based on the access category of the MSDU that is to be transmitted. The access category (AC) is derived from the UPs as shown in Table 0.1.

Table 0.1 – User priority to Access Category mappings

Priority	User priority	802.1D	Access	Designation
----------	---------------	--------	--------	-------------

	(UP – Same as 802.1D User Priority)	Designation	Category (AC)	(Informative)
lowest  highest	1	BK	AC_BK	Background
	2	-	AC_BK	Background
	0	BE	AC_BE	Best Effort
	3	EE	AC_BE	Video
	4	CL	AC_VI	Video
	5	VI	AC_VI	Video
	6	VO	AC_VO	Voice
	7	NC	AC_VO	Voice

6.1.1.4 Interpretation of service class parameter in MAC service primitives in a QSTA

In QSTAs, the value of the service class parameter in the MAC service primitive (see subclause 6.2) may be a non-integer value of QoSack, QoSNoAck or QoSLocalMulticast.

When an MSDU is received from the MAC_SAP, and the recipient STA is a QSTA and with the service class set to

- a) QoSack, the MSDU is transmitted using a QoS Data type frame with the Ack Policy subfield in the QoS Control field set to either “Normal Acknowledgement” or “Block Acknowledgement”.
- b) QoSNoAck, the MSDU is transmitted using a QoS Data type frame with the Ack Policy subfield in the QoS Control field set to “No Acknowledgement”.
- c) QoSLocalMulticast, the MSDU is transmitted using a QoS Data type frame with a multicast/broadcast address in address 1 field of the frame and the ToDS and FromDS subfields in the Frame Control field both set to 0.

When an MSDU is received from the MAC_SAP, and the recipient STA is not a QSTA, the MSDU is transmitted using a non-QoS Data type frame.

When a frame is received from another QSTA, the service class parameter in MA-UNITDATA.indication is set to

- a) QoSack, if the frame is a QoS Data type frame with the Ack Policy subfield in the QoS Control field set to either “Normal Acknowledgement” or “Block Acknowledgement”.
- b) QoSNoAck, if the frame is a QoS Data type frame with the Ack Policy subfield in the QoS Control field set to “No Acknowledgement”.
- c) QoSLocalMulticast, if the frame is a QoS Data type frame with the address 1 field set to a multicast/broadcast address and the ToDS and FromDS subfields in the Frame Control field both set to 0.

6.1.3 MSDU ordering

Change the first paragraph of 6.1.3 as follows:

The services provided by the MAC sublayer permit, and may in certain cases require, the reordering of MSDUs.

In a non-QoS STA, the MAC does not intentionally reorder MSDUs except as may be necessary to improve the likelihood of successful delivery based on the current operational (“power management”) mode of the designated recipient station(s), or to honor the priority parameter specified in the MA-UNITDATA.request of individual MSDUs. The sole effect of this reordering (if any), for the set of MSDUs received at the MAC

service interface of any single station, is a change in the delivery order of broadcast and multicast MSDUs, relative to directed MSDUs, originating from a single source station address. If a higher-layer protocol using the ~~asynchronous~~-data service cannot tolerate this possible reordering, the optional StrictlyOrdered service class should be used. MSDUs transferred between any pair of stations using the StrictlyOrdered service class are not subject to the relative reordering that is possible when the ReorderableMulticast service class is used. However, the desire to receive MSDUs sent using the StrictlyOrdered service class at a station precludes simultaneous use of the MAC power management facilities at that station.

In non-AP QSTAs operating in a QBSS, there are three service classes, designated as QoSAck, QoSNoAck and QoSLocalMulticast. In a QAP, QoSAck and QoSNoAck are the only service classes available (see 6.1.1.4) for more information. The MSDUs are reordered, not only to improve the likelihood of successful delivery based on the current operational mode of the designated recipient station(s), but also to honor the priority parameters, specified in the MA-UNITDATA.request primitive, of the individual MSDUs. The effects of this reordering (if any), for the set of MSDUs received at the MAC service interface of any single station, are

- a) a change in the delivery order of broadcast and multicast MSDUs, relative to unicast MSDUs,
- b) the reordering of MSDUs with different TID values, originating from a single source station address and
- c) the reordering of broadcast and multicast MSDUs with the same TID but different service classes.

There shall be no reordering of unicast MSDUs with the same TID value and addressed to the same destination.

QSTAs operating in an nQBSS shall follow the reordering rules as defined for a non-QoS STA.

Insert the following subclause, including figures therein, after 6.1.3:

6.1.4 MAC Data service architecture (Informative)

The MAC data plane architecture (i.e. those processes that are involved in the transport of all or part of an MSDU) is shown in Figure 11.1. During transmission an MSDU goes through some or all of the following processes: Power Save-defer queuing, sequence number assignment, fragmentation, encryption and frame formatting. At some point the Data frames that contain all or part of the MSDU are queued per access category/TS. This queuing may be at any of the three points indicated in Figure 11.1.

During reception, a received Data frame goes through processes of MPDU header + CRC validation, duplicate removal, possible reordering if the Block Ack mechanism is used, decryption, and defragmentation. After defragmentation, the MSDU is delivered to the MAC-DATA SAP or to the DS.

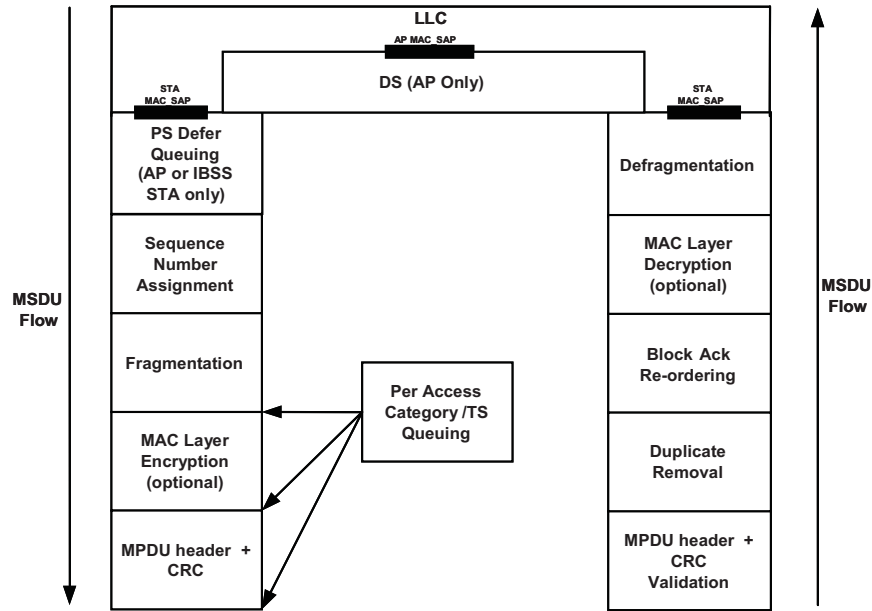


Figure 11.1 – MAC Data-Plane Architecture

6.2 Detailed service specification

6.2.1 MAC data services

6.2.1.1 MA-UNITDATA.request

6.2.1.1.2 Semantics of the service primitive

Change the final 2 paragraphs as follows:

The priority parameter specifies the priority desired for the data unit transfer. ~~IEEE 802.11 allows two values: Contention or ContentionFree. The allowed values of priority are described in 6.1.1.2.~~

The service class parameter specifies the service class desired for the data unit transfer. ~~IEEE 802.11 allows two values: ReorderableMulticast or StrictlyOrdered. The allowed values of service class are described in 6.1.3.~~

6.2.1.1.4 Effect of receipt

Change the existing paragraph as follows:

~~On~~The receipt of this primitive, ~~causes~~the MAC sublayer entity determines whether the request can be fulfilled according to the requested parameters. A request that cannot be fulfilled according to the requested parameters is discarded and this action is indicated to the LLC sublayer entity using an MA-UNITDATA-STATUS.indication primitive that describes why the MAC was unable to fulfill the request. If the request can be fulfilled according to the requested parameters, the MAC sublayer entity ~~to~~appends all MAC specified fields; (including DA, SA, FCS, and all fields that are unique to IEEE 802.11[®]), and passes the properly formatted frame to the lower layers for transfer to a peer MAC sublayer entity or entities (see 6.1.4), and indicates this action to the LLC sublayer entity using an MA-UNITDATA-STATUS.indication primitive with transmission status set to "Successful".

6.2.1.2 MA-UNITDATA.indication

6.2.1.2.2 Semantics of the service primitive

Change the final 3 paragraphs as follows:

The reception status parameter indicates the success or failure of the received frame for those frames that IEEE 802.11[®] reports via a MA-UNITDATA.indication. This MAC only reports "success" ~~when~~because all failures of reception are discarded without generating MA-UNITDATA.indication.

The priority parameter specifies the receive processing priority that was used for the data unit transfer. ~~IEEE 802.11 allows two values: Contention or ContentionFree.~~ The allowed values of priority are described in 6.1.1.2.

The service class parameter specifies the receive service class that was used for the data unit transfer. ~~IEEE 802.11[®] allows two values: ReorderableMulticast or StrictlyOrdered.~~ The allowed values of service class are described in 6.1.3.

6.2.1.3 MA-UNITDATA-STATUS.indication

6.2.1.3.2 Semantics of the service primitive

Change the final 3 regular paragraphs, including the indented list, as follows:

The transmission status parameter is used to pass status information back to the local requesting LLC sublayer entity. IEEE 802.11[®] specifies the following values for transmission status:

- a) Successful;
- b) Undeliverable (for unacknowledged directed MSDUs when the ~~adot11ShortRetryLimitMax~~ or ~~adot11LongRetryLimitMax~~ retry limit would otherwise be exceeded);
- c) Undeliverable (Excessive data length);
- d) Undeliverable (Non-null source routing);
- e) Undeliverable. Unsupported priority (for priorities other than Contention or ContentionFree at a non-QoS STA; or for priorities other than Contention, ContentionFree or an integer between and including 0 and 15 at a QSTA);
- f) Undeliverable. Unsupported service class (for service classes other than ReorderableMulticast or StrictlyOrdered} for non-QoS STAs and service classes other than QoSack, QoSNoAck or QoSLocalMulticast for QSTAs;
- g) Unavailable priority (for ContentionFree when no point coordinator or hybrid coordinator is available, or an integer between and including 1 and 15 at a QSTA that is associated in an nQBSS, or an integer between and including 8 and 15 at a QSTA that is a member of an IBSS, in which case the MSDU is transmitted with and a provided priority of Contention);
- h) Undeliverable. Unavailable service class (for StrictlyOrdered service when the station's power management mode is other than "active" for non-QoS STAs; QSTAs do not return this value as they do not provide the StrictlyOrdered service);
- i) Undeliverable (TransmitMSDUTimer reached dot11MaxTransmitMSDULifetime before successful delivery);
- j) Undeliverable (no BSS available);
- k) Undeliverable (cannot encrypt with a null key).
- l) Undeliverable (MSDU lifetime expired prior to successful delivery).
- m) Undeliverable (BlockAck Timeout value has been exceeded).

- 1 The provided priority parameter specifies the priority that was used for the associated data unit transfer
2 (Contention, ~~or~~ ContentionFree or an integer between and including 0 and 15).
- 3 The provided service class parameter specifies the class of service used for the associated data unit transfer.
4 In non-QoS STAs, the value of this parameter is {ReorderableMulticast or StrictlyOrdered}. In QSTAs, it is
5 QoSAck, QoSNoAck or QoSLocalMulticast.

7. Frame formats

Change the text of the paragraph in 7 as follows:

The format of the MAC frames is specified in this clause. ~~All stations~~ A STA shall be able to properly construct a subset of the frames specified in this subclause for transmission, and to decode a (potentially different) subset of the frames specified in this subclause upon validation following reception, ~~as specified in this clause~~. The particular subset of these frames that a station constructs and decodes is determined by the functions supported by that particular station, as specified in 7.5. All stations shall be able to validate every received frame using the frame check sequence (FCS), and to interpret certain fields from the MAC headers of all frames.

7.1 MAC frame formats

Change the text of 7.1 as follows:

Each frame consists of the following basic components:

- a) A MAC header, which comprises frame control, duration, address, sequence control information, and, for QoS data frames, QoS control information;
- b) A variable length frame body, which contains information specific to the frame type and subtype;
- c) A frame check sequence (FCS), which contains an IEEE 32-bit cyclic redundancy code (CRC).

7.1.1 Conventions

Insert the following text above the final paragraph of 7.1.1:

Reception, in references to frames or fields within frames (e.g. "... beacon frames received by a STA ..." or "... the value of a received Duration/ID field ...") applies to MPDUs or MMPDUs indicated from the PHY layer without error and validated by FCS within the MAC sublayer. Without further qualification, "reception" by the MAC sublayer implies that the frame contents are valid, and that the protocol version is supported (see 7.1.3.1.1), with no implication regarding frame addressing, or if the frame type or other fields in the MAC header are meaningful to the MAC entity that has received the frame.

Parentheses enclosing portions of names or acronyms are used to designate a set of related names that vary based on the inclusion of the parenthesized portion. For example:

- "QoS +CF-Poll frame" refers to the 3 QoS data subtypes that include "+CF-Poll," the QoS Data+CF-Poll, subtype 1010, QoS Data+CF-Ack+CF-Poll, subtype 1011, and QoS CF-Ack+CF-Poll, subtype 1111;
- "QoS CF-Poll frame" refers specifically to the QoS CF-Poll frame, subtype 1110;
- "QoS (+)CF-Poll frame" refers to all 4 QoS data subtypes with CF-Poll mentioned above, the QoS CF-Poll frame, subtype 1110, the QoS CF-Ack+CF-Poll frame, subtype 1111, the QoS Data+CF-Poll frame, subtype 1010, and the QoS Data+CF-Ack+CF-Poll frame, subtype 1011;
- "QoS (+)Null frame" refers to all 4 QoS data subtypes with "no data", the QoS Null (no data) frame, subtype 1100, the QoS CF-Ack (no data) frame, subtype 1101, the QoS CF-Poll (no data) frame, subtype 1110, and the QoS CF-Ack+CF-Poll frame, subtype 1111;
- "QoS CF-Ack(+CF-Poll) frame" refers to the QoS CF-Ack frame, subtype 1101, and the QoS CF-Ack+CF-Poll frame, subtype 1111; whereas
- "(QoS) CF-Poll frame" refers to the QoS CF-Poll frame, subtype 1110, and the CF-Poll frame, subtype 0110.

7.1.2 General frame format

Change the text in 7.1.2 and Figure 12 as follows:

The MAC frame format comprises a set of fields that occur in a fixed order in all frames. Figure 12 depicts the general MAC frame format. The first three fields (Frame Control, Duration/ID and Address 1) and the last field (FCS) in Figure 12 constitute the minimal frame format and are present in all frames, including reserved types and subtypes. The fields Address 2, Address 3, Sequence Control, Address 4, QoS Control, and Frame Body are only present in certain frame types and subtypes. Each field is defined in 7.1.3. The format of each of the individual subtypes of each frame types is defined in 7.2. The components of management frame bodies are defined in 7.3. The formats of management action frames are defined in 7.4

The frame body is of variable size. The maximum frame body size is determined by the maximum MSDU size (2304 octets) plus any overhead from security encapsulation.

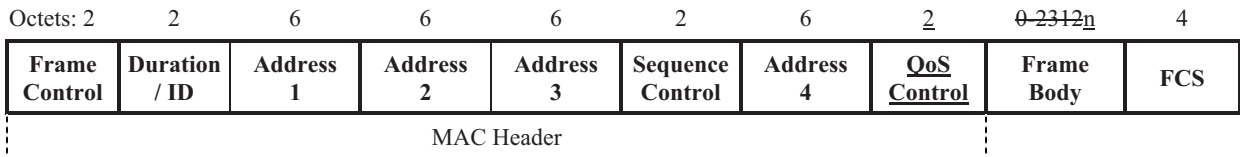


Figure 12 – MAC frame format

7.1.3 Frame fields

7.1.3.1 Frame Control field

7.1.3.1.2 Type and Subtype fields

Change the text of the subclause as shown:

The Type field is 2 bits in length, and the Subtype field is 4 bits in length. The Type and Subtype fields together identify the function of the frame. There are three frame types: control, data, and management. Each of the frame types have several defined subtypes. In type data frames, the most significant bit of the subtype field, b7, is defined as the QoS subfield. Table 1 defines the valid combinations of type and subtype.

Change the contents of Table 1 and insert the note below Table 1 as follows:

**Table 1 - Valid type and subtype combinations
(numeric values in Table 1 are shown in binary)**

Type value b3 b2	Type description	Subtype value ² b7 b6 b5 b4	Subtype description
00	Management	0000	Association request
00	Management	0001	Association response
00	Management	0010	Reassociation request
00	Management	0011	Reassociation response

² Decoding the subtypes of data type frames can take advantage of the fact that each subtype field bit position is used to indicate a specific modification of the basic data frame (subtype 0). Frame control bit 4 is set to 1 in data subtypes which include +CF-Ack, bit 5 is set to 1 in data subtypes which include +CF-Poll, bit 6 is set to 1 in data subtypes that contain no Frame Body, and bit 7 is set to 1 in the QoS data subtypes, which have QoS Control fields in their MAC headers.

Type value b3 b2	Type description	Subtype value ² b7 b6 b5 b4	Subtype description
00	Management	0100	Probe request
00	Management	0101	Probe response
00	Management	0110-0111	Reserved
00	Management	1000	Beacon
00	Management	1001	Announcement traffic indication message (ATIM)
00	Management	1010	Disassociation
00	Management	1011	Authentication
00	Management	1100	Deauthentication
<u>00</u>	<u>Management</u>	<u>1101</u>	<u>Action</u>
00	Management	11010-1111	Reserved
01	Control	0000- ~~1001~~ 11	Reserved
<u>01</u>	<u>Control</u>	<u>1000</u>	<u>Block Acknowledgement Request (BlockAckReq)</u>
<u>01</u>	<u>Control</u>	<u>1001</u>	<u>Block Acknowledgement (BlockAck)</u>
01	Control	1010	Power Save Poll (PS-Poll)
01	Control	1011	Request To Send (RTS)
01	Control	1100	Clear To Send (CTS)
01	Control	1101	Acknowledgement (ACK)
01	Control	1110	Contention-Free (CF)-End
01	Control	1111	CF-End + CF-Ack
10	Data	0000	Data
10	Data	0001	Data + CF-Ack
10	Data	0010	Data + CF-Poll
10	Data	0011	Data + CF-Ack + CF-Poll
10	Data	0100	Null ~~function~~ (no data)
10	Data	0101	CF-Ack (no data)
10	Data	0110	CF-Poll (no data)
10	Data	0111	CF-Ack + CF-Poll (no data)
10	Data	1000- ~~1111~~	<u>QoS Data</u> ~~Reserved~~
<u>10</u>	<u>Data</u>	<u>1001</u>	<u>QoS Data + CF-Ack</u>
<u>10</u>	<u>Data</u>	<u>1010</u>	<u>QoS Data + CF-Poll</u>
<u>10</u>	<u>Data</u>	<u>1011</u>	<u>QoS Data + CF-Ack + CF-Poll</u>
<u>10</u>	<u>Data</u>	<u>1100</u>	<u>QoS Null (no data)</u>
<u>10</u>	<u>Data</u>	<u>1101</u>	<u>QoS CF-Ack (no data)</u>
<u>10</u>	<u>Data</u>	<u>1110</u>	<u>QoS CF-Poll (no data)</u>
<u>10</u>	<u>Data</u>	<u>1111</u>	<u>QoS CF-Ack + CF-Poll (no data)</u>
11	Reserved	0000-1111	Reserved

7.1.3.1.3 To DS field

Change the text in 7.1.3.1.3 as follows:

The To DS field is one bit in length and is set to 1 in data type frames destined for the DS. This includes all data type frames sent to an AP, including all data type frames sent by non-QoS STAs associated with an AP, and all frames being sent between APs using the WM as a wireless distribution system (WDS). The To DS field is set to 0 in all other frames. For additional details, see Table 2.

7.1.3.1.4 From DS field

Change the text in 7.1.3.1.4 and Table 2 as follows:

The From DS field is 1 bit in length and is set to 1 in data type frames exiting the DS. This includes all data type frames sent by APs, including frames sent using the WM as a WDS. ~~The From DS field~~ is set to 0 in all other frames.

The permitted To/From DS bit combinations and their meanings are given in Table 2.

Table 2 – To/From DS combinations in data type frames

To/From DS values	Meaning
To DS = 0 From DS = 0	A data type frame direct from one STA to another STA within the same IBSS, or a data type frame direct from one non-AP QSTA to another non-AP QSTA within the same QBSS, or a data type broadcast/multicast frame from a QSTA to other STAs within the same QBSS using the QoSLocalMulticast service class, as well as all management and control type frames. STAs are not required to be capable of receiving frames delivered with the QoSLocalMulticast service class.
To DS = 1 From DS = 0	~~Data frame destined for the DS.~~ A data frame from a STA to the DS via an AP.
To DS = 0 From DS = 1	~~Data frame exiting the DS.~~ A data frame from the DS to a STA via an AP.
To DS = 1 From DS = 1	Wireless distribution system (WDS) data frame being distributed from one AP to another AP via the WM.

7.1.3.1.8 More Data field

Insert the following paragraph between second and third paragraphs of the subclause:

For a non-AP QSTA that has the 'More Data Ack' subfield set in its QoS Capability information element and also has APSD enabled, a QAP may set the More Data field to 1 in ACK frames to this non-AP QSTA to indicate that it has a pending transmission for it.

7.1.3.1.10 Order field

Change the contents of the subclause as shown:

The Order field is 1 bit in length and is set to 1 in any non-QoS data type frame that contains an MSDU, or fragment thereof, which is being transferred using the StrictlyOrdered service class. This field is set to 0 in all other frames. All QSTAs set this subfield to 0.

7.1.3.2 Duration/ID field

Change the contents of 7.1.3.2 as follows:

The Duration/ID field is 16 bits in length. The contents of this field ~~are vary with frame type and subtype, whether the frame is transmitted during the CFP, and QoS capabilities of the sending station, and are defined~~ as follows:

- a) In control type frames of subtype Power Save (PS)-Poll the Duration/ID field carries the association identity (AID) of the station that transmitted the frame in the 14 least-significant bits (lsb), ~~with~~and the 2 most-significant bits (msb) both set to 1. The value of the AID is in the range 1-2007.
- b) In frames transmitted by the PC and non-QoS STAs, during the contention free period (CFP), the Duration field is set to a fixed value of 32768. In all other frames sent by non-QoS STAs and frames of type control sent by QSTAs, the Duration/ID field contains a duration value as defined for each frame type in 7.2. In frames of type data and management sent by QSTAs, the Duration/ID field contains a duration value as defined for each frame type in 7.1.4. For frames transmitted during the contention free period (CFP), the duration field is set to 32768.

Whenever the contents of a received Duration/ID field, treated as an unsigned integer and without regard for address values, type and subtype (even when type or subtype contain reserved values), are less than 32768, the duration value is used to update the network allocation vector (NAV) according to the procedures defined in ~~Clause 9.2.5.4 or 9.9.2.2.1, as appropriate.~~

Whenever the contents of a received Duration/ID field, treated as an unsigned integer, are greater than 32768, the contents are interpreted as appropriate for the frame type and subtype, or ignored if the receiving MAC entity does not have a defined interpretation for that type and subtype.

The encoding of the Duration/ID field is given in Table 3.

Delete Table 3.

Insert Table 3 as shown below

Table 3 – Duration/ID field encoding

Bits 0-13	Bit 14	Bit 15	Usage
0 – 32767		0	<u>Duration value (in units of microseconds) within all frames other than PS-Poll frames transmitted during the CP, and under HCF for frames transmitted during the CFP.</u>
0	0	1	Fixed value <u>under PCF</u> within frames transmitted during the CFP.
1 - 16383	0	1	Reserved
0	1	1	Reserved
1-2007	1	1	AID in PS-Poll frames
2008 - 16383	1	1	Reserved

7.1.3.4 Sequence Control field

Insert at the end of the paragraph in the subclause:

Sequence Control field is not present in control frames.

7.1.3.4.1 Sequence Number field

Change the text of 7.1.3.4.1 as shown (and splitting the text into multiple paragraphs):

The Sequence Number field is a 12-bit field indicating the sequence number of an MSDU or MMPDU. Each MSDU or MMPDU transmitted by a STA is assigned a sequence number. Sequence numbers are not assigned to control frames, as the Sequence Control field is not present.

~~Sequence numbers are~~Non-QoS STAs, as well as QSTAs operating as non-QoS STAs because they are in an nQBSS or nQIBSS, ~~assigned sequence numbers, to management type frames and data type frames (QoS subfield is set to 0),~~ from a single modulo-4096 counter, starting at 0 and incrementing by 1 for each MSDU or MMPDU.

QSTAs associated in a QBSS maintain one modulo-4096 counter for each TID that they source to a unique destination. Sequence numbers for QoS data type frames are assigned using the counter identified by the TID subfield of the QoS Control field of the frame, and that counter is incremented by 1 for each MSDU belonging to that TID. Sequence numbers for management type frames, QoS Data subtype frames with a broadcast/multicast address in the address 1 field, and all non-QoS Data subtype frames sent by QSTAs and QAPs are assigned using an additional single modulo-4096 counter, starting at 0 and incrementing by 1 for each MSDU or MMPDU. Sequence numbers for QoS (+)Null type frames may be set to any value.

Each fragment of an MSDU or MMPDU contains the assigned sequence number. The sequence number remains constant in all retransmissions of an MSDU, MMPDU, or fragment thereof.

Insert after 7.1.3.4.2 the following subclauses 7.1.3.5, as well as the new figures contained therein, and renumber subsequent 7.1.x-subclauses and figures as necessary:

7.1.3.5 QoS Control field

The QoS Control field is a 16-bit field that identifies the TC or TS to which the frame belongs and various other QoS-related information about the frame that varies by frame type and subtype. The QoS Control field is present in all frames of type Data in which the QoS subfield is set to one (see 7.1.3.1.2). Each QoS Control field comprises 5 subfields, as defined for the particular sender (HC or non-AP QSTA) and frame type and subtype. The usage of these subfields and the various possible layouts of the QoS Control field are described below and illustrated in Table 3.1.

Table 3.1 – QoS Control field

Applicable Frame (sub) Types	Bits 0-3	Bit 4	Bits 5-6	Bit 7	Bits 8-15
QoS (+)CF-Poll frames sent by HC	TID	EOSP	Ack Policy	Reserved	TXOP limit in units of 32 microseconds
QoS Data, QoS Null, QoS CF-Ack and QoS Data+CF-Ack frames sent by HC	TID	EOSP	Ack Policy	Reserved	Reserved
QoS data type frames sent by non-AP QSTAs	TID	0	Ack Policy	Reserved	TXOP duration requested in units of 32 microseconds
	TID	1	Ack Policy	Reserved	Queue size in units of 256 octets

7.1.3.5.1 TID field

The TID field identifies the TC or TS to which the corresponding MSDU, or fragment thereof, in the frame body field belongs. The TID field also identifies the TC or TS of traffic for which a TXOP is being requested, through the setting of TXOP duration requested or queue size. The most significant bit of the TID, when set

to 0, indicates that the request is for data associated with prioritized QoS and when set to 1, indicates that the request is for data associated with parameterized QoS. The TID field contains the value of the priority parameter from the MA-UNITDATA.request primitive that provided the MSDU to which the QoS control field applies. The format of the TID field is shown in Table 3.1.1. Additional information on the interpretation of the contents of this field appears in 6.1.1.2.³

Table 3.1.1 – TID field

Bits 0-3	Usage
0-7	UP for prioritized QoS (TC)
8-15	TSID for parameterized QoS (TS)

For QoS Data+CF-Poll the TID subfield in the QoS Control indicates the TID of the data. For all QoS (+)CF-Poll frames of subtype Null, the TID subfield in the QoS Control indicates the TID for which the poll is intended. The requirement to respond to that TID is non-binding and a QSTA may respond with any frame.

7.1.3.5.2 End of Service Period (EOSP) field

The End of Service Period (EOSP) field is one bit in length and is used by the HC to indicate the end of the current service period (SP). The HC sets the EOSP field to 1 if the SP ends after the successful transmission of the current frame and sets it to 0 otherwise.

The HC uses the EOSP subfield to indicate the end of SP in APSD. The More Data Bit is used to indicate whether there are MSDUs buffered at the AP at the end of the SP.

7.1.3.5.3 Ack Policy Field

The Ack policy is two bits in length and identifies the Ack policy that is followed upon the delivery of the MPDU. The interpretation of these two bits is given in Table 3.2.

Table 3.2 – Ack policy field in QoS control field of QoS data frames

Bit in QoS Control field:	Bit 5	Bit 6	Meaning
	0	0	Normal acknowledgement. The addressed recipient returns an ACK or QoS (+) CF-ACK frame after a SIFS period, according to the procedures defined in 9.2.8, 9.3.3 and 9.9.2.3. The Ack Policy field is set to this value in all directed frames in which the sender requires acknowledgement. For QoS Null (no data) frames, this is the only permissible value for the Ack Policy field.

³ The presence of the TID field allows an 802.11 QBSS to function as a LAN that is "able to signal the priority".

Bit 5	Bit 6	Meaning
1	0	No Acknowledgement The addressed recipient takes no action upon receipt of the frame. More details are provided in 9.11. The Ack Policy is set to this value in all directed frames in which the sender does not require acknowledgement. For QoS CF-Ack frames, this is the only permissible value for the Ack Policy field. This combination is also used for broadcast and multicast frames that use the QoS frame format.
0	1	No Explicit Acknowledgement. There may be a response frame to the frame that is received, but it is neither the ACK nor any Data frame of subtype +CF-Ack. For Data frames of subtype QoS CF-Poll and subtype QoS CF-Ack+CF-Poll, this is the only permissible value for the Ack Policy field.
1	1	Block Acknowledgement The addressed recipient takes no action upon the receipt of the frame except for recording the state. The recipient can expect a BlockAckReq frame in the future to which it responds using the procedure described in 9.10.

An MSDU is sent using Ack Policy of “Normal Acknowledgement” or “Block Acknowledgement” if the service class parameter in MA-UNITDATA.request is set to QoSAck, and to “No Acknowledgement” if the service class parameter in MA-UNITDATA.request is set to QoSNoAck or QoSLocalMulticast.

7.1.3.5.4 TXOP limit field

The TXOP limit field is an 8-bit field that is present in QoS data type frames of subtypes that include CF-Poll and specifies the time limit on a TXOP granted by a QoS (+)CF-Poll from an HC in a QBSS. In QoS data type frames with subtypes that include CF-Poll, the addressed QSTA is granted a TXOP that begins a SIFS period after this frame and lasts no longer than the number of 32-microsecond periods specified by the TXOP limit value. The range of time values is 32 to 8160 microseconds. A TXOP limit value of 0 implies that no more than one MSDU may be transmitted immediately following the QoS (+)CF-Poll. In QoS control fields of frames transmitted by an HC with subtypes that do not include CF-Poll, the TXOP limit field value is reserved. The TXOP limit is inclusive of the PHY overhead, and a QAP should account for the overhead when granting TXOPs.

7.1.3.5.5 Queue size field

The queue size field is an 8-bit field that indicates the amount of buffered traffic for a given TC or TS at the non-AP QSTA sending this frame. The queue size field is present in data frames of subtype QoS sent by STAs associated in a QBSS with bit 4 of the QoS control field set to 1. The queue size value is the total size, rounded up to the nearest multiple of 256 octets and expressed in units of 256 octets, of all MSDUs buffered at the QSTA (excluding the MSDU of the present QoS data frame) in the delivery queue used for MSDUs with TID values equal to the value in the TID subfield of this QoS Control field. A queue size value of 0 is used solely to indicate the absence of any buffered traffic in the queue used for the specified TID. A queue size value of 254 is used for all sizes greater than 64768 octets. A queue size value of 255 is used to indicate an unspecified or unknown size. If a QoS data type frame is fragmented, the queue size value may remain constant in all fragments even if the amount of queued traffic changes as successive fragments are transmitted.

7.1.3.5.6 TXOP duration requested field

The TXOP duration requested field is an 8-bit field that indicates the duration, in units of 32 microseconds, which the sending station desires for its next TXOP. The range of time values is 32 to 8160 microseconds. The TXOP duration requested field is present in data frames of subtype QoS sent by non-AP QSTAs associated in a QBSS with bit 4 of the QoS control field set to 0. A value of zero in the TXOP duration requested field indicates that no TXOP is requested.

TXOP duration requested field values are not cumulative. A TXOP duration requested for a particular TID supercedes any prior TXOP duration requested for that TID. A value of zero indicates that no TXOP is required for that TID in the current SP. This may be used to cancel a pending unsatisfied TXOP request when its MSDU is no longer queued for transmission. The TXOP duration requested is inclusive of the PHY overhead, and a QSTA should account for this when attempting to determine whether a given transmission fits within a specified TXOP duration.

7.1.3.6 Frame Body field

7.1.3.7 FCS field

Insert the following subclause at the end of 7.1.3.7

7.1.4 Duration/ID field in Data and Management frames

Within all data type frames containing QoS CF-Poll, the Duration/ID value is set to one SIFS duration plus the TXOP Duration Limit.

Within all frames of type data or management sent in contention period by the QSTAs outside of a CAP, following a contention access of the channel, the Duration/ID field is set to one of the following values:

- a) For management type frames, Data type frames with QoS subfield set to 0 and, for unicast data type frames with Ack policy set to "Normal Acknowledgement",
 - 1) the time required for the transmission of one ACK frame (including appropriate IFS values), if the frame is the final fragment of the TXOP; or
 - 2) the time required for the transmission of one ACK plus the time required for the transmission of the following MPDU, and this following MPDU's response frame if one is required (including appropriate IFS values).
- b) For unicast data type frames with Ack policy set to "No Acknowledgement" or "Block Acknowledgement" and for multicast/broadcast frames,
 - 1) 0, if the frame is the final fragment of the TXOP; or
 - 2) the time required for the transmission of the following MPDU and this following MPDU's response frame if one is required (including appropriate IFS values).
- c) the minimum of
 - 1) the time required for the transmission the pending MPDUs of the AC and the associated ACKs, if any, and applicable SIFS durations, and
 - 2) the time limit imposed by the MIB attribute dot11EDCATableTXOPLimit (dot11EDCAQAPTableTXOPLimit for the QAP) for that AC minus the already used time within the TXOP..

Within all data or management type frames sent under HCCA, to ensure NAV protection for the entire CAP, the Duration/ID field is set to one of the following values:

- a) the remaining duration of the TXOP, if the frame is a non-final frame in a TXOP with multiple frame exchanges.

- b) the actual remaining time needed for this frame exchange sequence, if the frame is the sole or final frame in the TXOP.

7.2 Format of individual frame types

7.2.1 Control frames

7.2.1.1 Request To Send (RTS) frame format

Change the final paragraph of 7.2.1.1 and insert the note below this paragraph as shown:

For all RTS frames sent by non-QoS STAs, the duration value is the time, in microseconds, required to transmit the pending data or management frame, plus one CTS frame, plus one ACK frame, plus three SIFS intervals. If the calculated duration includes a fractional microsecond, that value is rounded up to the next higher integer. For all RTS frames sent by QSTAs under EDCA, following a contention access of the channel, the duration value is set in the following manner:

- a) If the NAV protection is desired for only the first or sole frame in the TXOP the duration value is set to, the time, in microseconds, required to transmit the pending frame, plus one CTS frame, plus one ACK frame if required, plus three SIFS intervals.
- b) Otherwise, the duration value is set to the remaining duration of the TXOP.

For all RTS frames sent under HCCA, the duration value is set to one of the following values:

- a) If the pending frame is the final frame, the duration value is set to the time, in microseconds, required to transmit the pending frame, plus one CTS frame, plus one ACK frame if required, plus three SIFS intervals.
- b) If the pending frame is not the final frame in the TXOP, the duration value is set to the remaining duration of the TXOP.

7.2.1.2 Clear To Send (CTS) frame format

Change the final paragraph of 7.2.1.2 as shown:

For all CTS frames sent in response to RTS frames, the duration value is the value obtained from the duration field of the immediately previous RTS frame, minus the time, in microseconds, required to transmit the CTS frame and its SIFS interval. If the calculated duration includes a fractional microsecond, that value is rounded up to the next higher integer.

At a non-AP QSTA, if the CTS is the first frame in the exchange and the pending data or management frame requires acknowledgement, the duration value is the time, in microseconds, required to transmit the pending data or management frame, plus one SIFS interval, plus one ACK frame, and an additional SIFS interval. At a non-AP QSTA, if the CTS is the first frame in the exchange and the pending data or management frame does not require acknowledgement, the duration value is the time, in microseconds, required to transmit the pending data or management frame, plus one SIFS interval. If the calculated duration includes a fractional microsecond, that value is rounded up to the next higher integer.

For all CTS frames sent by QSTAs as the first frame in the exchange under EDCA, the duration value is set in the following manner:

- a) If the NAV protection is desired for only the first or sole frame in the TXOP the duration value is set to,
 - 1) the time, in microseconds, required to transmit the pending frame, plus one SIFS interval, plus the response frame (ACK or Block Ack), plus an additional SIFS interval, if there is a response frame.
 - 2) the time, in microseconds, required to transmit the pending frame, plus one SIFS interval, if there is no response frame.

b) Otherwise, the duration value is set to the remaining duration of the TXOP.

For CTS frames sent under HCCA, the duration value is set to one of the following values:

a) If the pending frame is the sole frame in the TXOP, the duration value is set to

1) the time, in microseconds, required to transmit the pending frame, plus the response frame (ACK or Block Ack), plus an additional SIFS interval, if there is a response frame.

2) the time, in microseconds, required to transmit the pending frame, plus one SIFS interval, if there is no response frame.

b) If the pending frame is not the final frame in the TXOP, the duration value is set to the remaining duration of the TXOP.

7.2.1.3 Acknowledgement (ACK) frame format

Change the final paragraph of 7.2.1.3 as shown:

For ACK frames sent by non-QoS STAs, if the More Fragments bit was set to 0 in the Frame Control field of the immediately previous directed data or management frame, the duration value is set to 0. ~~If the More Fragments bit was set to 1 in the Frame Control field of the immediately previous directed data or management frame,~~ In all other frames, the duration value is the value obtained from the duration field of the immediately previous data, or management frame, BlockAckReq frame, or BlockAck frame minus the time, in microseconds, required to transmit the ACK frame and its SIFS interval. If the calculated duration includes a fractional microsecond, that value is rounded up to the next higher integer.

Insert after 7.2.1.6 the following subclauses 7.2.1.7 through 7.2.1.8, as well as the new figures contained therein, and renumber subsequent figures as necessary:

7.2.1.7 Block Acknowledgement Request (BlockAckReq) frame format

The frame format of the Block Acknowledgement Request (BlockAckReq) frame is defined in Figure 21.1.

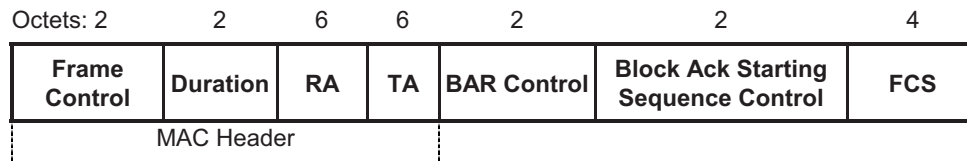


Figure 21.1 – BlockAckReq frame

The duration value is greater than or equal to the time⁴, in microseconds, required to transmit, one ACK or BlockAck frame, as applicable, plus one SIFS interval. If the calculated duration includes a fractional microsecond, that value is rounded up to the next higher integer.

The RA field of the BlockAckReq frame is the address of the recipient QSTA.

The TA field is the address of the QSTA transmitting the BlockAckReq frame.

The BAR Control field is shown in Figure 21.2.

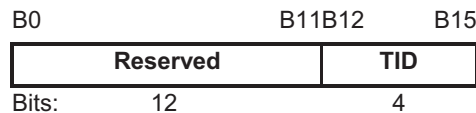
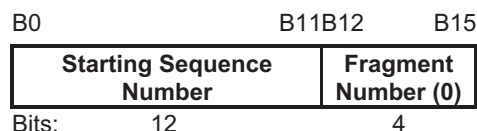


Figure 21.2 – BAR Control field

⁴ To allow the possibility of time remaining in the TXOP which the sender may use to schedule other transmissions.

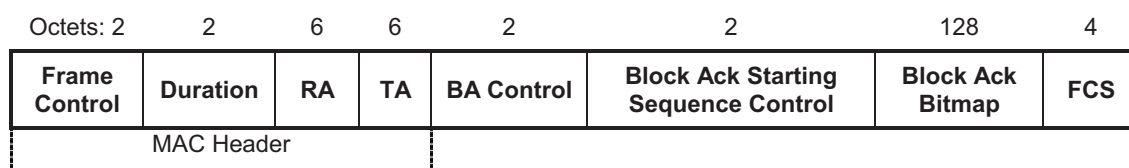
- 1 The TID subfield of BAR Control contains the TID for which a BlockAck frame is requested.
- 2 The Block Ack Starting Sequence Control field is shown in Figure 21.3. The Starting Sequence number
- 3 subfield is the sequence number of the first MSDU for which this BlockAckReq is sent. The Fragment
- 4 number subfield is always set to 0.



5 **Figure 21.3– Block Ack Starting Sequence Number**

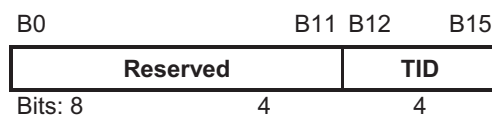
6 7.2.1.8 Block Acknowledgement (BlockAck) frame format

- 7 The frame format of the Block Acknowledgement (BlockAck) frame is defined in Figure 21.4.



8 **Figure 21.4 – BlockAck Frame**

- 9 If the BlockAck frame is sent in response to the BlockAckReq frame, the duration value is the value obtained
- 10 from the Duration field of the immediate BlockAckReq frame, minus the time, in microseconds, required to
- 11 transmit the BlockAck frame and its SIFS interval. If the BlockAck frame is not sent in response to the
- 12 BlockAckReq, the duration value is greater than (subject to TXOP Limit) or equal to the time⁵ for
- 13 transmission of an ACK frame plus a SIFS interval. If the calculated duration includes a fractional
- 14 microsecond, that value is rounded to the next higher integer.
- 15 The RA field of the BlockAck frame is the address of the recipient QSTA that requested the Block Ack.
- 16 The TA field is the address of the QSTA transmitting the BlockAck frame.
- 17 The BA control field defined in Figure 21.4 consists of the TID subfield.



18 **Figure 21.4 – BA Control field**

- 19 The Block Ack Starting Sequence Control field is defined in 7.2.1.7 and is set to the same value as in the
- 20 immediately previously received BlockAckReq frame.
- 21 The Block Ack Bitmap is 128 octets in length and is used to indicate the receiving status of up to 64 MSDUs.
- 22 Bit position *n* of the Block Ack bitmap, if set to 1, acknowledges receipt of an MPDU with MPDU Sequence
- 23 Control value equal to (Block Ack Starting Sequence Control + *n*). Bit position *n* of the Block Ack bitmap, if
- 24 set to 0, indicates that an MPDU with MPDU Sequence Control value equal to (Block Ack Starting Sequence
- 25 Control + *n*) has not been received. For unused fragment numbers of an MSDU, the corresponding bits in the
- 26 bitmap are set to 0.

⁵ ibid

7.2.2 Data frames

Change the text, figures and tables in 7.2.2 as shown, renumber subsequent figures as necessary:

The frame format for a Data frame is independent on the QoS subfield of the subtype field and is as defined in Figure 22.

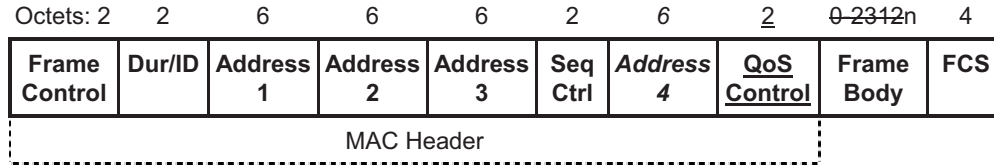


Figure 22 – Data Frame

Data frames with a value of 1 in the QoS subfield of the subtype field are collectively referred to as QoS data type frames. Each of these data subtypes contains "QoS" in their names, and this frame format is distinguished by the presence of a QoS Control field in the MAC header. Data frames with a value of 0 in the QoS subfield of the subtype field do not have the QoS Control field.

A QSTA always uses QoS data type frames for data transmissions to other QSTAs. A QSTA uses frames with QoS subfield of the subtype field set to 0 for data transmissions to non-QoS STAs. A non-QoS STA always uses frames with QoS subfield of the subtype field set to 0 for data transmissions to other STAs. All STAs use frames with QoS subfield of the subtype field set to 0 for broadcast data frames unless a transmitting QSTA knows that all STAs in a QBSS have QoS capability, in which case the transmitting QSTAs uses QoS data frames. All STAs use frames with QoS subfield of subtype field set to 0 for multicast data frames unless it is known to the transmitter that all STAs in the QBSS that are members of the multicast group, have QoS capability, in which case QSTAs use QoS data frames.

The contents of the Address fields of frames of type data frames are dependent upon the values of the To DS and From DS bits in the frame control field, and are defined in Table 4. Where the content of a field is shown as not applicable (N/A), the field is omitted. Note that Address 1 always holds the receiver address of the intended receiver (or, in the case of multicast frames, receivers), and that Address 2 always holds the address of the station that is transmitting the frame.

Table 4 – Address Field Contents

To DS	From DS	Address 1	Address 2	Address 3	Address 4	Usage
0	0	DA	SA	BSSID	N/A	<u>STA to STA traffic in an IBSS and QSTA-to-QSTA traffic in a QBSS</u>
0	1	DA	BSSID	SA	N/A	<u>AP-to-STA traffic in a BSS</u>
1	0	BSSID	SA	DA	N/A	<u>STA-to-AP traffic in a BSS</u>
1	1	RA	TA	DA	SA	<u>WDS traffic between APs</u>

A station uses the contents of the Address 1 field to perform address matching for receive decisions. In cases where the Address 1 field contains a group address, the BSSID also is validated to ensure that the broadcast or multicast originated in the same BSS.

A station uses the contents of the Address 2 field to direct the acknowledgment if an acknowledgment is necessary.

The DA is the destination of the MSDU (or fragment thereof) in the frame body field.

The SA is the address of the MAC entity which initiated the MSDU (or fragment thereof) in the frame body field.

1 The RA is the address of the STA contained in the AP in the wireless distribution system that is the next
2 immediate intended recipient of the frame.

3 The TA is the address of the STA contained in the AP in the wireless distribution system that is transmitting
4 the frame.

5 The BSSID of the Data frame is determined as follows:

6 a) If the station is an AP or is associated with an AP, the BSSID is the address currently in use by the
7 STA contained in the AP.

8 b) If the station is a member of an IBSS, the BSSID is the BSSID of the IBSS.

9 The Sequence Control field is defined in 7.1.3.4. The Sequence Control field for QoS (+) Null type frames is
10 ignored by the receiver upon reception.

11 The QoS Control field is defined in 7.1.3.5.

12 The frame body consists of the MSDU or a fragment thereof, and a WEP IV and ICV (if and only if the WEP
13 subfield in the frame control field is set to 1). The frame body is null (0 octets in length) in data frames of
14 Subtype Null function (no data), CF-Ack (no data), CF-Poll (no data), and CF-Ack+CF-Poll (no data),
15 regardless of the encoding of the QoS subfield in the Frame Control.

16 For Data frames of Subtype Null (no data), CF-Ack (no data), CF-Poll (no data) and CF-Ack+CF-Poll (no
17 data) and the corresponding four QoS data frame subtypes, the frame body is omitted – these subtypes are
18 used for MAC control purposes. For Data frames of subtypes Data, Data+CF-Ack, Data+CF-Poll and
19 Data+CF-Ack+CF-Poll and the corresponding four QoS data frame subtypes, the frame body contains all of
20 or a fragment of an MSDU after any encapsulation for security.

21 The maximum length of the frame body can be determined from the maximum MSDU length plus any
22 overhead from encapsulation for encryption (i.e. it is always possible to send a maximum length MSDU, with
23 any encapsulations provided by the MAC layer within a single Data MPDU).

24 Within all data type frames sent by STAs during the CFP under PCF, the Duration/ID field is set to 32768.

25 Within all data type frames sent by the QSTA, the duration/ID contains a duration value as defined in 7.1.4.

26 Within all data type frames sent during the contention period (CP) by non-QoS STAs, the Duration/ID field is
27 set according to the following rules:

28 — If the Address 1 field contains a group address, the duration value is set to 0.

29 — If the More Fragments bit is set to 0 in the Frame Control field of a frame and the Address 1 field
30 contains an individual address, the duration value is set to the time, in microseconds, required to
31 transmit one ACK frame, plus one SIFS interval.

32 — If the More Fragments bit is set to 1 in the Frame Control field of a frame, and the Address 1 field
33 contains an individual address, the duration value is set to the time, in microseconds, required to
34 transmit the next fragment of this data frame, plus two ACK frames, plus three SIFS intervals.

35 The duration value calculation for the data frame is based on the rules in 9.6 that determine the data rate at
36 which the control frames in the frame exchange sequence are transmitted. If the calculated duration includes a
37 fractional microsecond, that value is rounded up to the next higher integer. All stations process Duration field
38 values less than or equal to 32767 from valid data frames to update their NAV settings as appropriate under
39 the coordination function rules.

40 7.2.3 Management frames

41 *Change the text in 7.2.3 as shown:*

42 The frame format for management frames is independent of frame subtype and is as defined in Figure 23.

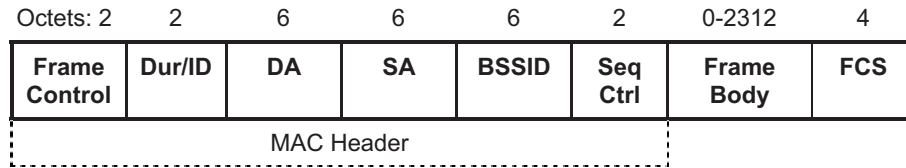


Figure 23 – Management frame format

A STA uses the contents of the Address 1 (DA) field to perform the address matching for receive decisions. In the case where the Address 1 (DA) field contains a group address and the frame type is other than Beacon, the BSSID also is validated to ensure that the broadcast or multicast originated in the same BSS. If the frame type is Beacon, other address matching rules apply, as specified in 11.1.2.3.

The address fields for management frames do not vary by frame subtype.

The BSSID of the management frame is determined as follows:

- a) If the station is an AP or is associated with an AP, the BSSID is the address currently in use by the STA contained in the AP.
- b) If the station is a member of an IBSS, the BSSID is the BSSID of the IBSS.
- c) In management frames of subtype Probe Request, the BSSID is either a specific BSSID, or the broadcast BSSID as defined in the procedures specified in clause 11.1.3.2.

The DA is the destination of the frame.

The SA is the address of the station transmitting the frame.

Within all management type frames sent by STAs during the CFP under PCE, the Duration field is set to the value 32768. Within all management type frames sent by the QSTA, the duration/ID contains a duration value as defined in 7.1.4. Within all management type frames sent during the CP by non-QoS STAs, the Duration field is set according to the following rules:

- If the DA field contains a group address, the duration value is set to 0.
- If the More Fragments bit is set to 0 in the Frame Control field of a frame and the DA contains an individual address, the duration value is set to the time, in microseconds, required to transmit one ACK frame, plus one SIFS interval.
- If the More Fragments bit is set to 1 in the Frame Control field of a frame, and the DA contains an individual address, the duration value is the time, in microseconds, required to transmit the next fragment of this Management frame, plus two ACK frames, plus three SIFS intervals.

The duration value calculation for the management frame is based on the rules in 9.6 that determine the data rate at which the control frames in the frame exchange sequence are transmitted. If the calculated duration includes a fractional microsecond, that value is rounded up to the next higher integer. All stations process Duration field values less than or equal to 32767 from valid management frames to update their NAV settings as appropriate under the coordination function rules.

The frame body consists of the fixed fields ~~and~~ followed by the information elements defined below for each management frame subtype. All fixed fields and information elements are mandatory unless stated otherwise, and ~~they can~~ they can only appear in the specified, relative order. Stations that ~~encountering~~ encountering an element ~~type ID~~ type ID they do not ~~understand~~ recognize in the frame body of a management type frame received, ignore that element and continue to parse the remainder of the management frame body (if any) for additional information elements

with recognizable element IDs. Element type codes not explicitly defined in this standard are reserved and do not appear in any frames. Unused Element ID codes are reserved.

Change the text and table within 7.2.3.1 as follows:

7.2.3.1 Beacon frame format

The frame body of a management frame of subtype Beacon contains the information shown in Table 5. If the dot11MultiDomainCapabilityEnabled attribute is true, a STA shall include a Country Information element in the transmission of Beacon frames. Optionally, the Beacon frame format may also include the information described in either or both of “FH Parameters” orders 12 and “FH Pattern Table” 13. If the information in both “FH Parameters” orders 12 and “FH Pattern Table” 13 are sent, they shall describe the same hopping pattern. Note that the information described in FH Parameters and FH Pattern Table orders 12 and 13 may be also contained in the Probe Response frame.

Insert the following rows at the appropriate location in Table 5:

Order	Information	Notes
14	QBSS Load	The QBSS Load element shall be present when dot11QoSOptionImplemented and dot11QBSSLoadImplemented are both true. The QBSS Load information element is always present within Beacon frames generated by QAPs.
15	EDCA Parameter Set	The EDCA Parameter Set element is present when dot11QoSOptionImplemented is true And QoS Capability element is not present.
23	QoS Capability	The QoS Capability element shall be present when dot11QoSOptionImplemented is true and EDCA Parameter Set element is present.

7.2.3.4 Association Request frame format

Insert the following rows at the appropriate location in Table 5:

Order	Information
9	QoS Action (ADDTS Request)
10	QoS Capability

7.2.3.5 Association Response frame format

Change the contents of Table 8 in 7.2.3.5 as shown:

Table 8 – Association Response frame body

Order	Information
5	EDCA Parameter Set
7	QoS Action (ADDTS Response)

7.2.3.6 Reassociation Request frame format

Change the contents of Table 9 in 7.2.3.6 as shown:

Table 9 – Reassociation Request frame body

Order	Information
<u>10</u>	<u>QoS Action (ADDTS Request)</u>
<u>11</u>	<u>QoS Capability</u>

7.2.3.7 Reassociation Response frame format

Change the contents of Table 10 in 7.2.3.7 as shown:

Table 10 – Reassociation Response frame body

Order	Information
<u>5</u>	<u>EDCA Parameter Set</u>
<u>7</u>	<u>QoS Action (ADTS Response)</u>

7.2.3.9 Probe Response frame format

Change the text and contents of Table 12 in 7.2.3.9 as shown:

The frame body of a management frame of subtype Probe Response ~~contains~~ begins with the information items shown in the "always present" section at the top of Table 12, followed by one or more of the items of order 6 through 9 in Table 12, as appropriate for the characteristics of the active BSS and PHY. These items constitute the mandatory frame body. The additional items in the frame body of Probe Response frames vary depending on the items in the frame body of the Probe Request to which this frame is a response. If the Probe Request was requesting information pertaining to operation in multiple regulatory domains, and if the dot11MultiDomainCapabilityEnabled attribute is true, the Probe Response frame contains a Country information element and all information elements identified by the Requested Element IDs of a Request Information Element. Note that the information returned as a result of a Probe Request frame with a Request Information Element may include the FH parameters and/or the FH Pattern Table possibly replicating optional elements identified by orders 12 and 13.

A STA shall return only those information elements that it supports. In an improperly formed Request information element, a STA may ignore the first information element requested that is not ordered properly and all subsequent information elements requested. In the probe response frame, the STA shall return the requested information elements in the same order as requested in the Request Information Element.

Table 12 – Probe Response frame body

Order	Information	Notes
13	QBSS Load	The QBSS Load element shall be present when <u>dot11QoSOptionImplemented</u> and <u>dot11QBSSLoadImplemented</u> are both true. The QBSS Load information element is always present within Probe Response frames generated by QAPs.
14	EDCA Parameter Set	The EDCA Parameter Set element shall be present when <u>dot11QoSOptionImplemented</u> is true. The EDCA Parameter Set information element is always present within Probe Response frames generated by QAPs.

7.3 Management frame body components

Change the text in the subclause as shown:

Within management frames except in Action frames, fixed-length mandatory frame body components are defined as fixed fields; variable length mandatory and all optional frame body components are defined as information elements. Action frames allow variable fields as explained in clause 7.4.

7.3.1 Fixed fields

7.3.1.4 Capability Information field

Change the contents of the text, Figure 27 and Tables 16 & 17 in 7.3.1.4 as shown:

The Capability Information field contains a number of subfields that are used to indicate-requested or advertised optional capabilities.

The length of the Capability Information field is 2 octets. ~~The Capability Information field consists of the following subfields: ESS, IBSS, CF Pollable, CF Poll Request, Privacy, Short Preamble, PBCC Channel Agility, Spectrum Management, Short Slot Time, and DSSS OFDM. The remaining bits in the Capability Information field are reserved. The format of the Capability Information field is defined as illustrated in Figure 27. No subfield is supplied for ERP as a STA supports ERP operation if it includes all the clause 19 mandatory rates in its supported rate set.~~

B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12	B13	B14	B15
ESS	IBSS	CF-pollable	CF-poll request	Privacy	Short preamble	PBCC	Channel agility	Spectrum Mgmt	QoS	Short Slot Time	APSD	rsrv	DSSS-OFDM	Block Ack	rsrv

Octets:

2

Figure 27 – Capability Information fixed field

Each Capability Information subfield is interpreted according to only in the management frame subtype,s for which the transmission rules are as defined below.

APs set the ESS subfield to 1 and the IBSS subfield to 0 within transmitted Beacon or Probe Response management frames. STAs within an IBSS set the ESS subfield to 0 and the IBSS subfield to 1 in transmitted Beacon or Probe Response management frames.

STAs set the QoS, CF-Pollable and CF-Poll Request subfields in Association and Reassociation Request management frames according to Table 16

Table 16 – STA usage of QoS, CF-Pollable and CF-Poll Request

<u>QoS</u>	<u>CF-Pollable</u>	<u>CF-Poll Request</u>	<u>Meaning</u>
<u>0</u>	0	0	STA is not CF-Pollable
<u>0</u>	0	1	STA is CF-Pollable, not requesting to be placed on the CF-Polling list
<u>0</u>	1	0	STA is CF-Pollable, requesting to be placed on the CF-Polling list
<u>0</u>	1	1	STA is CF-Pollable, requesting never to be polled
<u>1</u>	<u>0</u>	<u>0</u>	<u>QSTA requesting association in a QBSS.</u>
<u>1</u>	<u>0</u>	<u>1</u>	<u>Reserved</u>
<u>1</u>	<u>1</u>	<u>0</u>	<u>Reserved</u>
<u>1</u>	<u>1</u>	<u>1</u>	<u>Reserved</u>

APs set the CF-Pollable and CF-Poll Request subfields in Beacon, and Probe Response, Association Response, and Reassociation Response management frames according to Table 17. An nQAP sets the CF-Pollable and CF-Poll Request subfield values in Association Response and Reassociation Response management frames equal to the values in the last Beacon or Probe Response frame that it transmitted.

Table 17 – AP usage of QoS, CF-Pollable and CF-Poll Request

<u>QoS</u>	<u>CF-Pollable</u>	<u>CF-Poll Request</u>	<u>Meaning</u>
<u>0</u>	0	0	No PC at <u>nQAP</u>
<u>0</u>	0	1	PC at <u>nQAP</u> for delivery only (no polling)
<u>0</u>	1	0	PC at <u>nQAP</u> for delivery and polling
<u>0</u>	1	1	Reserved
<u>1</u>	<u>0</u>	<u>0</u>	<u>QAP (HC) does not use CFP for delivery of unicast data type frames</u>
<u>1</u>	<u>0</u>	<u>1</u>	<u>QAP (HC) uses CFP for delivery, but does not send CF-Polls to non-QoS STAs</u>
<u>1</u>	<u>1</u>	<u>0</u>	<u>QAP (HC) uses CFP for delivery, and sends CF-Polls to non-QoS STAs</u>
<u>1</u>	<u>1</u>	<u>1</u>	<u>Reserved</u>

APs set the Privacy subfield to 1 within transmitted Beacon, Probe Response, Association Response and Reassociation Response management frames if WEP encryption is required for all data type frames exchanged within the BSS. If WEP encryption is not required, the Privacy subfield is set to 0.

STAs within an IBSS set the Privacy subfield to 1 in transmitted Beacon or Probe Response management frames if WEP encryption is required for all data type frames exchanged within the IBSS. If WEP encryption is not required the Privacy subfield is set to 0.

STAs within an ESS set the Privacy subfield to 1 in DLP request and response frames if encryption is required for all data type frames exchanged. If encryption is not required the Privacy subfield is set to 0.

APs (as well as STAs in IBSSs) shall set the Short Preamble subfield to 1 in transmitted Beacon, Probe Response, Association Response and Reassociation Response management MMPDUs to indicate that the use of the Short Preamble, as described in subclause 18.2.2.2, is allowed within this BSS. To indicate that the use of the Short Preamble is not allowed the Short Preamble subfield shall be set to 0 in Beacon, Probe Response, Association Response, and Reassociation Response management MMPDUs transmitted within the BSS.

ERP STAs shall set the MIB variable dot11ShortPreambleOptionImplemented to true as all ERP devices support both long and short preamble formats.

STAs shall set the Short Preamble subfield to 1 in transmitted Association Request and Reassociation Request ~~MMPDU management frames, and DLP request and response frames,~~ when the MIB attribute dot11ShortPreambleOptionImplemented is true. Otherwise STAs shall set the Short Preamble subfield to 0 ~~in transmitted Association Request and Reassociation Request MMPDUs.~~

APs (as well as STAs in IBSSs) shall set the PBCC subfield to 1 in transmitted Beacon, Probe Response, Association Response and Reassociation Response management ~~MMPDU frames~~ to indicate that the PBCC modulation option, as described in subclause 18.4.6.6 and 19.6, is allowed within this BSS. To indicate that the PBCC modulation option is not allowed the PBCC subfield shall be set to 0 ~~in Beacon, Probe Response, Association Response, and Reassociation Response management MMPDUs transmitted within the BSS.~~

STAs shall set the PBCC subfield to 1 in transmitted Association Request and Reassociation Request frames, ~~and DLP Request and Response frames,~~ when the MIB attribute dot11PBCCOptionImplemented is true. Otherwise STAs shall set the PBCC subfield to 0 ~~in transmitted Association Request and Reassociation Request MMPDUs.~~

Bit 7 of the Capabilities Information field shall be used to indicate the usage of Channel Agility by the High Rate direct sequence spread spectrum (HR/ DSSS) PHY or ERP PHY. STAs shall set the Channel Agility bit to 1 when Channel Agility is in use, and shall set it to 0 otherwise.

A STA shall set the Spectrum Management subfield in the Capability Information field to 1 if the STA's dot11SpectrumManagementRequired is true; otherwise, it shall be set to 0.

~~Bits 8 – 15 of the Capability Information field are reserved.~~ STAs set the QoS subfield to 1 with the capability information field when the MIB attribute dot11QosOptionImplemented is true and set it to 0 otherwise.

STAs shall set the Short Slot Time subfield to 1 in transmitted Association Request, ~~and Reassociation Request,~~ DLP request and response MMPDUs when the MIB attribute dot11ShortSlotTimeOptionImplemented and dot11ShortSlotTimeOptionEnabled are true. Otherwise, the STA shall set the Short Slot Time subfield to 0 in transmitted Association Request and Reassociation Request MMPDUs.

If a STA that does not support Short Slot Time associates, the AP shall use long slot time beginning at the first Beacon subsequent to the association of the long slot time STA. APs shall set the Short Slot Time subfield in transmitted Beacon, Probe Response, Association Response, and Reassociation Response MMPDUs to indicate the currently used slot time value within this BSS.

STAs shall set the MAC variable aSlotTime to the short slot value upon transmission or reception of Beacon, Probe Response, Association Response, and Reassociation Response MMPDUs from the BSS that the STA has joined or started and that have the short slot subfield set to 1 when the MIB attribute dot11ShortSlotTimeOptionImplemented is true. STAs shall set the MAC variable aSlotTime to the long slot value upon transmission or reception of Beacon, Probe Response, Association Response, and Reassociation Response MMPDUs from the BSS that the STA has joined or started and that have the short slot subfield set to 0 when the MIB attribute dot11ShortSlotTimeOptionImplemented is true. STAs shall set the MAC variable aSlotTime to the long slot value at all times when the MIB attribute dot11ShortSlotTimeOptionImplemented is false. When the dot11ShortSlotTimeOptionImplemented MIB attribute is not present, or when the PHY supports only a single slot time value, then the STA shall set the MAC variable aSlotTime to the slot value appropriate for the attached PHY.

For IBSS, the Short Slot Time subfield shall be set to 0.

APs set the APSD subfield to 1 within the capability information field when the MIB attribute dot11APSDOptionImplemented is true and set it to 0 otherwise. STAs always set this subfield to 0.

APs as well as STAs in IBSSs shall set the DSSS-OFDM subfield to 1 in transmitted Beacon, Probe Response, Association Response, and Reassociation Response management MMPDUs to indicate that the use of DSSS-OFDM, as described in 19.7, is allowed within this BSS. To indicate that the use of DSSS-OFDM is not allowed, the DSSS-OFDM subfield shall be set to 0 in Beacon, Probe Response, Association Response, and Reassociation Response MMPDUs transmitted within the BSS.

STAs shall set the DSSS-OFDM subfield to 1 in transmitted Association Request, ~~and~~ Reassociation Request, DLP request and response MMPDUs when the MIB attribute dot11DSSS-OFDMOptionImplemented and dot11DSSS-OFDMOptionEnabled are true. Otherwise, STAs shall set the DSSS-OFDM subfield to 0 in transmitted Association Request and Reassociation Request MMPDUs.

STAs set the Block Ack subfield to 1 within the capability information field when the MIB attribute dot11BlockOptionImplemented is true and set it to 0 otherwise.

Unused bits of the Capability Information field are reserved.

7.3.1.7 Reason Code field

Insert the Reason codes shown in the following table into Table 18 replacing previously reserved values:

<u>32</u>	<u>Disassociated for unspecified, QoS-related reason</u>
<u>33</u>	<u>Disassociated because QAP lacks sufficient bandwidth for this QSTA</u>
<u>34</u>	<u>Disassociated because of excessive number of frames that need to be acknowledged, but are not acknowledged for AP transmissions and/or poor channel conditions</u>
<u>35</u>	<u>Disassociated because QSTA is transmitting outside the limits of its TXOPs</u>

7.3.1.9 Status Code field

Insert the Status codes shown in the following table into Table 19 replacing previously reserved values:

<u>32</u>	<u>Unspecified, QoS-related failure</u>
<u>33</u>	<u>Association denied due to QAP having insufficient bandwidth to handle another QSTA</u>
<u>34</u>	<u>Association denied due to excessive frame loss rates and/or poor conditions on current operating channel</u>
<u>35</u>	<u>Association (with QBSS) denied due to requesting station not supporting the QoS facility</u>
<u>36</u>	<u>Association denied due to requesting station not supporting Block Ack</u>
<u>37</u>	The request has been declined.
<u>38</u>	The request has not been successful as one or more parameters have invalid values.
<u>39</u>	The TS has not been created because the request cannot be honored. However, a suggested TSPEC is provided so that the initiating QSTA may attempt to set another TS with the suggested changes to the TSPEC.
<u>40</u>	The TS has not been created. However, the HC may be capable of creating a TS, in response to a request, after the time indicated in the TS Delay element.
<u>41</u>	Direct Link is not allowed in the BSS by policy
<u>42</u>	Destination STA is not present within this QBSS.
<u>43</u>	The Destination STA is not a QSTA.

7.3.1.11 Action field

Change the second paragraph in the text as shown:

The Category field ~~shall be~~ is set to one of the non-reserved values shown in Table 19a. Action frames of a given category are referred to as <category name> Action frames. For example, frames in the "QoS" category are called "QoS Action frames".

If a STA receives a unicast Action frame with an unrecognized Category field or some other syntactic error and the most significant bit of the Category field set equal to 0 then the STA returns the Action frame to the source without change except that the most significant bit of the Category field ~~shall be~~ is set equal to 1.

Insert the following three rows into Table 19a:

Code	Meaning	See subclause
1	QoS	7.4.2
2	DLP	7.4.3
3	Block Ack	7.4.4

Insert the following sentence at the end of the subclause:

The following subclauses detail additional fixed fields used within the action field.

7.3.1.12 Dialog Token field

The Dialog Token field is used for matching action responses with action requests when there are multiple, concurrent action requests. The length of the Dialog Token field is 1 octet. The Dialog Token field is illustrated in Figure 33.2. The value of the Dialog Token field in a request action frame is arbitrary. The value of the Dialog Token field in a response frame is copied from each request action frame.



Octets: 1

Figure 33.2 – Dialog Token fixed field

7.3.1.13 DLP Timeout Value

The DLP Timeout Value field is used in the DLP request frame to indicate the timeout value for the direct link. The length of the DLP Timeout Value is 2 octets. The DLP Timeout Value field is illustrated in Figure 33.4.

B0
B15
DLP Timeout Value
 Octets: 2

Figure 33.4 – DLP Timeout Value fixed field

The DLP Timeout Value contains the duration, in seconds, after which the direct link is terminated, if there are no frame exchanges within this duration with the peer. A value of 0 implies that the direct link is never to be terminated based on a timeout.

7.3.1.14 Block Ack Parameter Set field

The Block Ack Parameter Set field is used in ADDBA frames to signal the parameters for setting up a Block Ack. The length of the Block Ack Parameter Set field is 2 octets. The Block Ack Parameter Set field is illustrated in Figure 33.6.

B0
B1
B2
B5
B6
B15

Reserved	Block Ack Policy	TID	Buffer Size
----------	------------------	-----	-------------

 Octets: 2

Figure 33.6 – Block Ack Parameter Set fixed field

The Block Ack Policy subfield is set to 1 for immediate Block Ack and 0 for delayed Block Ack. The Block Ack Policy subfield value assigned by the originator of the QoS Data frames is advisory.

The TID subfield contains the value of the TC or TS for which the Block Ack is being requested.

The Buffer Size indicates the number of buffers of size 2304 octets available for this particular TID.⁶

In an ADDBA request frame, the Buffer Size subfield is intended to provide guidance for the frame receiver to decide its re-ordering buffer size, and is advisory only. If the Buffer Size subfield is set to 0, it implies that the Originator of the Block Ack has no information to specify its value.

In an ADDBA response frame, when the Status Code is set to 0, the Buffer size is set to a value of at least 1.

7.3.1.15 Block Ack Timeout Value

The Block Ack Timeout Value field is used in the ADDBA request frame to indicate the timeout value for Block Ack. The length of the Block Ack Timeout Value is 2 octets. The Block Ack Timeout Value field is illustrated in Figure 33.7.

B0
B15

⁶ For Buffer size, the recipient of data advertises a single scalar number that is the number of maximum-size fragment buffers available. Every buffered MPDU will consume one of these buffers regardless of whether the frame contains a whole MSDU or a fragment of an MSDU. In other words, ten maximum-size unfragmented MSDUs will consume the same amount of buffer space at the recipient as 10 small fragments.



Figure 33.7 – Block Ack Timeout Value fixed field

The Block Ack Timeout is the duration in seconds for which the Block Ack set up is active. If no data frame is received by the recipient or no BlockAck frame is received by the originator the Block Ack is considered to be inactive and is torn down.

7.3.1.16 DELBA Parameter Set field

The DELBA Parameter Set field is used in a DELBA frame to terminate an already set up Block Ack. The length of the DELBA Parameters field is 2 octets. The DELBA Parameters field is illustrated in Figure 33.9.

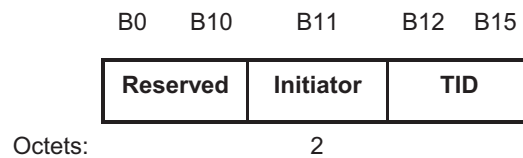


Figure 33.9 – DELBA Parameters fixed field

The Initiator subfield indicates if the originator or the recipient of the data is sending this frame. It is set to 1 to indicate the originator and is set to 0 to indicate the recipient. TID subfield indicates the TSID or the UP for which the Block Ack has been originally set up.

7.3.2 Information elements

Insert the element IDs shown in the following table into Table 20 replacing previously reserved values:

Information Element	Element ID
QBSS Load	11
EDCA Parameter Set	12
Traffic Specification	13
Traffic Classification	14
Schedule	15
TS Delay	43
TCLAS Processing	44
QoS Action	45
QoS Capability	46

Insert the following paragraph below Table 20:

A station that encounters an unknown or reserved element ID value in a management frame received without error shall ignore that element and shall parse any remaining management frame body for additional information elements with recognizable element ID values. The frame body components specified for many management subtypes result in elements ordered by ascending element ID.

Insert the following subclauses with the figures and tables included therein, renumbering as necessary:

7.3.2.13 QBSS Load element

The QBSS Load element contains information on the current station population and traffic levels in the QBSS. The element information field is defined in Figure 46.1. This element may be used by the non-AP QSTA for vendor specific AP selection algorithm when roaming.

Octets: 1	1	2	1	2
Element ID (11)	Length (5)	Station Count	Channel Utilization	Available Admission Capacity

Figure 46.1 – QBSS Load element format

The station count field is interpreted as an unsigned integer that indicates the total number of STAs currently associated with this QBSS.

The channel utilization field is defined as the percentage of time, normalized to 255, the QAP sensed the medium busy, as indicated by either the physical or virtual carrier sense mechanism. This percentage is computed using the formula, $((\text{channel busy time} / (\text{dot11ChannelUtilizationBeaconIntervals} * \text{dot11BeaconPeriod} * 1024)) * 255)$, where ‘channel busy time’ is defined to be the number of microseconds during which the carrier sense mechanism, as defined in 9.2.1, has indicated a channel busy indication, and the MIB attribute dot11ChannelUtilizationBeaconIntervals represents the number of consecutive beacon intervals during which the channel busy time is measured. The default value of dot11ChannelUtilizationBeaconIntervals is defined in Annex D.

The Available Admission Capacity field is 2 octets long and contains an unsigned integer that specifies the remaining amount of medium time available via explicit admission control, in units of 32 microsecond periods per 1 second. The field is helpful for roaming non-AP QSTAs to select a QAP that is likely to accept future admission control requests, but it does not represent a guarantee that the HC will admit these requests.

7.3.2.14 EDCA Parameter Set element

The EDCA Parameter Set element provides information needed by QSTAs for proper operation of the QoS facility during the contention period. The format of the EDCA Parameter Set element is defined in Figure 46.2.

The EDCA Parameter Set element is used by the QAP to establish policy (by changing default MIB attribute values), to change policies when accepting new stations or new traffic, or to adapt to changes in offered load. The most recent EDCA parameter set element received by a QSTA is used to update the appropriate MIB values.

Octets: 1	1	1	1	4	4	4	4
Element ID (12)	Length (18)	QoS Info	Reserved	AC_BE Parameters Record	AC_BK Parameters Record	AC_VI Parameters Record	AC_VO Parameters Record

Figure 46.2 – EDCA Parameter Set element

The format of the QoS Info field is illustrated in Figure 46.15.1. The QoS Info field contains the EDCA Parameter Set Update Count, which is initially set to 0 and is incremented each time any of the AC parameters changes. This field can be used by QSTAs to determine whether the EDCA parameter set has changed and requires updating the appropriate MIB values. This field is used by QSTAs to determine whether the EDCA parameter set has changed and requires updating the appropriate MIB attributes.

The formats of AC_BE, AC_BK, AC_VI and AC_VO Parameter Record fields are identical and are illustrated in Figure 46.4.

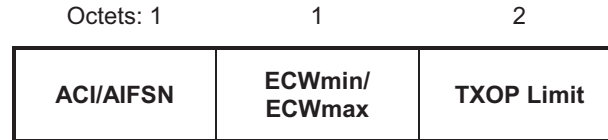


Figure 46.4 – AC_BE, AC_BK, AC_VI and AC_VO Parameter record field formats

The format of the ACI/AIFS field is illustrated in Figure 46.5.

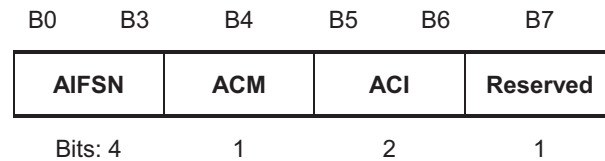


Figure 46.5 – ACI/AIFS Field

The value of the ACI references the AC to which all parameters in this record correspond. The mapping between AC index (ACI) and AC is defined in Table 20.1. The ACM (Admission Control Mandatory) subfield indicates that Admission Control is required for the AC. If ACM subfield is set to 0, then there is no admission control for the corresponding AC. If ACM subfield is set to 1, admission control is used prior to transmission for the corresponding AC. The AIFS subfield indicates the number of slots after a SIFS duration a non-AP QSTA should defer before either invoking a backoff or starting a transmission. The minimum value for AIFS is 2.

Table 20.1 – ACI to AC coding

ACI	AC	Access Category
00	AC_BE	Best Effort
01	AC_BK	Background
10	AC_VI	Video
11	AC_VO	Voice

The ECWmin/ECWmax fields are illustrated in Figure 46.6.

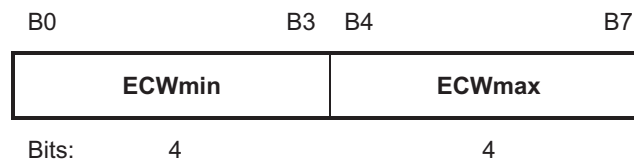


Figure 46.6 – ECWmin/ECWmax field

The fields ECWmin and ECWmax encode the values of CWmin and CWmax respectively in an exponent form. The values ECWmin and ECWmax are defined such that:

$$CW_{min} = 2^{ECW_{min}} - 1$$

$$CW_{max} = 2^{ECW_{max}} - 1$$

Hence the minimum encoded value of CWmin and CWmax is 0, and the maximum value is 32767.

The value of TXOP limit is specified as an unsigned integer, with the least significant octet transmitted first, in units of 32 microseconds. A TXOP limit value of 0 indicates that a single MSDU or MMPDU, in addition to a possible RTS/CTS exchange or CTS to itself, may be transmitted at any rate for each TXOP.

The default values used by non-AP QSTAs for the parameters in EDCA Parameter set are defined in Table 20.2⁷.

Table 20.2 – Default EDCA Parameter Set

<u>AC</u>	<u>CWmin</u>	<u>CWmax</u>	<u>AIFS_N</u>	<u>TXOP Limit</u>		
				<u>DS-CCK</u> ⁸	<u>Extended Rate /OFDM</u> ⁹	<u>Other PHYs</u>
<u>AC_BK</u>	<u>aCWmin</u>	<u>aCWmax</u>	<u>7</u>	<u>0</u>	<u>0</u>	<u>0</u>
<u>AC_BE</u>	<u>aCWmin</u>	<u>aCWmax</u>	<u>3</u>	<u>0</u>	<u>0</u>	<u>0</u>
<u>AC_VI</u>	$\frac{(aCWmin+1)/2 - 1}{1}$	<u>aCWmin</u>	<u>2</u>	<u>6.016ms</u>	<u>3.008ms</u>	<u>0</u>
<u>AC_VO</u>	$\frac{(aCWmin+1)/4 - 1}{1}$	$\frac{(aCWmin+1)/2 - 1}{1}$	<u>2</u>	<u>3.264ms</u>	<u>1.504ms</u>	<u>0</u>

7.3.2.15 Traffic Specification (TSPEC) element

The Traffic Specification (TSPEC) element contains the set of parameters that define the characteristics and QoS expectations of a traffic flow, in the context of a particular non-AP QSTA, for use by the HC and non-AP QSTA(s) in support of QoS traffic transfer using the procedures defined in 11.4. The element information field comprises the items as defined below and the structure is defined in Figure 46.7.

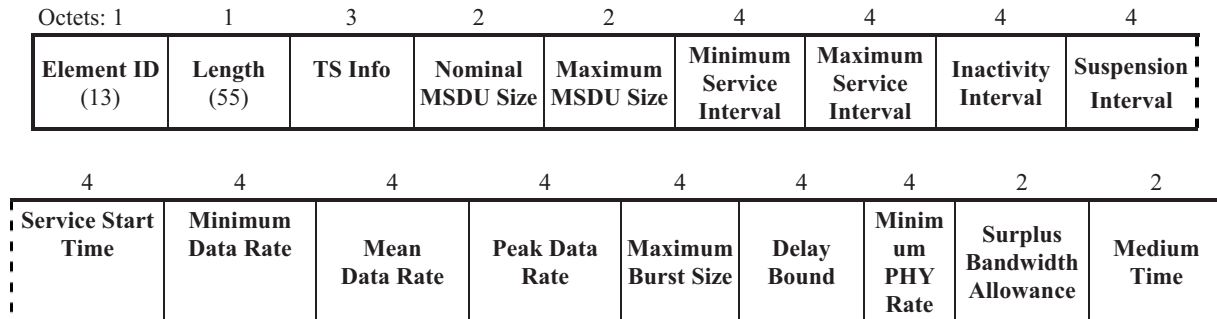


Figure 46.7 – Traffic Specification element format

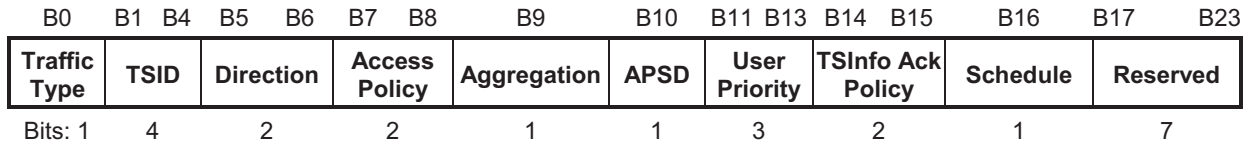
The Traffic Specification allows a set of parameters more extensive than may be needed, or may be available, for any particular instance of parameterized QoS traffic. Unless indicated otherwise, for fields that follow the TSInfo field, the fields are set to zero for any unspecified parameter values. Non-AP QSTAs set the value of any parameters to "unspecified" if they have no information for setting that parameter. The HC may change the value of parameters that have been set "unspecified" by the QSTA to any value that it deems appropriate, including leaving them "unspecified".

⁷ The default values for TXOP limit are expressed in milliseconds and are multiples of 32 microseconds.

⁸ Refers to the PHYs in clauses 15 and 18.

⁹ Refers to the PHYs in clauses 17 and 19.

1 The structure of the TS Info field is defined in Figure 46.8.



2 **Figure 46.8 – TS Info field**

3 The Traffic Type subfield is a single bit and is set to 1 for a periodic traffic pattern (e.g. isochronous traffic
 4 stream of MSDUs, with constant or variable sizes, that are originated at fixed rate), or is set to 0 for an
 5 aperiodic, or unspecified traffic pattern (e.g. asynchronous traffic stream of low-duty cycles).

6 The TSID subfield is 4 bits in length and contains the TSID values in the format defined in 7.1.3.5.1. The
 7 combination of TSID and Direction identify the traffic stream, in the context of the non-AP QSTA, to which
 8 the traffic specification applies. The same TSID may be used for multiple traffic streams at different non-AP
 9 QSTAs. A non-AP QSTA may use the TSID value for a downlink TSPEC and either an uplink or a direct link
 10 TSPEC at the same time. A non-AP QSTA shall not use the same TSID for both uplink and direct link TS. A
 11 bidirectional link request is equivalent to a downlink traffic stream and an uplink traffic stream, each with the
 12 same TSID and parameters.

13 The Direction field specifies the direction of Data carried by the traffic stream as defined in Table 20.3.

14 **Table 20.3 – Direction field encoding**

Bit 5	Bit 6	Usage
0	0	Uplink (non-AP QSTA to HC)
1	0	Downlink (HC to non-AP QSTA)
0	1	Direct link (non-AP QSTA to non-AP QSTA)
1	1	Bidirectional link (equivalent to a downlink request plus an uplink request, each direction having the same parameters) The fields in the TSPEC element specify resources for a single direction. Double the specified resources are required to support both streams.

15

16 The access policy subfield is 2 bits in length and specifies the access that would be used for the traffic stream
 17 and is defined in Table 20.4.

18 **Table 20.4 – Access Policy**

Bit 7	Bit 8	Usage
0	0	Reserved
1	0	Contention-based channel access (EDCA)
0	1	Controlled channel access (HCCA)

1	1	Both controlled and contention-based channel access.
---	---	--

The Aggregation subfield is 1 bit in length. It is set to 1 by a non-AP QSTA to indicate that an aggregate schedule is required. It is set to 1 by the QAP if an aggregate schedule is being provided to the non-AP QSTA. It is set to 0 otherwise.

The APSD subfield is a single bit and is set to 1 to indicate that Automatic Power Save Delivery is to be used for the traffic associated with the TSPEC. The APSD subfield is set to 0 if the station does not engage in power saving. When the APSD subfield is set to 1, the aggregation subfield is also set to 1. If the APSD field values of two admitted traffic streams associated with the same QSTA differ, the QSTA is assumed to use APSD mechanism until termination of the traffic stream with a TSPEC with the APSD bit set to 1.

The User priority subfield is 3 bits and indicates the actual value of the UP to be used for the transport of MSDUs belonging to this traffic stream in cases where relative prioritization is required. When the TCLAS element is present in the request, the user priority subfield in TSInfo field of the TSPEC element is reserved.

The TSInfo Ack Policy subfield is 2 bits in length and indicates whether MAC acknowledgements are required for MPDUs belonging to this TID, and the desired form of those acknowledgements. The encoding of the TSInfo Ack Policy field is shown in Table 20.5. If the TS Info Ack Policy is set to Block Acknowledgement, and the type of Block Acknowledgement policy is unknown to the HC, the HC shall assume, for TXOP scheduling, that the Immediate Block Ack policy is being used (see 9.10).

Table 20.5 – TSInfo Ack Policy field encoding

Bit 2	Bit 3	Usage
0	0	Normal IEEE 802.11 [®] acknowledgement. The addressed recipient returns an ACK or QoS (+)CF-Ack frame after a SIFS period, according to the procedures defined in 9.2.8, 9.3.3 and 9.9.2.3.
1	0	No acknowledgement The recipient(s) do not acknowledge the transmission.
0	1	Reserved
1	1	Block Acknowledgement A separate Block Ack set up mechanism described in 9.10 is used.

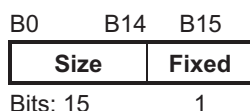
The schedule subfield is 1 bit in length and specifies the requested type of schedule. The setting of the subfield when the Access Policy is EDCA is shown in Table 20.6. When the Access Policy is set to any value other than EDCA, the schedule subfield is reserved.

Table 20.6 – Setting of Schedule subfield

APSD	Schedule	Usage
0	0	No Schedule
1	0	Unscheduled SP
0	1	Scheduled SP

1	1	Scheduled APSD
---	---	----------------

- 1 The Nominal MSDU Size field is 2 octets long and contains an unsigned integer that specifies the nominal size, in octets, of MSDUs belonging to the TS under this traffic specification and is defined in Figure 46.9. If
 2 the Fixed subfield is set to 1, then the size of the MSDU is fixed and is indicated by the Size Subfield. If the
 3 Fixed subfield is set to 0, then the size of the MSDU might not be fixed and the Size indicates the nominal
 4 MSDU size. If both Fixed Subfield and Size are set to 0, then the nominal MSDU size is unspecified.
 5



6 **Figure 46.9 – Nominal MSDU Size Field**

- 7 The Maximum MSDU Size field is 2 octets long and contains an unsigned integer that specifies the maximum
 8 size, in octets, of MSDUs belonging to the TS under this traffic specification.
- 9 The Minimum Service Interval field is 4 octets long and contains an unsigned integer that specifies the
 10 minimum interval, in units of microseconds, between the start of two successive SPs.
- 11 The Maximum Service Interval field is 4 octets long and contains an unsigned integer that specifies the
 12 maximum interval, in units of microseconds, between the start of two successive SPs.
- 13 The Inactivity Interval field is 4 octets long and contains an unsigned integer that specifies the maximum
 14 amount of time in units of microseconds that may elapse without arrival or transfer of an MSDU belonging to
 15 the TS before this TS is deleted by the MAC entity at the HC.
- 16 The Suspension Interval field specifies the maximum amount of time in units of microseconds that may elapse
 17 without arrival or transfer of an MSDU belonging to the TS before the generation of successive QoS(+)CF-
 18 Poll is stopped for this TS. A value of 4294967295 ($= 2^{32} - 1$) disables the Suspension Interval, indicating
 19 that polling for the TS is not to be interrupted based on inactivity.
- 20 The Service Start Time field is 4 octets and indicates the time, expressed in microseconds, when the SP starts.
 21 The Service Start Time indicates to QAP the time when a non-AP QSTA first expects to be ready to send
 22 frames and a power-saving non-AP QSTA will be awake to receive frames. This may help the QAP to
 23 schedule service so that the MSDUs encounter small delays in the MAC and the power-saving non-AP
 24 QSTAs to reduce power consumption. The field represents the four lower-order bytes of the TS timer at the
 25 start of the SP. If APSD subfield is set to 0, this field is also set to 0 (unspecified).
- 26 The Minimum Data Rate field is 4 octets long and contains an unsigned integer that specifies the lowest data
 27 rate specified at the MAC_SAP, in units of bits per second, for transport of MSDUs belonging to this TS
 28 within the bounds of this traffic specification. The Minimum Data Rate does not include the MAC and PHY
 29 overheads incurred in transferring the MSDUs.
- 30 The Mean Data Rate field is 4 octets long and contains an unsigned integer that specifies the average data rate
 31 specified at the MAC_SAP, in units of bits per second, for transport of MSDUs belonging to this TS within
 32 the bounds of this traffic specification. The Mean Data Rate corresponds to the rate of the second token
 33 bucket in a twin token bucket [B6–B8]¹⁰ based traffic policer. The Mean Data Rate does not include the MAC
 34 and PHY overheads incurred in transferring the MSDUs.
- 35 The Peak Data Rate field is 4 octets long and contains an unsigned integer that specifies the maximum
 36 allowable data rate in units of bits/second, for transfer of the MSDUs belonging to this TS within the bounds
 37 of this traffic specification. The Peak Data Rate corresponds to the rate of the first token bucket in a twin

¹⁰ The token bucket model provides standard terminology for describing the behavior of a traffic source. The TSPEC parameters defined above have an analogy to the parameters of the twin token bucket implementation. The analogy is used for clarification, but it is left to the implementer to use any traffic policer. The token bucket model is described in IETF RFC 2215[B5].

1 token bucket based traffic policer. If “p” is the peak rate in bit/s, then the maximum amount of data,
 2 belonging to this TS, arriving in any time interval $[t_1, t_2]$, where $t_1 < t_2$ and $t_2 - t_1 > 1$ TU, must not exceed $p \cdot (t_2 -$
 3 $t_1)$ bits.

4 The Maximum Burst Size field is 4 octets long and contains an unsigned integer that specifies the maximum
 5 burst, in units of octets, of the MSDUs belonging to this TS that arrive at the MAC SAP at the peak data rate.
 6 This corresponds to the second token bucket size in a twin token bucket based traffic policer. A value of 0
 7 indicates that there are no bursts.

8 The Delay Bound field is 4 octets long and contains an unsigned integer that specifies the maximum amount
 9 of time, in units of microseconds, allowed to transport an MSDU belonging to the TS in this TSPEC,
 10 measured between the time marking the arrival of the MSDU at the local MAC sublayer from the local MAC
 11 SAP and the time of completion of the successful transmission or retransmission of the MSDU to the
 12 destination. The completion of the MSDU transmission includes the relevant acknowledgement frame
 13 transmission time, if present.

14 The Minimum PHY Rate field is 4 octets long and contains an unsigned integer that specifies the desired
 15 minimum PHY rate to use for this TS, in units of bits per second, that is required for transport of the MSDUs
 16 belonging to the TS in this TSPEC.¹¹

17 The Surplus Bandwidth Allowance Factor field is 2 octets long and specifies the excess allocation of time
 18 (and bandwidth) over and above the stated application rates required to transport an MSDU belonging to the
 19 TS in this TSPEC. This field is represented as an unsigned binary number. The thirteen least significant bits
 20 indicate the decimal part while the three most significant bits indicate the integer part of the number.. This
 21 field takes into account retransmissions, as the rate information does not include retransmissions. It represents
 22 the ratio of over-the-air bandwidth (time that the scheduler allocates for the transmission of MSDUs at the
 23 required rates) to bandwidth of the transported MSDUs required for successful transmission (time that would
 24 be necessary at the minimum PHY rate if there were no errors on the channel) to meet throughput and delay
 25 bounds under this TSPEC, when specified. As such, it should be greater than unity.

26 The Medium Time is a 16-bit unsigned integer and contains the amount of time admitted to access the
 27 medium, in units of 32 microsecond periods per second. This field is reserved in the ADDTS Request frame
 28 and is set by the HC in the ADDTS Response frame. The derivation of this field is described in annex H.1.2.
 29 This field is not used for controlled channel access.

30 The User Priority, Minimum Data Rate, Mean Data Rate, Peak Data Rate, Maximum Burst Size, Minimum
 31 PHY Rate, and Delay Bound fields in a TSPEC express the QoS expectations requested by a non-AP QSTA,
 32 if this TSPEC was issued by that non-AP QSTA, or provided by the HC if this TSPEC was issued by the HC,
 33 when these fields are specified with non-zero values. Unspecified parameters in these fields as indicated by a
 34 zero value indicate that the non-AP QSTA does not have specific requirements for these parameters if the
 35 TSPEC was issued by that non-AP QSTA, or that the HC does not provide any specific values for these
 36 parameters if the TSPEC was issued by the HC.

37 7.3.2.16 Traffic Classification (TCLAS) Element

38 The Traffic Classification (TCLAS) element specifies an information element that contains a set of
 39 parameters necessary to identify incoming MSDUs (from a higher layer in all QSTAs or from the DS in an
 40 AP) with a particular traffic stream to which they belong. The structure of this element is shown in figure
 41 46.10.

Octets: 1	1	1	L
Element ID (14)	Length (L+1)	User Priority	Frame Classifier

¹¹ This rate information is intended to ensure that the TSPEC parameter values resulting from an admission control negotiation are sufficient to provide the required throughput for the traffic stream. In a typical implementation, a TS is admitted only if the defined traffic volume can be accommodated at the specified rate within an amount of WM occupancy time that the admissions control entity is willing to allocate to this TS.

Figure 46.10 –TCLAS element format

The User Priority field contains the value of the UP of the associated MSDUs. The Frame Classifier field is 3-255 octets in length and is defined in Figure 46.10.1.

Octets: 1	1	1-253
Classifier Type	Classifier Mask	Classifier Parameters

Figure 46.10.1 – Frame Classifier

The Frame Classifier field comprises a Classifier Type, a Classifier Mask, and a set of Classifier Parameters. The Classifier Type is one octet in length and specifies the type of Classifier Parameters in this TCLAS as defined in Table 20.7. Three classifier types are defined below.

Table 20.7 – Frame classifier type

Classifier Type	Classifier Parameters
0	Ethernet parameters
1	TCP/UDP IP parameters
2	IEEE 802.1D/Q Parameters
3 – 255	Reserved

The Classifier Mask is Classifier Type dependent in length and specifies a bitmap wherein bits that are set to 1 identifies a subset, out of the set of the Classifier Parameters whose values must match those of the corresponding parameters in an given MSDU for that MSDU to be classified to the TS of the affiliated TSPEC. The bitmap is ordered from the least to the most significant bit, with each bit pointing to one of the following Classifier Parameters of the same relative position. An incoming MSDU that failed to be classified to a particular TS may be classified to another activated TS based on the Frame Classifier for that TS, unless all the Frame Classifiers for the activated TS have been exhausted—in which case the MSDU does not belong to any activated TS and is classified to be a best effort MSDU. In cases where there are more bits in the bitmap than Classifier Parameters that follow, those most significant bits that do not point to any Classifier Parameters are reserved.

For Classifier Type 0, the Classifier Parameters are the following parameters contained in an Ethernet packet header: Source Address, Destination Address, and Type. The Frame Classifier field for Classifier Type 0 is defined in Figure 46.10.2. It has a length of 16 octets.

Octets: 1	1	6	6	2
Classifier Type (0)	Classifier Mask	Source Address	Destination Address	Type

Figure 46.10.2 – Frame Classifier of Classifier Type 0 Field

For Classifier Type 1, frame classifier is defined for both IPv4 and IPv6, shown in figures 46.10.3 and 46.10.4, and distinguished by the Version field. The Classifier Parameters are the following parameters contained in a TCP or UDP header: Source Address, Destination Address, Source Port Destination Port and Version plus the following parameters in an IPv4 header: Differentiated Services Code Point (DSCP) [B9] and Protocol, or the following parameter in an IPv6 header: Flow Label. The DSCP field will contain the value in the least significant six bits and the two most significant bits are set to 0.

Octets: 1	1	1	4	4	2	2	1	1	1
-----------	---	---	---	---	---	---	---	---	---

Classifier Type (1)	Classifier Mask	Version	Source IP Address	Destination IP Address	Source Port	Destination Port	DSCP	Protocol	Reserved
---------------------	-----------------	---------	-------------------	------------------------	-------------	------------------	------	----------	----------

Figure 46.10.3 – Frame Classifier of Classifier Type 1 Field for traffic over IPv4

Octets: 1 1 1 16 16 2 2 3

Classifier Type (1)	Classifier Mask	Version	Source IP Address	Destination IP Address	Source Port	Destination Port	Flow Label
---------------------	-----------------	---------	-------------------	------------------------	-------------	------------------	------------

Figure 46.10.4 – Frame Classifier of Classifier Type 1 Field for traffic over IPv6

For Classifier Type 2, the Classifier Parameters are the following parameters in an IEEE 802.1Q tag header: IEEE 802.1D User Priority and IEEE 802.1Q VLAN ID. The Frame Classifier field for Classifier Type 2 is defined in figure 46.10.5.

Octets: 1 1 2

Classifier Type (2)	Classifier Mask	802.1Q Tag Type
---------------------	-----------------	-----------------

Figure 46.10.5 – Frame Classifier of Classifier Type 2 Field

7.3.2.17 TS Delay

The TS Delay element is used in the ADDTS Response frame transmitted by the HC to a non-AP QSTA and indicates the time after which the ADDTS may be retried. The TS Delay element is defined in Figure 46.11.

Octets: 1 1 4

Element ID (43)	Length (4)	Delay
-----------------	------------	-------

Figure 46.11 – TS Delay

The Delay field, is 4 octets long, and specifies the amount of time, in TUs, a non-AP QSTA should wait before it reinitiates setup of a traffic stream. When set to 0, this field indicates that the non-AP QSTA were never to attempt setting up of a traffic stream.

7.3.2.18 TCLAS Processing

The TCLAS Processing element is present in the ADDTS Request and Response frames if there are multiple TCLAS associated with the request. It indicates how an MSDU received from higher layers should be processed by the classifier. The TCLAS Processing element is defined in Figure 46.12.

Octets: 1 1 1

Element ID (44)	Length (1)	Processing
-----------------	------------	------------

Figure 46.12 – TCLAS Processing

The Processing subfield is 1 octet long. It is set to 0 if the incoming MSDUs parameters' have to match to the parameters in all the associated TCLAS elements and is set to 1 if the incoming MSDU parameters' have to match to at least one associated TCLAS elements.

7.3.2.19 Schedule Element

The schedule element is transmitted by the HC to a non-AP QSTA to announce the schedule that the HC/QAP follows for admitted streams originating from or destined to that non-AP QSTA in future. The information in this element may be used by the non-AP QSTA for power management, internal scheduling or for any other purpose. The element information field is shown in Figure 46.13.

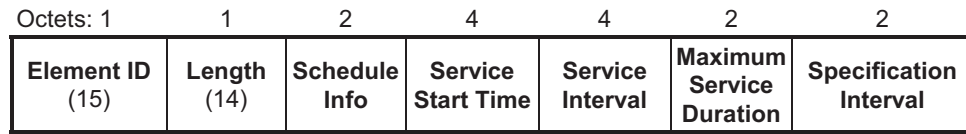


Figure 46.13 – Schedule Element

The Schedule Info Field is shown in Figure 46.14.

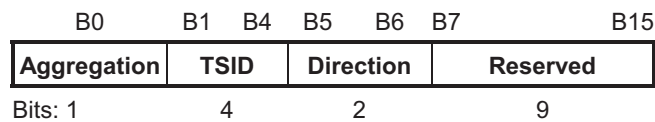


Figure 46.14 – Schedule Info Field

The aggregation subfield is set to 1 if the schedule is an aggregate schedule for all TSID associated with the non-AP QSTA to which the frame is directed. It is set to zero otherwise. The TSID subfield is as defined in clause 7.1.3.5.1, and indicates the TSID for which this schedule applies. The Direction subfield is as defined in clause 7.3.2.15 and defines the direction of the TSPEC associated with the schedule. The TSID and Direction subfields are only valid when the Aggregation subfield is set to 0. If the Aggregation subfield is set to 1, the TSID and Direction subfields are reserved.

The Service Start Time field is 4 octets and indicates the anticipated time, expressed in microseconds, when service starts and represents the lower order four bytes of the TSF timer value at the start of the first SP. The Service Start Time is unspecified if the APSD subfield is set to 0 in the corresponding TSPEC element. If the APSD subfield is set to 1 in the corresponding TSPEC element, the QAP uses this field to confirm or modify the Service Start Time indicated in the TSPEC request.

The Service Interval field is 4 octets and indicates the time, expressed in microseconds, between two successive SPs, and represents the measured time from the start of one service period to the start of the next service period.

The Maximum Service Duration field is 2 octets long and contains an unsigned integer that specifies the maximum duration of an SP in units of microseconds.

The Specification Interval field is 2 octets long and contains an unsigned integer that specifies the time interval in units of TUs to verify schedule conformance.

7.3.2.20 QoS Capability element

The QoS Capability element contains a number of subfields that are used to advertise optional QoS capabilities at a QSTA. The QoS Capability element is present in beacons that do not contain the EDCA Parameter Set element, and in (re) association request frames. The QoS Capability element is defined in Figure 46.15.

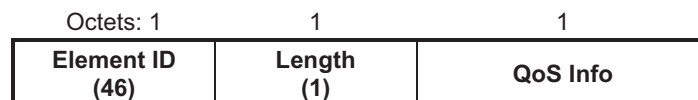


Figure 46.15 – QoS Capability element format

The QoS Info field is 1 octet in length, and contains capability information bits as defined in Figure 46.15.1.

B0	B1	B2	B3	B4	B7
Q-Ack	Queue Request	TXOP Request	More Data Ack	EDCA Parameter Set Update Count	

Figure 46.15.1 – QoS Info field

QSTAs set the Q-Ack subfield to 1 when the MIB attribute dot11QAckOptionImplemented is true and set it to 0 otherwise.

QAPs set the Queue Request subfield to 1 if they can process a non-zero queue size subfield in QoS Control in frames of subtype QoS data and set it to 0 otherwise. For non-AP QSTAs, this field is reserved and is set to 0.

QAPs set the TXOP Request subfield to 1 if they can process non-zero TXOP duration requested subfield in QoS Control in frames of subtype QoS data and set it to 0 otherwise. For non-AP QSTAs, this field is reserved and is set to 0.

Non-AP QSTAs set the More Data Ack subfield to 1 to indicate that they can process Ack frames with the More Data bit in the frame control set to 1 and will remain in the Awake state. QSTAs set the More Data Ack subfield to 0 otherwise. QAPs always set this subfield to 0.

In the beacon frame, the EDCA Parameter Set Update Count subfield is initially set to 0 and is incremented every time any of the AC parameters changes. In Association Request frames, Re-association Request frames the EDCA Parameter Set Update count subfield is set to 0.

7.3.2.21 QoS Action Element

The QoS Action element contains the fixed fields of a QoS Action as described in 7.4.2. The QoS Action element provides an alternative method to sending the action within its own action frame. The QoS Action element may be included in (Re)Association Request and (Re)Association Response frames in order to activate various facilities at association time. Any other elements associated with the action directly follow the QoS Action element, as specified by the definition of the particular action in 7.4.2. The element information field is defined in Figure 46.16.

Octets: 1	1	1	L-1
Element ID (45)	Length (L)	QoS Action (see Table 20.8)	Fixed fields of QoS Action Body

Figure 46.16 – QoS Action element format**7.4 Action frame format details**

Insert the following subclauses after subclause 7.4.1.

7.4.2 QoS action frame details

Several Action frame formats are defined for QoS purposes. The Action field values, associated with each frame format within the QoS Category, are defined in Table 20.8.

Table 20.8 – QoS Action field values

Action field value	Meaning
0	ADDTS request
1	ADDTS response
2	DELTS
3	Schedule
4– 255	Reserved

7.4.2.1 ADDTS request frame format

The ADDTS frames are used to carry TSPEC and optionally TCLAS Elements to set up and maintain traffic streams using the procedures defined in 11.5.

The frame body of the ADDTS request frame contains the information shown in Table 20.9.

Table 20.9 – ADDTS request frame body

Order	Information
1	Category
2	Action
3	Dialog Token
4	TSPEC
5 – n	TCLAS (optional)
n + 1	TCLAS Processing (optional)

The Category field is set to 1 (representing QoS).

The Action field is set to 0 (representing ADDTS request).

The Dialog Token, TCLAS and TCLAS Processing fields of this frame are contained in an MLME-ADDTS.request primitive that causes the frame to be sent. Some of the TSPEC parameters are contained in the MLME-ADDTS.request while the other parameters, Surplus Bandwidth Allowance, Minimum Service Interval, Maximum Service Interval, and Minimum PHY rate, are generated within the MAC.

The TSPEC element, defined in 7.3.2.15, and the optional TCLAS Element, defined in 7.3.2.16, contains the QoS parameters that define the TS. The TS is identified by the TSID and Direction fields within the TSPEC. The TCLAS is optional at the discretion of the non-AP QSTA that sends the ADDTS Request frame, regardless of the setting of the Access Policy (EDCA or HCCA). There may be one or more TCLAS elements in the ADDTS frame. The TCLAS Processing element is present when there is more than one TCLAS elements and is defined in 7.3.2.18.

7.4.2.2 ADDTS response frame format

The ADDTS response frame is transmitted in response to an ADDTS request frame. The frame body of the ADDTS response frame contains the information shown in Table 20.10.

Table 20.10 – ADDTS response frame body

Order	Information
1	Category
2	Action
3	Dialog Token
4	Status Code
5	TS Delay
6	TSPEC
7 – n	TCLAS (optional)
n + 1	TCLAS Processing (optional)
n + 2	Schedule

The Category field is set to 1 (representing QoS).

The Action field is set to 1 (representing ADDTS response).

The Status code field is defined in 7.3.1.9.

The Dialog Token, TS Delay, TSPEC, TCLAS and TCLAS Processing in this frame are contained in an MLME-ADDTS.response primitive that causes the frame to be sent.

The HC announces the schedule in the ADDTS response frame if the schedule subfield in the TSPEC Information Element in the corresponding ADDTS request frame is set to 1.

7.4.2.3 DELTS frame format

The DELTS frame is used to delete a traffic stream using the procedures defined in 11.4.7.

The frame Body of a DELTS frame contains the information shown in Table 20.11.

Table 20.11 – DELTS frame body

Order	Information
1	Category
2	Action
3	TSInfo

The Category field is set to 1 (representing QoS).

The Action field is set to 2 (representing DELTS).

The TSInfo field is defined in 7.3.2.15.

A DELTS frame is used to delete a traffic stream characterized by the TSPEC element included in the frame. A DELTS frame may be sent from the HC to the source station of that traffic stream, or vice versa, to indicate an imperative request, to which no response is required from the recipient station.

7.4.2.4 Schedule frame body format

The Schedule frame is transmitted by the HC to a non-AP QSTA to announce the schedule of delivery of data and polls. The frame body of the Schedule Frame contains the information shown in Table 20.12.

Table 20.12 – Schedule frame body

Order	Information
1	Category
2	Action
3	Schedule

The Category field is set to 1 (representing QoS).

The Action field is set to 3 (representing Schedule).

The Schedule element is defined in 7.3.2.19.

7.4.3 Direct Link Protocol (DLP) action frame details

Several Action frame formats are defined for DLP Management purposes. An Action field, in the octet field immediately after the Category field, differentiates the formats. The Action field values, associated with each frame format are defined in Table 20.13.

Table 20.13 – DLP Action field values

Action field value	Meaning
0	DLP request
1	DLP response
2	DLP Teardown
3-255	Reserved

7.4.3.1 DLP request

The DLP request frame is used to set up a direct link with a peer MAC. The frame body of the DLP request frame contains the information shown in Table 20.14.

Table 20.14 – DLP request frame body

Order	Information
1	Category
2	Action
3	Destination MAC Address
4	Source MAC Address
5	Capability Information
6	DLP Timeout Value
7	Supported rates

The Category field is set to 2 (representing DLP).

The Action field is set to 0 (representing DLP request).

The Destination MAC address is the MAC address of the target destination.

The Source MAC address is the MAC address of the initiating STA.

The Capability information is the capability information of the originator of the request.

The DLP Timeout Value field is defined in 7.3.1.13.

The supported rates information element contains the supported rates information of the originator.

7.4.3.2 DLP response

The DLP response frame is sent in response to a DLP request frame. The frame body of a DLP response frame contains the information shown in Table 20.15.

Table 20.15 – DLP response frame body

Order	Information
1	Category
2	Action
3	Status Code
4	Destination MAC Address
5	Source MAC Address
6	Capability Information
7	Supported rates

The Category field is set to 2 (representing DLP).

The Action field is set to 1 (representing DLP response).

The Status Code field is defined in 7.3.1.9.

The Destination MAC address in is and the source MAC address are copied from the corresponding fields in the DLP request frame.

The Capability information is the capability information of the target destination. This information is included only if the DLP result code corresponds to Successful (DLP status code 0).

The supported rates information element contains the supported rates information of the target destination. This information is included only if the DLP result code corresponds to Successful (DLP status code 0).

7.4.3.3 DLP teardown

The DLP teardown frame is sent to terminate a direct link with a peer MAC. The frame body of the DLP Teardown contains the information shown in Table 20.16.

Table 20.16 – DLP Teardown frame body

Order	Information
1	Category
2	Action
3	Destination MAC Address
4	Source MAC Address

The Category field is set to 2 (representing DLP).

The Action field is set to 2 (representing DLP teardown).

The Destination MAC address is the MAC address of the target destination.

The Source MAC address is the MAC address of the initiating STA.

7.4.4 Block Ack action frame details

The ADDBA frames are used to set up Block acknowledgement for a specific TC or TS. The Action field values, associated with each frame format within the Block Ack Category, are defined in Table 20.17.

Table 20.17 – Block Ack Action field values

Action field values	Meaning
0	ADDDBA request
1	ADDDBA response
2	DELBA
3-255	Reserved

7.4.4.1 ADDBA request

An ADDDBA request is sent by an originator of Block Ack to another QSTA. The frame body of an ADDDBA request frame contains the information shown in Table 20.18.

Table 20.18 – ADDBA request frame body

Order	Information
1	Category
2	Action
3	Dialog Token
4	Block Ack Parameter Set
5	Block Ack Timeout Value

The Category field is set to 3 (representing Block Ack).

The Action field is set to 0 (representing ADDDBA request).

The Dialog Token field is set to a non-zero value chosen by the STA.

The Block Ack Parameter Set field is defined in 7.3.1.14.

The Block Ack Timeout value is defined in 7.3.1.15.

7.4.4.2 ADDBA response

The ADDDBA response frame is sent in response to an ADDDBA request frame. The frame body of an ADDDBA response contains the information shown in Table 20.19.

Table 20.19 – ADDBA response frame body

Order	Information
1	Category
2	Action
3	Dialog Token
4	Status code
5	Block Ack Parameter Set
6	Block Ack Timeout Value

- The Category field is set to 3 (representing Block Ack).
- The Action field is set to 1 (representing ADDBA response).
- The dialog token is copied from the corresponding received ADDBA request frame.
- The Status code field is defined in 7.3.1.9.
- The Block Ack Parameter Set field is defined in 7.3.1.14.
- The Block Ack Timeout value field is defined in 7.3.1.15.

7.4.4.3 DELBA

The DELBA frame is sent by either the originator of the traffic or the recipient to terminate the Block Ack participation. The frame body of a DELBA frame format contains the information shown in Table 20.20.

Table 20.20 – DELBA frame body

Order	Information
1	Category
2	Action
3	DELBA Parameter Set

- The Category field is set to 3 (representing DELBA).
 - The Action is set to 2 (representing DELBA).
 - The DELBA Parameters field is defined in 7.3.1.16.
- Insert after 7.4 the following new subclause, including the table therein, renumber items as appropriate:*

7.5 Frame usage

Table 20.21 shows which frame subtypes are transmitted and received by different kinds of MAC entities operating in the different types of BSS and under the available coordination functions.

Table 20.21 – Frame subtype usage by BSS type, MAC entity type, and coordination function

Frame Subtype	IBSS		Infrastructure BSS							
	non-QoS	QoS	non-QoS				QoS			
	CP	CP	CP		CFP		CP		CFP	
	STA	QSTA	STA	AP	STA	AP	QSTA	QAP	QSTA	QAP
(Re) Association Request	---	---	T	R	---	---	T	R	---	---
(Re) Association Response	---	---	R	T	R	T	R	T	R	T
Probe Request	T, R	T, R	T	R	---	---	T	R	---	---
Probe Response	Tbe, R	Tbe, R	R	T	R	T	R	T	R	T
Beacon	Tb, R	Tb, R	R	T	R	T	R	T, R	R	T, R
ATIM	T, R	T, R	---	---	---	---	---	---	---	---
Disassociation	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R
Authentication	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R
Deauthentication	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R
ADDTTS Request	---	---	---	---	---	---	T	R	T	R
ADDTTS Response	---	---	---	---	---	---	R	T	R	T
DELTS	---	---	---	---	---	---	T, R	T, R	T, R	T, R
Schedule	---	---	---	---	---	---	R	T	R	T
DLP Action frames	---	---	---	---	---	---	T, R	T, R	T, R	T, R
Block Ack Action frames	---	T, R	---	---	---	---	T, R	T, R	T, R	T, R
BlockAckReq/BlockAck	---	T, R	---	---	---	---	T, R	T, R	T, R	T, R
PS-Poll	---	---	T	R	---	---	T	R	T	R
RTS	T, R	T, R	T, R	T, R	---	---	T, R	T, R	T, R	T, R
CTS	T, R	T, R	T, R	T, R	---	---	T, R	T, R	T, R	T, R
ACK	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R
CF-End	(R)	(R)	(R)	(R)	R	T	(R)	(R)	R	T
CF-End+CF-Ack	(R)	(R)	(R)	(R)	R	T	(R)	(R)	R	T
Null	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R
Data	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R	T, R
(Data+)CF-Poll+(CF-Ack)	---	---	---	---	R	T	---	---	---	T
(Data+)CF-Ack	---	---	---	---	T, R	T, R	---	---	T, R	T, R
QoS Null	---	T, R	---	---	---	---	T, R	T, R	T, R	T, R
QoS Data	---	T, R	---	---	---	---	T, R	T, R	T, R	T, R
QoS (Data+)CF-Poll	---	---	---	---	---	---	R	T	R	T
QoS (Data+)CF-Poll+CF-Ack	---	---	---	---	---	---	Rq, Rda	Tda, Tq	Rq, Rda	Tda, Tq
QoS (Data+)CF-Ack	---	---	---	---	---	---	T, Rq, Rda	Tda, Tq, R	T, Rq, Rda	Tda, Tq, R

1	Symbols:	
2	T	frame subtype for row is transmitted by MAC entity for column.
3	R	frame subtype for row is received by MAC entity for column.
4	(R)	frame subtype for row is received, but only from other BSSs, by MAC entity for column.
5	Tb	frame subtype for row is transmitted by station that most recently won beacon arbitration.
6	Tbe	frame subtype for row is transmitted by a QSTA in an IBSS pursuant to receiving directed
7		request.
8	Tda	frame subtype for row is transmitted only if recipient of +CF-Ack function is addressee.
9	Rda	frame subtype for row is received if QSTA is addressee.
10	Tq	frame subtype for row is transmitted only if recipient of +CF-Ack function has set the Q-
11		Ack subfield in QoS Capability Element to 1.
12	Rq	frame subtype for row is received if QSTA is not the addressee but has set the Q-Ack
13		subfield in QoS Capability Element to 1.
14	- - -	frame subtype for row is neither received nor transmitted by MAC entity for column.

9. MAC sublayer functional description

Change the text in the introductory paragraph of 9 as shown below:

The MAC functional description is presented in this clause. The architecture of the MAC sublayer, including the distributed coordination function (DCF), the point coordination function (PCF), the hybrid coordination function (HCF), and their coexistence in an IEEE 802.11[®] LAN are introduced in 9.1. These functions are expanded on in 9.2 (DCF), and 9.3 (PCF), and 9.9 (HCF), and a complete functional description of each is provided. Fragmentation and defragmentation are defined covered in 9.4 and 9.5. Multirate support is addressed in 9.6. The allowable frame exchange sequences are defined listed in 9.127. Finally, a number of additional restrictions to limit the cases in which MSDUs are reordered or discarded are described in 9.87. Operation across regulatory domains is defined in 9.98. The Block Ack mechanism is described in 9.10. The No Ack mechanism is described in 9.11.

9.1 MAC architecture

Change the text of the paragraph and Update Figure 47 as shown:

The MAC architecture can be described as shown in Figure 47 as providing the PCF and HCF through the services of the DCF. Note that in a non-QoS STA, HCF is not present. In a QSTA implementation, both DCF and HCF (at least to the extent of EDCA) are present.. PCF is optional in all STAs.

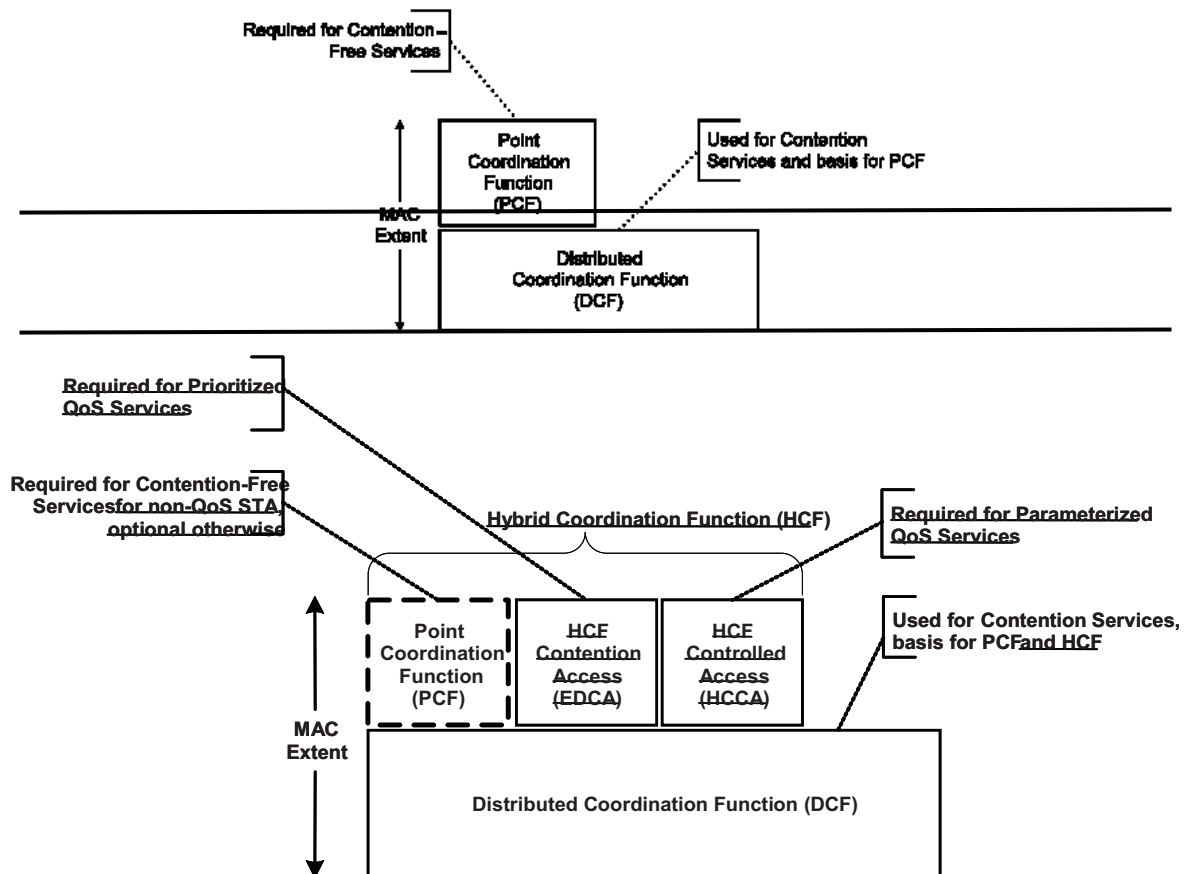


Figure 47 – MAC architecture

Insert after 9.1.2 the following subclause and renumber the current 9.1.3, 9.1.4 and 9.1.5 as 9.1.4, 9.1.5 and 9.1.6, respectively:

9.1.3 Hybrid coordination function (HCF)

The QoS facility includes an additional coordination function called HCF that is only usable in QoS network (QBSS) configurations. The HCF shall be implemented in all QSTAs. The HCF combines functions from the DCF and PCF with some enhanced, QoS-specific mechanisms and frame subtypes to allow a uniform set of frame exchange sequences to be used for QoS data transfers during both the CP and CFP. The HCF uses both a contention-based channel access method, called the enhanced distributed channel access (EDCA) mechanism for contention based transfer and a controlled channel access, referred to as HCF Controlled Channel Access (HCCA) mechanism, for contention free transfer.

QSTAs may obtain transmission opportunities (TXOPs) using one or both of the channel access mechanisms specified in 9.9. If a TXOP is obtained using the contention-based channel access, it is defined as EDCA TXOP. If a TXOP is obtained using the controlled channel access, it is defined as HCCA TXOP. If the TXOP is obtained due to a QoS (+)CF-Poll from the HC, the TXOP is defined as a polled TXOP.

9.1.3.1 HCF contention-based channel access (EDCA)

The EDCA mechanism provides differentiated, distributed access to the WM for QSTAs using 8 different UPs. EDCA mechanism defines four access categories (ACs) that provide support for the delivery of traffic with UPs at the QSTAs. The mapping from UPs to ACs is defined in Table 0.1 in 6.1.1.3.

For each AC, an enhanced variant of the DCF, called a channel access function, contends for TXOPs using a set of EDCA parameters from the EDCA Parameter Set element or from the default values for the parameters when no EDCA Parameter Set element is received from the QAP of the QBSS with which the QSTA is associated., where:

- a) The parameters used by the channel access function to control its operation are defined by MIB attribute table dot11QAPEDCATable at the QAP, and by MIB attribute table dot11EDCATable at the non-AP QSTA.
- b) The minimum specified idle duration time is not the constant value (DIFS) as defined for DCF, but is a distinct value (contained in the MIB attribute table dot11QAPEDCATableAIFSN for a QAP, and in the MIB table dot11EDCATableAIFSN for a non-AP QSTA, see 9.9.1) assigned either by a management entity or by a QAP.
- c) The contention window limits aCWmin and aCWmax, from which the random backoff is computed, are not fixed per PHY, as with DCF, but are variable, contained in the MIB attribute tables dot11QAPEDCACWmin and dot11QAPEDCACWmax for a QAP, and in the MIB attribute tables dot11EDCATableCWmin and dot11EDCATableCWmax for a non-AP QSTA, assigned by a management entity or by a QAP.
- d) Collisions between contending channel access functions within a QSTA are resolved within the QSTA such that the data frames from the higher-priority AC receives the TXOP and the data frames from the lower-priority colliding AC(s) behave as if there were an external collision on the WM. Note, however, that this collision behavior does not include setting retry bits in the MAC headers of MPDUs at the head of the lower priority ACs, as would be done after a transmission attempt that was unsuccessful due to an actual external collision on the WM.
- e) During an EDCA TXOP won by a channel access function, a QSTA may initiate multiple frame exchange sequences to transmit MMPDUs and/or MSDUs within the same AC. The duration of this EDCA TXOP is bounded, for an AC, by the value, in dot11QAPEDCATXOPLimit MIB variable for a QAP and in dot11EDCATableTXOPLimit MIB table for a non-AP QSTA. A value of 0 for this duration means that the EDCA TXOP is limited to a single MSDU or MMPDU at any rate in the operational set of the QBSS.

The QAP announces the EDCA parameters in selected beacons, and in all probe response and (re)association response frames by the inclusion of the EDCA Parameter Set information element. If no such element is received, the QSTAs shall use the default values for the parameters. The fields following the AC Info field in the EDCA Parameter Set information element shall be included in all beacons occurring within two or more DTIM periods following a change in AC parameters to assure that all QSTAs are able to receive the updated EDCA parameters. A QSTA shall update its MIB values of the EDCA parameters within an interval of time equal to one beacon interval after receiving an updated EDCA parameter set. QSTAs update the MIB attributes and store the EDCA Parameter Set Update Count Value in the QoS Info field. A QAP may change the EDCA access parameters by changing the EDCA Parameter Set element in the beacon and probe response. However, the QAP should change them only rarely. A QSTA shall use the EDCA Parameter Set Update Count Value subfield in the QoS Capability element of all beacons to determine if the QSTA is using the current EDCA Parameter Values. If the EDCA Parameter Set Update Count Value in the QoS Capability element is different from the value that has been stored, the QSTA shall query the updated EDCA parameter values by sending a Probe Request to the QAP.

The QAP may use a different set of EDCA parameters than it advertises to the QSTAs in its QBSS.

On receiving a frame with TID subfield in QoS Control corresponding to an EDCF AC, the QSTA shall consider all contexts, including block ack status, to be related to each AC.

The management type frames shall be sent using the access category AC_VO without being restricted by admission control procedures. BlockAckReq and BlockAck Control frames shall be sent using the same QoS parameters as the corresponding QoS Data frames. PS-Poll Control frames shall be sent using the access category AC_BE to reduce the likelihood of collision following a beacon. For the purpose of determining the proper access category for an RTS frame, the RTS frame shall inherit the UP of the data or management frame(s) that is (are) included in the frame exchange sequence wherein the RTS is the first frame.

The priority mapping of Data type frames with a multicast or broadcast destination address is described in Table 20.20.

Table 20.20 – Priority mapping for multicast and broadcast frames in a QBSS

To DS	From DS	User Priority (same as 802.1D)	Priority handling	Frame format
0	0	Any	mapping to AC specified in Table 0.1	Data frames with QoS subfield set to 0, unless it is known to the transmitter that all members of the group are QSTAs, in which case Data frames with the QoS subfield set to 1 are used.
0	1	0	handling as specified in clause 9.9.	
		> 0	mapping to AC as specified in Table 0.1	
1	0	Any	these frames always have a unicast receiver address and are to be handled like other unicast data MPDUs	
<u>1</u>	<u>1</u>			

The operation rules of HCF contention-based channel access are defined in 9.9.1.

9.1.3.2 HCF controlled channel access (HCCA)

The HCF controlled channel access mechanism uses a QoS-aware centralized coordinator, called a hybrid coordinator (HC), and operates under rules that are different from the point coordinator (PC) of the PCF. The HC is collocated with the QoS access point (QAP) of the QBSS and uses the HC's higher priority of access to

the WM to initiate frame exchange sequences and to allocate TXOPs to itself and other QSTAs, so as to provide limited-duration controlled access phase (CAP) for contention-free transfer of QoS data.

The HC traffic delivery and TXOP allocation may be scheduled during the CP and any locally-generated CFP, to meet the QoS requirements of a particular TC or TS. TXOP allocations and contention free transfers of QoS traffic can be based on the HC's QBSS-wide knowledge of the amounts of pending traffic belonging to different TS and/or TCs and is subject to QBSS-specific QoS policies.

A QAP may indicate availability of CF-Polls to non-QoS STAs, thereby providing non-QoS contention free transfers during the CFP. This provisioning of contention free transfers during the CFP to non-QoS STAs, however, is not recommended. Implementers are cautioned that QSTAs are not required to interpret data subtypes that include +CF-Ack in frames not addressed to themselves unless they set the Q-Ack subfield in the QoS Capability information element to 1. QSTAs are also not required to interpret data subtypes that are non-QoS (+)CF-Polls (i.e., frames of subtype data with bits 7, 5 and 4 in Frame Control set to 0, 1 and 0 respectively), and therefore QSTAs cannot be treated as CF-Pollable stations. This requires a QAP that provides non-QoS CF-polling to adhere to frame sequence restrictions considerably more complex than, and less efficient than, those specified for either PCF or HCF. In addition, the achievable service quality is likely to be degraded when non-QoS STAs are associated and being polled.

The HCF protects the transmissions during each CAP using the virtual carrier sense mechanism.

A QSTA may initiate multiple frame exchange sequences during a polled TXOP of sufficient duration to perform more than one such sequence. The use of virtual carrier sense by the HC provides improved protection of the CFP, in addition to the protection provided by having all STAs in the BSA setting their NAVs to `dot11CFPMaxDuration` at TBTT of DTIM Beacons.

The operation rules of the HCF controlled channel access are defined in 9.9.2.

Change the heading and text of clause 9.1.3 (renumbered 9.1.4 due to the insertion above) as follows:

9.1.4 Coexistence of DCF, and PCF and HCF

The DCF and ~~the~~ a centralized coordination function (either PCF or HCF) shall coexist in a manner that permits both to operate concurrently within the same BSS. When a PC is operating in a BSS, the PCF and DCF access methods alternate, with a contention-free period (CFP) followed by a contention period (CP). This is described in greater detail in 9.3. When an HC is operating in a QBSS, it may generate an alternation of CFP and CP in the same way as a PC, using the DCF access method only during the CP. The HCF access methods (controlled and contention-based) operate concurrently when the channel is in CP. Concurrent operation allows the polled and contention-based access methods to alternate, within intervals as short as the time to transmit a frame exchange sequence, under rules defined in 9.9.

Change the heading and text of clause 9.1.5 (renumbered 9.1.5 due to the insertion above) as follows:

9.1.5 Fragmentation/defragmentation overview

The process of partitioning a MAC service data unit (MSDU) or a MAC management protocol data unit (MMPDU) into smaller MAC level frames, MAC protocol data units (MPDUs), is called fragmentation. Fragmentation creates MPDUs smaller than the original MSDU or MMPDU length to increase reliability, by increasing the probability of successful transmission of the MSDU or MMPDU in cases where channel characteristics limit reception reliability for longer frames. QSTAs may use fragmentation to use the medium efficiently in consideration of the duration available in granted TXOPs, as long as the rules in 9.4 are followed. Fragmentation is accomplished at each immediate transmitter. The process of recombining MPDUs into a single MSDU or MMPDU is defined as defragmentation. Defragmentation is accomplished at each immediate recipient.

Only MPDUs with a unicast receiver address shall be fragmented. Broadcast/multicast frames shall not be fragmented even if their length exceeds `dot11FragmentationThreshold`.

When a directed MSDU is received from the LLC or a directed MMPDU is received from the MAC sublayer management entity (MLME) with a length greater than `dot11FragmentationThreshold`, the MSDU or

MMPDU shall be fragmented. The MSDU or MMPDU is divided into MPDUs. Each fragment is a frame no larger than ~~adot11~~FragmentationThreshold. It is possible that any fragment may be a frame smaller than ~~adot11~~FragmentationThreshold. An illustration of fragmentation is shown in Figure 48.

The MPDUs resulting from the fragmentation of an MSDU or MMPDU are sent as independent transmissions, each of which is separately acknowledged. This permits transmission retries to occur per fragment, rather than per MSDU or MMPDU. Unless interrupted due to medium occupancy limitations for a given PHY or TXOP limitations for QSTA, the fragments of a single MSDU or MMPDU are sent as a burst during the CP, using a single invocation of the DCF or EDCA medium access procedure. The fragments of a single MSDU or MMPDU are either sent during a CFP as individual frames obeying the rules of the PC medium access procedure or sent as a burst in a TXOP.

Change the heading and text of clause 9.1.5 (renumbered 9.1.6 due to the insertion above) as follows:

9.1.6 MAC Data Service

The MAC Data Service provides the transport of MSDUs between MAC peer entities as characterized in 6.1.1.

The transmission process is started by receipt of an MA-UNITDATA.request containing an MSDU and the associated parameters. This may cause one or more Data MPDUs containing the MSDU to be transmitted following fragmentation and security encapsulation, as appropriate.

The MA-UNITDATA.indication is generated in response to one or more received Data MPDUs containing an MSDU following validation, address filtering, decryption decapsulation and defragmentation, as appropriate.

~~The MAC Data Service shall translate MAC service requests from LLC into input signals utilized by the MAC State Machines. The MAC Data Service shall also translate output signals from the MAC State Machines into service indications to LLC. The translations are given in the MAC Data Service State Machine defined in Annex C.~~

The MAC Data Service for QSTAs shall incorporate a traffic identifier (TID) with each MA-UNITDATA.request service. This TID associates the MSDU with the AC or TS queue for the indicated traffic.

9.2 DCF

9.2.3 Interframe space (IFS)

Change the text and figure 49 in clause 9.2.3 as follows:

The time interval between frames is called the IFS. A STA shall determine that the medium is idle through the use of the carrier sense function for the interval specified. ~~Four~~Five different IFSs are defined to provide priority levels for access to the wireless media; they are listed in order, from the shortest to the longest except for AIFS. Figure 49 shows some of these relationships.

- | | |
|--------------------|--|
| a) SIFS | Short Interframe Space |
| b) PIFS | Point Coordination Function (PCF) Interframe Space |
| c) DIFS | Distributed Coordination Function (DCF) Interframe Space |
| d) AIFS | Arbitration Interframe Space (used by the QoS facility) |
| ~~e)~~ EIFS | Extended Interframe Space |

The different IFSs shall be independent of the STA bit rate. The IFS timings are ~~shall be~~ defined as time gaps on the medium, and those except for AIFS are ~~shall be~~ fixed for each PHY (even in multirate-capable PHYs). The IFS values are determined from attributes specified by the PHY.

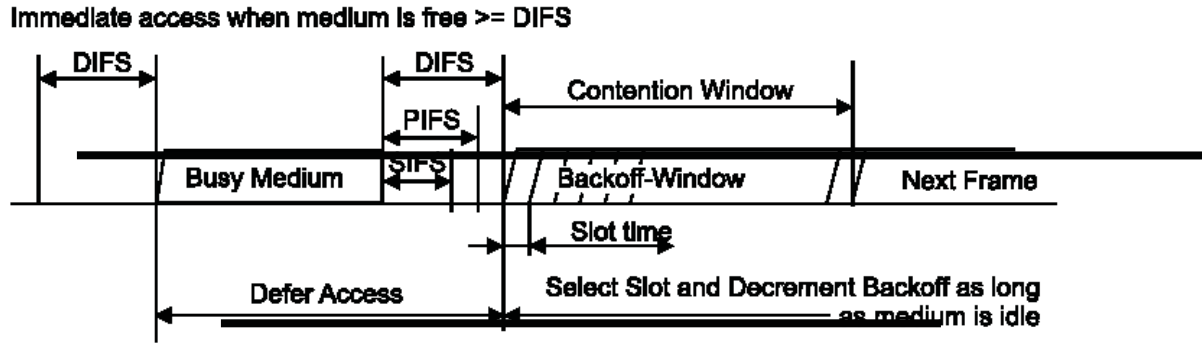


Figure 49—Some IFS relationships

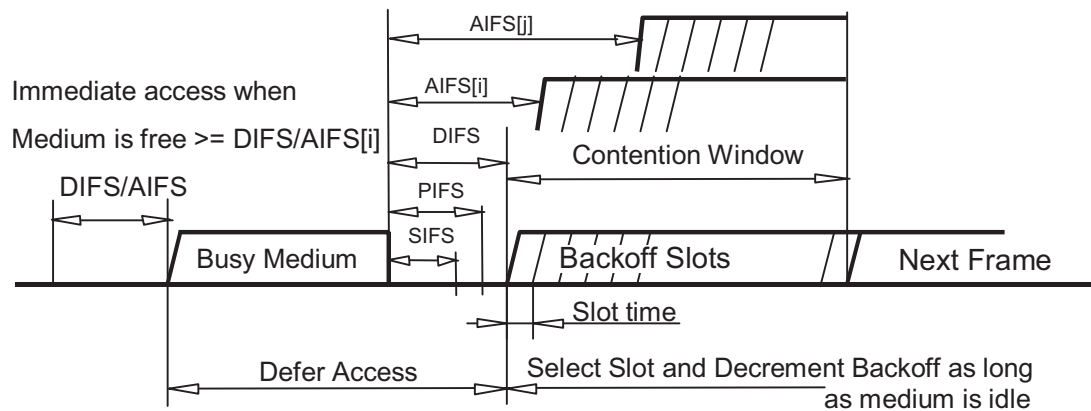


Figure 49 - Some IFS Relationships

Insert after 9.2.3.3 the following subclause and renumber the current 9.2.3.4 as 9.2.3.5

9.2.3.4 Arbitration IFS (AIFS)

The Arbitration Interframe Space shall be used by QSTAs to transmit Data type frames (MPDUs) and Management type frames (MMPDUs). A QSTA using the EDCA shall obtain a TXOP for an AC if the QSTA's carrier sense mechanism (see 9.2.1) determines that the medium is idle at the AIFS[AC] slot boundary (see 9.9.1.3), after a correctly-received frame, and the backoff time for that AC has expired.

A QSTA using the EDCA shall not transmit within an EIFS+DIFS+AIFS[AC] after that QSTA determines that the medium is idle following reception of a frame for which the PHYRXEND.indication primitive reported an error or a frame for which the MAC FCS value was not correct, unless a subsequent reception of an error-free frame resynchronizes the station. This resynchronization allows the QSTA to transmit using AIFS[AC] following that subsequent frame.

A Non-AP QSTA computes the time periods for each AIFS[AC] from the dot11EDCATableAIFSN attributes in the MIB. QSTAs update their dot11EDCATableAIFSN values using information in the most recent EDCA Parameter Set element of Beacons received from the QAP of the QBSS (see 7.3.2.14). An AP computes the time periods for each AIFS[AC] from the dot11QAPEDCATableAIFSN attributes in its MIB.

9.2.3.5 EIFS

Change the text in 9.2.3.5 as follows:

The EIFS shall be used by the DCF and the EDCA mechanism under HCF when the PHY has indicated to the MAC that a frame transmission was begun that did not result in the correct reception of a complete MAC

frame with a correct FCS value. The duration of an EIFS is defined in 9.2.10. The EIFS interval shall begin following indication by the PHY that the medium is idle after detection of the erroneous frame, without regard to the virtual CS mechanism. The EIFS is defined to provide enough time for another STA to acknowledge what was, to this STA, an incorrectly received frame before this STA commences transmission. Reception of an error-free frame during the EIFS resynchronizes the STA to the actual busy/idle state of the medium, so the EIFS is terminated and normal medium access (using DIFS or AIFS as appropriate and, if necessary, backoff) continues following reception of that frame.

9.2.9 Duplicate detection and recovery

Change the text in 9.2.9 as follows:

Because MAC-level acknowledgments and retransmissions are incorporated into the protocol, there is the possibility that a frame may be received more than once. Such duplicate frames shall be filtered out within the ~~destination~~ receiver MAC.

Duplicate frame filtering is facilitated through the inclusion of a Sequence Control field (consisting of a sequence number and fragment number) within data and management frames as well as TID field in QoS Control within QoS data frames. MPDUs that are part of the same MSDU shall have the same sequence number, and different MSDUs shall (with a high probability) have a different sequence number.

The sequence number, for frames of type management and frames of subtype data with QoS subfield set to 0, is generated by the transmitting STA as an incrementing sequence of integers. In a QSTA, the sequence numbers for QoS (+)Data frames are generated by different counters for each TID and receiver pair and shall be incremented one by one.

The receiving STA shall keep a cache of recently received <Address 2, sequence-number, fragment-number> tuples. The receiving QSTA shall also keep a cache of recently received <Address 2, TID, sequence-number, fragment-number> tuples for all QSTAs from whom it has received QoS data frames. A receiving STA is required to keep only the most recent cache entry per Address 2-sequence-number pair, storing only the most recently received fragment number for that pair. A receiving QSTA is also required to keep only the most recent cache entry per Address 2-TID-sequence-number triple, storing only the most recently received fragment number for that triple. A receiving STA may omit tuples obtained from broadcast/multicast or ATIM frames from the cache.

A ~~destination~~ receiver STA shall reject as a duplicate frame any frame that has the Retry bit set in the Frame Control field and that matches an <Address 2, sequence-number, and fragment-number> tuple of an entry in the cache. A receiver QSTA shall also reject as a duplicate frame any frame that has the Retry bit set in the Frame Control field and that matches an <Address 2, TID, sequence-number, and fragment-number> tuple of an entry in the cache.

There is a small possibility that a frame may be improperly rejected due to such a match; however, this occurrence would be rare and simply results in a lost frame (similar to an FCS error in other LAN protocols).

The ~~destination~~ receiver STA shall perform the ACK procedure on all successfully received frames requiring acknowledgment, even if the frame is discarded due to duplicate filtering.

9.6 Multirate support

Change the text in 9.6 as follows:

Some PHYs have multiple data transfer rate capabilities that allow implementations to perform dynamic rate switching with the objective of improving performance. The algorithm for performing rate switching is beyond the scope of this standard, but in order to ensure coexistence and interoperability on multirate-capable PHYs, this standard defines a set of rules that shall be followed by all STAs.

With the exception of BlockAckReq and BlockAck frames, aAll control frames that initiate a frame exchange shall be transmitted at one of the rates in the BSSBasicRateSet, unless the transmitting STAs protection mechanism is enabled, and the control frame is a protection mechanism frame; in which case, the control

frame shall be transmitted at a rate according to the separate rules for determining the rates of transmission of protection frames in 9.10.

All frames with multicast and broadcast in the address1 field that have a user priority of zero shall be transmitted at one of the rates included in the BSS basic rate set, regardless of their type or subtype.

All data frames of subtype CF-Poll sent in the CP shall be transmitted at one of the rates in the BSS basic rate set so that they will be understood by all STAs in the BSS, unless an RTS-CTS exchange has already been performed before the transmission of the data frame of subtype CF-Poll and the duration field in the RTS frame covers the entire TXOP. All other frames of subtype Data, BlockAckReq and BlockAck frames and/or management MPDUs with unicast in address1 shall be sent using on any supported data rate subject to the following constraints, selected by the rate switching mechanism. No STA shall transmit at a rate that is not supported by the ~~receiver~~destination STA, as reported in the Supported Rates and Extended Supported Rates element in the management frames. For frames of type (QoS) Data+CF-ACK, (QoS) Data+CF-Poll+CF-ACK, and (QoS) CFPoll+CF-ACK, the rate chosen to transmit the frame must be supported by both the addressed recipient STA and the STA to which the ACK is intended.

Under no circumstances shall a STA initiate transmission of a data or management frame at a data rate higher than the greatest rate in the OperationalRateSet, a parameter of the MLME-JOIN.request primitive.

To allow the transmitting STA to calculate the contents of the Duration/ID field, a STA shall transmit its Control Response frame (either CTS or ACK) ~~frames, other than the BlockAck control frame,~~ at the highest rate in the BSSBasicRateSet that is less than or equal to the rate of the immediately previous frame in the frame exchange sequence (as defined in 9.7.12) and that is of the same modulation type as the received frame. If no rate in the basic rate set meets these conditions, then the control frame sent in response to a received frame shall be transmitted at the highest mandatory rate of the PHY that is less than or equal to the rate of the received frame, and that is of the same modulation type as the received frame. In addition, the Control Response frame shall be sent using the same PHY options as the received frame, unless they conflict with the requirement to use the BSSBasicRateSet. The BlockAck control frame shall be sent at the same rate as the BlockAckReq frame if it is sent in response to a BlockAckReq frame.

An alternative rate for the control response frame may be used, provided that the duration of the control response frame at the alternative rate is the same as the duration of the control response frame at the originally chosen rate and the alternative rate is in either the BSSBasicRateSet or the mandatory rate set of the PHY and the modulation of the control response frame at the alternative rate is the same type as that of the received frame.

For the Clause 17, Clause 18 and Clause 19 PHYs, the time required to transmit a frame for use in the Duration/ID field is determined using the PLME-TXTIME.request primitive (see 10.4.6) and the PLME-TXTIME.confirm primitive (see 10.4.7) , both defined in 17.4.3, 18.3.4, 19.8.3.1, 19.8.3.2, or 19.8.3.3 depending on the PHY options. In QSTAs, the Duration/ID field may cover multiple frames and may involve using the PLME-TXTIME.request several times.

Delete subclause 9.7 and move the title and contents to subclause 9.12, renumber all the following subclauses. The subclause numbers and titles for each of the re-numbered subclauses are shown below:

9.7 MSDU transmission requirements

Change the text in the second paragraph of the subclause (numbered 9.9 in the base standard) as shown:

~~The A non-QoS~~ STA shall ensure that no more than one MSDU or MMPDU from a particular SA to a particular individual RA is outstanding at a time. Note that a simpler, more restrictive invariant to maintain is that no more than one MSDU with a particular individual RA may be outstanding at a time.

1 *Insert the following paragraph between second and third paragraphs of the subclause:*

2 For all transmissions not using the Ack policy of Block Ack, a QSTA shall ensure that no more than one
 3 MSDU or MMPDU with a particular TID from a particular SA to a particular individual RA is outstanding at
 4 any time. Note that a simpler, more restrictive invariant to maintain is that no more than one MSDU with any
 5 particular TID with a particular individual RA may be outstanding at any time. This restriction is not
 6 applicable for MSDUs that are to be transmitted using Block Ack mechanism.

7 *Insert the following paragraphs at the end of the subclause:*

8 For MSDUs belonging to the service class of “QoSAck” and the receiver is a QSTA, the QoS Data frames
 9 which are used to send these MSDUs shall have the Ack Policy subfield in QoS Control set to “Normal
 10 Acknowledgement” or “Block Acknowledgement”. For MSDUs belonging to the service class of
 11 “QoSNoAck” and the receiver is a QSTA, the QoS Data frames which are used to send these MSDUs shall
 12 have the Ack Policy subfield in QoS Control set to “No Acknowledgement”.

13 For MSDUs belonging to the service class of “QoSLocalMulticast”, the data frames which are used to send
 14 these MSDUs shall have a broadcast/multicast address in the address1 field and the ToDS and FromDS
 15 subfields in Frame Control both set to 0. A QAP that receives these frames from a non-AP QSTA shall not
 16 forward the MSDU to the DS.

17 9.8 Operation across regulatory domains

18 *Insert the following subclause, including tables and figures included therein, after 9.8, renumbering tables*
 19 *and figures as necessary:*

20 9.9 HCF

21 This clause describes the QoS enhancements to the MAC functional description. QSTAs may access the
 22 channel in a more controlled manner, compared to a non-QSTA, to transmit MPDUs.

23 Under HCF the basic unit of allocation of the right to transmit onto the WM is the TXOP. Each TXOP is
 24 defined by a starting time and a defined maximum length. The TXOP may be obtained by a QSTA winning
 25 an instance of EDCA contention (see 9.9.1) during the CP, or by a non-AP QSTA receiving a QoS (+)CF-Poll
 26 (see 9.9.2) during the CP or CFP. The former is called EDCA TXOP, while the latter is called HCCA TXOP
 27 or polled TXOP. A TXOP shall not extend across a TBTT or exceed dot11MaxDwellTime (if using an FH
 28 PHY).

29 9.9.1 HCF contention-based channel access (EDCA)

30 9.9.1.1 Reference Implementation (Informative)

31 The channel access protocol is derived from the DCF procedures described in clause 9.2.

32

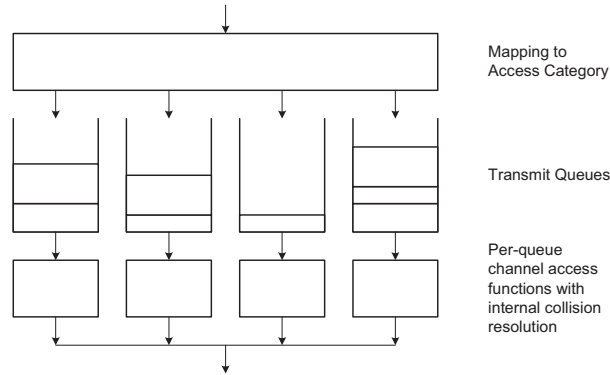


Figure 62.1 – Reference Implementation Model

A model of the reference implementation is shown in Figure 62.1 and illustrates a mapping from frame type or UP to AC; the four transmit queues and the four independent channel access functions, one for each queue. The mapping of UP to the AC is described in 6.1.1.3 and Table 0.1. The mapping of frame types to ACs is described in 9.1.3.1.

9.9.1.2 EDCA TXOPs

There are two modes of EDCA TXOP defined, the initiation of the EDCA TXOP and the EDCA TXOP continuation. An initiation of the TXOP occurs when the EDCA rules permit access to the medium. A TXOP continuation occurs when a channel access function retains the right to access the medium following the completion of a frame exchange sequence, such as on receipt of an ACK frame.

The TXOP limit duration values are advertised by the QAP in the EDCA Parameter Set Information Element in Beacons and Probe Response frames transmitted by the QAP. A TXOP limit value of 0 indicates that a single MSDU or MMPDU, in addition to a possible RTS/CTS exchange or CTS to itself, may be transmitted at any rate for each TXOP.

Non-AP QSTAs shall ensure that the duration of TXOPs obtained using the EDCA rules do not exceed the TXOP limit. The duration of a TXOP is the duration during which the TXOP holder maintains uninterrupted control of the medium, and it includes the time required to transmit frames sent as an immediate response to TXOP holder transmissions.

A QSTA shall fragment an MSDU so that the transmission of the first MPDU of the TXOP does not cause the TXOP limit to be exceeded at the PHY rate selected for the initial transmission attempt of that MPDU. The TXOP limit may be exceeded, when using a lower PHY rate than selected for the initial transmission attempt of the first MPDU, for a retransmission of an MPDU, or for the initial transmission of an MPDU if any previous MPDU in the current MSDU has been retransmitted. When the TXOP limit is exceeded due to the retransmission of a MPDU at a reduced PHY rate, the STA shall not transmit more than one MPDU in the TXOP.

It should be noted, that when transmitting frames using acknowledgement mechanisms other than immediate ACK a protective mechanism should be used (such as RTS/CTS or the protection mechanism described in 9.10).

9.9.1.3 Obtaining an EDCA TXOP

Each channel access timer shall maintain a backoff function (timer), which has a value measured in backoff slots.

The duration AIFS[AC] is a duration derived from the value AIFSN[AC] by the relation

$$\text{AIFS[AC]} = \text{AIFSN[AC]} \times \text{aSlotTime} + \text{aSIFSTime}.$$

1 The value of AIFSN[AC] shall be greater than or equal to 2 for non-AP QSTAs and is advertised by the QAP
 2 in the EDCA Parameter Set Information Element in Beacons and Probe Response frames transmitted by the
 3 QAP. The value of AIFSN[AC] shall be greater than or equal to 1 for QAPs. An EDCA TXOP is granted to a
 4 channel access function when the channel access function determines that it shall initiate the transmission of a
 5 frame exchange sequence. Transmission initiation shall be determined according to the following rules:

6 On specific slot boundaries, each channel access function shall make a determination to perform one and only
 7 one of the following functions:

- 8 a) Initiate the transmission of a frame exchange sequence for that access function
- 9 b) Decrement the backoff timer for that access function
- 10 c) Invoke the backoff procedure due to an internal collision
- 11 d) Do nothing for that access function

12 The specific slot boundaries at which exactly one of these operations shall be performed are defined as
 13 follows, for each channel access function:

- 14 a) Following $\text{AIFSN[AC]} \times \text{aSlotTime} - \text{aRxTxTurnaroundTime}$ of medium idle indication after SIFS
 15 (not necessarily idle time) after the last busy medium on the antenna that was the result of a frame
 16 reception that has not resulted in FCS error.
- 17 b) Following $\text{EIFS} - \text{DIFS} + \text{AIFSN[AC]} \times \text{aSlotTime}$ of medium idle indication after the last indicated
 18 idle medium as determined by the CS mechanism, which was the result of a frame reception that has
 19 resulted in FCS error or PHY-RXEND.indication(RXERROR), where the value of RXERROR is not
 20 NoError..
- 21 c) When any other channel access function at this QSTA transmitted a frame requiring
 22 acknowledgement, the earlier of
 - 23 1) the latter of
 - 24 i) the end of the ACK-Timeout interval plus $\text{AIFSN[AC]} \times \text{aSlotTime} -$
 25 $\text{aTxRxTurnaroundTime}$ of IDLE time after the PHY-TXEND.confirm, and
 - 26 ii) the end of the first $\text{AIFSN[AC]} \times \text{aSlotTime} - \text{aTxRxTurnaroundTime}$ of IDLE time
 27 following the PHY-TXEND.confirm, and
 - 28 2) at the end of the first $\text{AIFSN[AC]} \times \text{aSlotTime} - \text{aRxTxTurnaroundTime}$ of IDLE medium
 29 after the PHY-RXEND.indication when a PHY-RXSTART.indication occurs as specified in
 30 subclause 9.2.8,
- 31 d) Following $\text{AIFSN[AC]} \times \text{aSlotTime} - \text{aRxTxTurnaroundTime}$ of medium idle indication after SIFS(
 32 not necessarily idle time) after the last busy medium on the antenna that was the result of a
 33 transmission of a frame for any channel access function and which did not require an
 34 acknowledgement.
- 35 e) Following $\text{AIFSN[AC]} * \text{aSlotTime}$ of medium idle time indication after the last indicated idle
 36 medium as indicated by the CS mechanism that is not covered by a) through d)
- 37 f) Following aSlotTime of medium idle indication, which occurs immediately after any of these
 38 conditions, a) through f) is met for the channel access function.

39 At each of the above-described specific slot boundaries, each channel access function shall initiate a
 40 transmission sequence, if

- 41 a) there is a frame available for transmission at that channel access function, and
- 42 b) the backoff timer for that channel access function has a value of zero, and
- 43 c) initiation of a transmission sequence is not allowed to commence at this time for a channel access
 44 function of higher UP.

At each of the above-described specific slot boundaries, each channel access function shall decrement the backoff timer, if the backoff timer for that channel access function has a non-zero value.

At each of the above-described specific slot boundaries, each channel access function shall invoke the backoff procedure due to an internal collision, if

- there is a frame available for transmission at that channel access function, and
- the backoff timer for that channel access function has a value of zero, and
- initiation of a transmission sequence is allowed to commence at this time for a channel access function of higher UP.

At each of the above-described specific slot boundaries, a channel access function shall do nothing, if none of the above actions are taken.

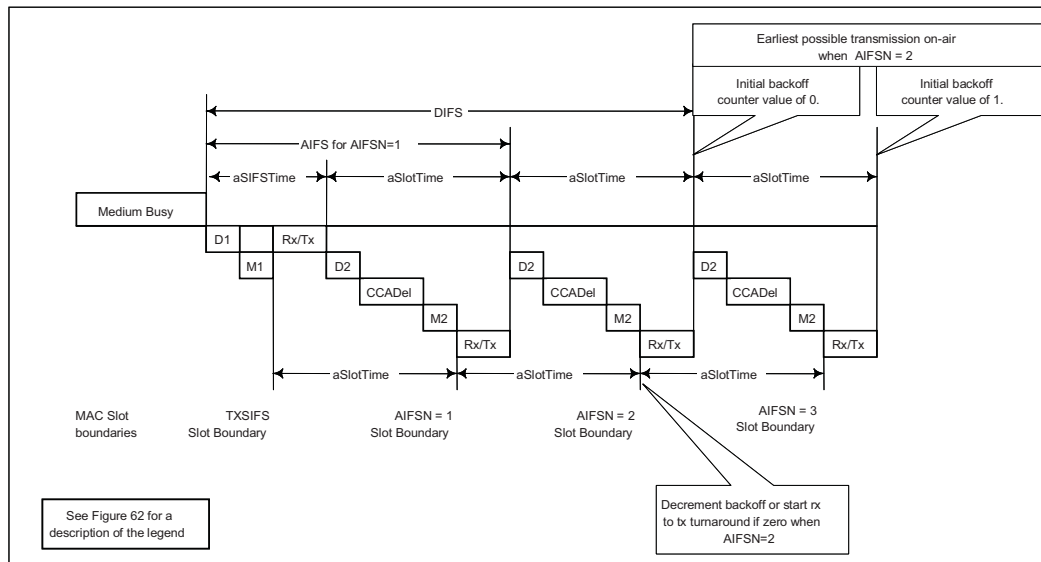


Figure 62.2 – EDCA Mechanism Timing Relationships

An example showing the relationship between AIFS, AIFSN, DIFS and slot times immediately following a medium busy condition (and assuming that medium busy condition was not caused by a frame in error) is shown in Figure 62.2. In this case, with AIFSN=2, the channel access function may decrement the backoff counter for the first time at $2 \times \text{aSlotTime}$ following the end of the medium busy condition (end of the medium busy condition happens at the end of M1 in Figure 62.2). If, in this example, the backoff counter contained a value of 1 at the time the medium became idle, transmission would start as a result of an EDCA TXOP on-air at a time

$$\text{aSIFSTime} + 3 \times \text{aSlotTime}$$

following the end of the medium busy condition.

9.9.1.4 Obtaining a Continuation of EDCA TXOP

A continuation of EDCA TXOP is granted to a channel access function at a SIFS period following the successful transmission of a frame, if there is a frame available for transmission that fits in any remaining EDCA TXOP duration. If a QSTA has in its transmit queue an additional frame of the same AC as the one just transmitted, and the duration of transmission of that frame plus any expected acknowledgement for that frame is less than the remaining medium occupancy timer value, then the QSTA may commence transmission of that frame at SIFS after the completion of the immediately preceding frame exchange sequence. The

intention of using the continuation TXOP shall be indicated by the QSTA through the setting of the duration/ID values in one of the following two ways (see 7.1.4):

- a) long enough to cover the response frame, the next frame and its response frame.
- b) long enough to cover the transmission of a burst of MPDUs subject to the limit set by `dot11EDCATableTXOPLimit`.

If the Duration/ID is set for continuation of TXOP, if there is a transmission failure, the corresponding channel access function may recover before the expiry of the NAV setting due to the setting of the Duration/ID field in the frame that resulted in a transmission failure. However, at the expiry of the NAV set by the frame that resulted in a transmission failure, the channel access function shall invoke backoff procedure.

A frame exchange may be either a multicast frame transmitted or a frame transmitted with “no acknowledgement” policy, for which there is no expected acknowledgement, or a unicast frame followed by a correctly received acknowledgement frame transmitted by either a non-AP QSTA or a QAP.

Note that, as for an EDCA TXOP, a continuation of TXOP is granted to a channel access function, not to a non-AP QSTA or QAP, such that the continuation of TXOP is only permitted for the transmission of a frame of the same access category as the access category of the frame that was granted the EDCA TXOP.

9.9.1.5 EDCA Backoff Procedure

Each channel access function shall maintain a state variable `CW[AC]`, which shall be initialized to the value of the parameter `CWmin[AC]`.

If a frame is successfully transmitted by a specific channel access function, indicated by either the successful reception of a CTS in response to an RTS, the successful reception of an ACK in response to a unicast MPDU or BlockAck, or the successful reception of a BlockAck or ACK in response to a BlockAckReq, or by transmitting a multicast frame or a frame with “no acknowledgement” policy, `CW[AC]` shall be reset to `CWmin[AC]`.

The backoff procedure shall be invoked for a channel access function when either:

- a) A frame with that AC is requested to be transmitted and the medium is busy as indicated by either physical or virtual carrier sense, and the backoff timer has a value of zero for that AC.
- b) The final transmission by the TXOP holder initiated during the TXOP for that AC was successful.
- c) The transmission of a frame of that AC fails, indicated by a failure to receive a CTS in response to an RTS, or a failure to receive an ACK that was expected in response to a unicast MPDU, or failure to receive a BlockAck or ACK in response to a BlockAckReq.
- d) The transmission attempt collides internally with another channel access function of an AC that has higher priority, that is, two or more channel access functions in the same QSTA are granted a TXOP at the same time.

If the backoff procedure is invoked for reason a) above, the value of `CW[AC]` shall be left unchanged. If the backoff procedure is invoked because of reason b) above, the value of `CW[AC]` shall be reset to `CWmin[AC]`.

If the backoff procedure is invoked because of a failure event (either reason c or d above) the value of `CW[AC]` shall be updated as follows before invoking the backoff procedure:

- a) if the short or long retry count for the QSTA has reached `aShortRetryLimit` or `aLongRetryLimit` respectively, `CW[AC]` shall be reset to `CWmin[AC]`.
- b) Otherwise,
 - 1) if `CW[AC]` is less than `CWmax[AC]`, `CW[AC]` shall be set to the value $(CW[AC]+1)*2-1$;

- 2) if $CW[AC]$ is equal to $CW_{max}[AC]$, $CW[AC]$ shall remain unchanged for the remainder of any retries.

Following the update of the value of $CW[AC]$, the backoff timer is set to an integer value chosen randomly with a uniform distribution taking values in the range $[0, CW[AC]]$ inclusive.

9.9.1.6 Retransmit Procedures

QSTAs shall maintain a short retry counter and a long retry counter for each MSDU or MMPDU requiring acknowledgement and awaiting transmission using the EDCA access methods. The initial value for the short and long retry counters shall be zero. QSTAs also maintain short retry counter and long retry counter for each AC and defined as $QSRC[AC]$ and $QLRC[AC]$, each initiated to a value of zero.

After transmitting a frame that requires acknowledgment, the QSTA shall perform the ACK procedure, as defined in 9.2.8. The short retry count for an MSDU or MMPDU and the $QSRC[AC]$ shall be incremented every time transmission of a MAC frame of length less than or equal to $dot11RTSThreshold$ fails for that MSDU or MMPDU. This short retry count and the QSTA $QSRC[AC]$ shall be reset when a MAC frame of length less than or equal to $dot11RTSThreshold$ succeeds for that MSDU or MMPDU. The long retry count for an MSDU or MMPDU and the $QLRC[AC]$ shall be incremented every time transmission of a MAC frame of length greater than $dot11RTSThreshold$ fails for that MSDU or MMPDU. This long retry count and the $QLRC[AC]$ shall be reset when a MAC frame of length greater than $dot11RTSThreshold$ succeeds for that MSDU or MMPDU. All retransmission attempts for an MSDU or MMPDU that has failed the ACK procedure one or more times shall be made with the Retry field set to 1 in the Data or Management type frame.

Retries for failed transmission attempts shall continue until the short retry count for the MSDU or MMPDU is equal to $dot11ShortRetryLimit$ or until the long retry count for the MSDU or MMPDU is equal to $aLongRetryLimit$. When either of these limits is reached, retry attempts shall cease, and the MSDU or MMPDU shall be discarded.

For internal collisions occurring with the EDCA access method the appropriate retry counters (short retry counter for MSDU or MMPDU and $QSRC[AC]$ or long retry counter for MSDU or MMPDU and $QLRC[AC]$) are incremented. For transmissions that use Block acknowledgement, the rules in 9.11 also apply. QSTAs shall retry failed transmissions until the transmission is successful or until the relevant retry limit is reached.

A QSTA shall not initiate the transmission of a frame of any management or data type to a specific RA while the transmission of another frame of management or data type with the same RA and having been assigned its sequence number from the same sequence counter, has not yet completed to the point of success, retry fail, or other MAC discard (e.g. lifetime expiry).

QSTAs shall maintain a transmit MSDU timer for each MSDU passed to the MAC. The MIB attribute $dot11EDCATableMSDULifetime$ specifies the maximum amount of time allowed to transmit an MSDU for a given AC. The transmit MSDU timer shall be started when the MSDU is passed to the MAC. If the value of this timer exceeds the appropriate entry in $dot11EDCATableMSDULifetime$, then the MSDU, or any remaining, undelivered fragments of that MSDU, shall be discarded by the source QSTA without any further attempt to complete delivery of that MSDU.

9.9.2 HCF controlled channel access

The hybrid coordination function (HCF) controlled channel access (HCCA) mechanism manages access to the WM, using a hybrid coordinator (HC) that has higher medium access priority than non-AP STAs. This allows it to transfer traffic from itself and to allocate transmission opportunities (TXOPs) to non-AP QSTAs.

The HC is a type of centralized coordinator, but differs from the PC used in PCF in several significant ways, although it may optionally implement the functionality of a PC. Most important is that HCF frame exchange sequences may be used among QSTAs associated in a QBSS during both the CP and the CFP. Another significant difference is that the HC grants a non-AP QSTA a polled TXOP with duration specified in a QoS

(+)CF-Poll frame. Non-AP QSTAs may transmit multiple frame exchange sequences within given polled TXOPs, subject to the limit on TXOP duration.

All STAs inherently obey the NAV rules of the HCF as each frame transmitted under HCF by the HC or by a non-AP QSTA contains a duration value to cause STAs in the BSS to set their NAV to protect the frames expected to follow that frame.

All non-AP QSTAs shall be able to respond to QoS (+)CF-Polls received from an HC with the address 1 field matching their own address.

The HC shall perform delivery of buffered broadcast and multicast frames following DTIM beacons. The HC may also operate as a PC, providing (non-QoS) CF-Polls to associated CF-Pollable STAs using the frame formats, frame exchange sequences, and other applicable rules for PCF specified in 9.3.¹²

An HC may perform a backoff following an interruption of a frame exchange sequence due to lack of an expected response under the rules described in clause 9.9.2.1.3, using the parameters, dot11HCCWmin, dot11HCCWmax and dot11HCAIFSN, and the backoff rules in 9.1 and 9.9.1.5. The decision to perform a backoff by the HC is dependent on conditions such as interference from an overlapping BSS. The mechanism to detect the interference from an overlapping BSS and the decision to perform a backoff, DFS (such as in clause 11.6), or other techniques (such as inter-BSS scheduling) is beyond the scope of this standard.

9.9.2.1 HCF controlled channel access procedure

The HC gains control of the WM as needed to send QoS traffic to non-AP QSTAs and to issue QoS (+)CF-Polls to non-AP QSTAs by waiting a shorter time between transmissions than the stations using the EDCA procedures. The duration values used in QoS frame exchange sequences reserve the medium to permit completion of the current sequence.

The HC may include a CF Parameter Set element in the Beacon frames it generates. This causes the BSS to appear to be a point-coordinated BSS to STAs. This causes STAs to set their NAVs to the CFPDURRemaining value in the CF Parameter set element value at TBTT, as specified in 9.3.3.2. This prevents most contention in the CFP by preventing non-polled transmissions by non-AP STAs whether or not they are CF-Pollable.

9.9.2.1.1 CFP generation

HC may function as a point coordinator that uses the CFP for delivery, generating a CFP as shown in Figure 63, with the restriction that the CFP initiated by an HC shall always end with a CF-End frame. The HC may also issue QoS (+)CF-Polls to associated non-AP QSTAs during the CFP. However, because the HC can also grant polled TXOPs, by sending QoS (+)CF-Poll frames, during the CP, it is not mandatory for the HC to use the CFP for QoS data transfers.

Only a QAP that also issues non-QoS CF-Polls to associated CF-Pollable STAs may end a CFP with a CF-End+CF-Ack frame, and only in the case where the CF-End+CF-Ack is acknowledging a reception from a CF-Pollable non-QoS STA. The use of a non-QoS CF-Poll by a QAP to a non-AP QSTA is deprecated (for further discussion see 7.3.1.4).

9.9.2.1.2 CAP generation

When the HC needs access to the WM to start a CFP or a TXOP in CP the HC shall sense the WM. When the WM is determined to be idle for one PIFS period, the HC shall transmit the first frame of any permitted frame

¹² Attempting to intersperse HCF frame exchange sequences and PCF frame exchange sequences in a single CFP can be extremely complex.

exchange sequence, with the duration value set to cover the CFP or the TXOP. The first permitted frame in a CFP after a TBTT is the beacon. CAPs along with the CFPs and the CPs are illustrated in Figure 62.3.

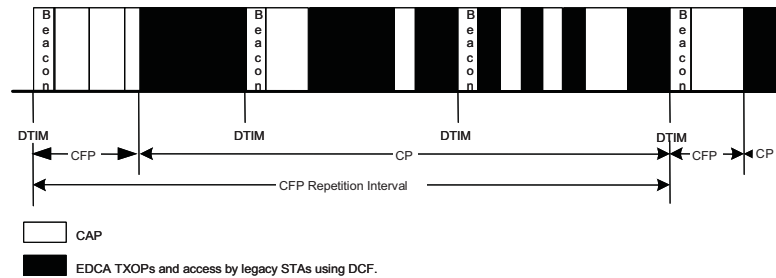


Figure 62.3 – CAP/CFP/CP periods

During a CFP or a TXOP in CP, after each data, QoS data or management type frame with a group address in the Address1 field, the HC shall wait for at least one PIFS period, and shall only continue to transmit if the WM is idle. After the last frame of all other non-final frame exchange sequences (e.g., sequences that convey unicast QoS data or management type frames) during a TXOP, the holder of the current TXOP shall wait for one SIFS period before transmitting the first frame of the next frame exchange sequence. HC may sense the channel and reclaim the channel after a duration of PIFS after the TXOP, if the channel remains idle.

9.9.2.1.3 Recovery from the absence of an expected reception

This subclause describes recovery from the absence of an expected reception in a CAP. It should be noted that the recovery rules from the absence of an expected reception are different from EDCA because in this case the NAVs of all the STAs in the QBSS have already been set up by the transmissions by the HC. The recovery rules for the continuation TXOP are different because a non-AP STA may always be hidden and may have not set its NAV due to the transmission by another non-AP STA. Finally, since an HC is collocated with the QAP, the QAP may recover using the rules described in this subclause even if it the recovery is from the absence of an expected reception.

If the beginning of reception of an expected response, as detected by the occurrence of PHY-CCA.indication(busy) at the QSTA that is expecting the response, does not occur during the first slot time following SIFS, then¹³

- a) If the transmitting station is the HC, it may initiate recovery by transmitting at a PIFS after the end of the last transmission.
- b) If the transmitting station is a non-AP QSTA, it shall initiate recovery by transmitting at a PIFS after the end of the last transmission, if the polled TXOP Limit is greater than 0 and at least one frame (re)transmissions can be completed within the remaining duration of a non-zero polled TXOP Limit.

If the transmitted frame is not of type QoS (+)CF-Poll and a PHY-CCA.indication(busy) occurs during the slot following SIFS, regardless of the occurrence of the PHY-RXSTART.indication, the QSTA may initiate recovery following the occurrence of PHY-CCA.indication(idle) such that a SIFS time interval occurs between the last energy on the air and the transmission of the recovery frame.

When there is a transmission failure within a polled TXOP, the frame retry counter corresponding to the TC of the failed frame shall be incremented. An MPDU belonging to a TC is subject to the respective retry limit as well as the dot11EDCATableMSDULifetime and is discarded when either of them is exceeded. An MPDU

¹³ This restriction is intended to avoid collisions due to inconsistent CCA reports in different QSTAs, not to optimize the bandwidth usage efficiency.

1 belonging to a TS with a specified delay bound is subject to delay bound and is discarded if the MPDU could
 2 not be transmitted successfully since it has been delivered to the MAC. An MPDU belonging to a TS with an
 3 unspecified delay is subject to `dot11MaxMSDULifetime` and is discarded when either of them is exceeded.

4 Non-AP QSTAs that receive a QoS (+)CF-Poll shall respond within a SIFS period. However, if a response is
 5 not received but a `PHY-CCA.indication(busy)` occurs during the slot following SIFS, and is followed by a
 6 `PHY-RXSTART.indication` or `PHY-RXEND.indication` prior to a `PHY-CCA.indication(idle)`, then the HC
 7 shall assume that the transmitted QoS (+)CF-Poll frame was successfully received by the polled non-AP
 8 QSTA. If the polled non-AP QSTA has no queued traffic to send, or if the MPDUs available to send are all
 9 too long to transmit within the specified TXOP limit, the non-AP QSTA shall send a QoS (+) Null frame. In
 10 the case of no queued traffic, this QoS (+) Null frame shall have a QoS control field that reports a queue size
 11 of 0 for any TID with the duration/ID set to the time required for the transmission of one ACK frame, plus
 12 one SIFS interval. In the case of insufficient TXOP size, such as in the case where the maximum MSDU size
 13 is not specified, this QoS (+) Null frame shall have a QoS control field that contains the TID and TXOP
 14 duration or a non-zero queue size needed to send the MPDU that is ready for transmission. In the case that a
 15 queue size is transmitted, the HC shall combine the queue size information with the rate of the received QoS
 16 (+) Null frame to determine the required size of the requested TXOP.

17 Within a polled TXOP, the unused portion of TXOPs shall be returned back to the HC. The recipient of the
 18 final frame, with the Ack policy set to Normal acknowledgement, shall be the HC if there will be time
 19 remaining in the TXOP after the transmission of the final frame and its expected ACK response. If there are
 20 no frames to be sent to the HC, then the non-AP QSTA shall send to the HC a QoS Null with queue size
 21 subfield in QoS Control set to 0. If the beginning of the reception of an expected ACK response to the final
 22 frame does not occur, detected as the non-occurrence of `PHY-CCA.indication(busy)` at the non-AP QSTA
 23 that is expecting the response during the first slot time following SIFS, the non-AP QSTA shall retransmit the
 24 frame or transmit a QoS Null frame, with the Ack policy set to Normal acknowledgement and the queue size
 25 field set to 0, after PIFS from the end of last transmission, until such time that it receives an
 26 acknowledgement or when there is not enough time remaining in the TXOP for sending such a frame. This is
 27 to avoid the situation where the HC may not receive the frame and may result in an inefficient use of the
 28 channel. If a `PHY-CCA.indication(busy)` occurs at the non-AP QSTA that is expecting the ACK response
 29 during the first slot following SIFS after the end of the transmission of the final frame, it shall be interpreted
 30 as indicating that the channel control has been successfully transferred and no further frames shall be
 31 transmitted by the non-AP QSTA in the TXOP, even though the ACK from HC may be incorrectly received.
 32 Note that while `PHY-CCA.indication(busy)` is used in this instance to determine the control of the channel, it
 33 is not used for determining the success or failure of the transmission.

34 9.9.2.2 TXOP structure and timing

35 Any QoS data type frame of a subclass that includes CF-Poll contains a TXOP limit in its QoS control field.
 36 The ensuing polled TXOP is protected by the NAV set by the Duration field of the frame that contained the
 37 QoS (+) CF-Poll function, as shown in Figure 62.4. Within a polled TXOP a QSTA may initiate the
 38 transmission of one or more frame exchange sequences, with all such sequences nominally separated by a
 39 SIFS interval. The QSTA shall not initiate transmission of a frame unless the transmission, and any
 40 acknowledgement or other immediate response expected from the peer MAC entity, are able to complete prior
 41 to the end of the remaining TXOP duration. All transmissions, including the response frames, within the
 42 polled TXOP are considered to be the part of the TXOP and the HC shall account for these when setting the
 43 TXOP limit. If the TXOP Limit subfield in the QoS Control field of the QoS data type frame that includes
 44 CF-Poll is set to 0, then the QSTA to which the frame is directed to shall respond with either one MPDU or
 45 one QoS Null frame.

46 A TXOP or transmission within a TXOP shall not extend across TBTT, `dot11CFPMaxDuration` (if during
 47 CFP), `dot11MaxDwellTime` (if using an FH PHY) or `dot11CAPLimit`. The HC shall ensure that the full
 48 duration of any granted TXOP meets these requirements so that non-AP QSTAs may use the time prior to the
 49 TXOP limit of a polled TXOP without checking for these constraints. Subject to these limitations, all

decisions regarding what MSDUs and/or MMPDUs are transmitted during any given TXOP are made by the QSTA that holds the TXOP.^{14 15}

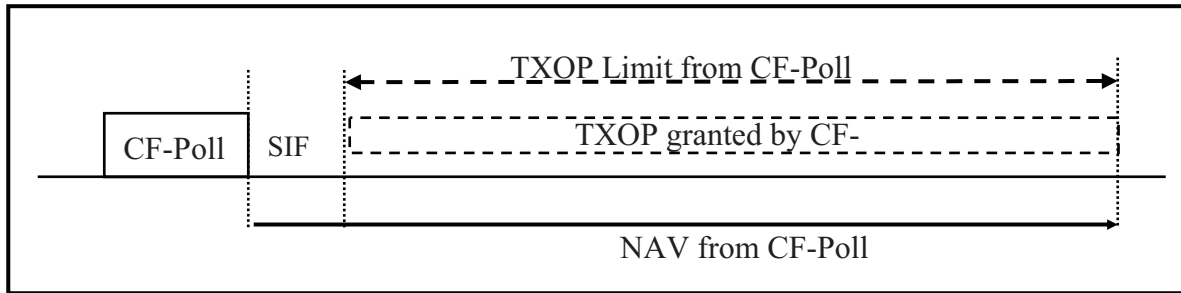


Figure 62.4 – Polled TXOP

9.9.2.2.1 NAV operation during a TXOP

A QSTA may track multiple NAV values. The NAV setting of the QSTA is the maximum of these NAV values for the QSTA. When a QSTA retains a new NAV value, that QSTA shall also save the BSSID (the AP address) from the frame that is setting the NAV value. A QSTA may retain a subset consisting of the longest NAV values; the shorter are discarded. A NAV value is removed when it expires. A NAV value is reset to 0 when a request for resetting the NAV comes, in the form of the CF-End frame or a QoS (+)CF-Poll frame with the Duration field set to 0, from the address stored for the NAV value.

When a QSTA sets a NAV value using the duration value from a QoS (+)CF-Poll containing the BSSID of a QBSS, that QSTA shall also save the TXOP holder address, which is the MAC address from the Address1 field of the frame containing the QoS (+)CF-Poll. If an RTS frame is received with the RA address matching the MAC address of the QSTA and the MAC address in the TA field in the RTS frame matches the saved TXOP holder address, then the QSTA shall send the CTS frame after SIFS, without regard for, and without resetting, its NAV. The saved TXOP holder address shall be cleared whenever the NAV is reset or when the NAV counts down to 0. A non-AP QSTA that receives a CF-End frame containing the BSSID of a QBSS shall reset the corresponding NAV value to 0. A QSTA shall track no more than one NAV and one TXOP holder pair per BSSID. A non-AP QSTA that receives a QoS (+) CF-Poll with a Duration/ID value equal to zero shall reset the NAV corresponding to the address in Address 1 of the received QoS CF-Poll. A NAV, TXOP holder pair is updated when such a request comes from the address stored for the TXOP holder or the BSSID.

In a CFP or CP, if the HC has no more QSTAs to poll and it has no more data, management, BlockAckReq or BlockAck frames to send, it may reset the NAVs of all QSTAs in the QBSS by sending a QoS CF-Poll with the RA matching its own MAC address and with the Duration/ID set to zero. As an HC also acts as a PC,

¹⁴ In certain regulatory domains, channel sensing must be done at periodic intervals (for example, in Japan, this period is 4 ms). This means that the duration of a TXOP in these regulatory domains might not be more than this periodic interval. If longer durations are desired then the TXOP holder needs to sense the channel at least once in the limit imposed in the regulatory domain, by waiting for at least for the duration of one PIFS during which it senses the channel. If it does not detect any energy, it may continue by sending the next frame. This means that the total TXOP size assigned should include an extra time allocated (i.e., $n \times \text{aSlotTime}$, where n is the number of times the QSTA needs to sense the channel and is given by $\text{floor}(\text{TXOP limit}/\text{limit imposed in the regulatory domain})$).

¹⁵ The TID value in the QoS Control field of a QoS Data(+)CF-Poll frame pertains only to the MSDU or fragment thereof in the Frame Body field of that frame. This TID value does not pertain to the TXOP limit value, and does not place any constraints on what frame(s) the addressed QSTA may send in the granted TXOP.

1 during the CFP, it may reset the NAVs of all STAs within the BSS (i.e., corresponding to the same BSSID)
2 by sending a CF-End frame, regardless of how the NAVs have been originally set.

3 When a QSTA receives a frame addressed to it and requires an acknowledgement, it shall respond with an
4 ACK or QoS +CF-Ack independent of its NAV. A non-AP QSTA shall accept a polled TXOP by initiating a
5 frame exchange sequence independent of its NAV.

6 9.9.2.3 HCF controlled channel access transfer rules

7 A TXOP obtained by receiving a QoS (+)CF-Poll uses the specified TXOP limit consisting of one or more
8 frame exchange sequences with the sole time-related restriction being that the final sequence shall end not
9 later than the TXOP limit. In frames of subtype QoS (+)CF-Poll, the TID subfield in the QoS Control field
10 indicates the TS or UP for which the poll is intended. MSDUs may be fragmented in order to fit within
11 TXOPs.

12 The QoS CF-Polls shall be sent only by an HC. Non-AP QSTAs are not allowed to send frames of subtype
13 (QoS) (+)CF-Poll. Non-AP QSTAs shall not send frames of subtype QoS (+)Data in response to any Data
14 frame other than the Data frames of subtype (QoS) (+)CF-Poll.

15 QSTAs shall use QoS data type frames for all MSDU transfers to another QSTA. The TID in the QoS control
16 fields of these frames shall indicate the TC or TS to which the MPDU belongs, and either the queue size field
17 shall indicate the amount of queued traffic present in the output queue that the QSTA uses for traffic
18 belonging to this TC or TS or the TXOP duration requested field shall indicate the duration that the QSTA
19 desires for the next TXOP for traffic belonging to this TC or TS. The queue size value reflects the amount on
20 the appropriate queue not including the present MPDU. A non-AP QSTA that wishes to inform the HC of
21 queue status, but does not have a data frame to send to the HC, may use the QoS Null indicating the TID and
22 the queue size or TXOP duration request (also see 9.9.2.3.1).

23 QSTAs shall be able to receive QoS +CF-Ack frames. QoS Data+CF-Ack frames shall be sent by the HC
24 only if it wants to send frames to the same or any other non-AP QSTA a SIFS after receiving the final
25 transmission of the previous TXOP. In both CFP and CP, QSTAs shall always respond to QoS data frames,
26 having the Ack Policy subfield in QoS Control field set to "Normal Acknowledgement", with an ACK frame
27 unless the acknowledgement is piggybacked in which case it shall use QoS +CF-ACK. Piggybacked frames
28 are allowed only in CFP or within TXOPs initiated by the HC. The HC shall not send a QoS data type frame
29 containing a +CF-ACK with an address 1 that does not correspond to the address of the QSTA for which the
30 +CF-ACK is intended, unless the QSTA, to which the +CF-Ack is intended, sets the Q-Ack subfield in the
31 QoS Capability information element in the (re)association request frame. QSTAs are not required to be able
32 to transmit QoS data type frames with subtypes that include +CF-Ack. A QAP that has
33 dot11QACKOptionImplemented true may allow associations with stations advertising support for the Q-Ack
34 option and such associations do not force the QAP to then be required to employ piggyback
35 acknowledgements directed toward that associated station in frames that are not directed to that associated
36 station. QSTAs shall be able to process received QoS data type frames with subtypes that include +CF-Ack
37 when the QSTA to which the acknowledgement is directed is the same as the QSTA addressed by the
38 Address1 field of that QoS data type frame. QSTAs are not required to handle received QoS data type frames
39 in which the +CF-Ack function pertains to a different QSTA than is addressed by the Address1 field of that
40 QoS data type frame, unless the QSTA to which the +CF-Ack is intended sets the Q-Ack subfield in the QoS
41 Capability information element in the (re)association request frame. The net effect of these restrictions on the
42 use of QoS +CF-Ack is that the principal QoS +CF-Ack subtype that is useful is the QoS Data+CF-Ack,
43 which can be sent by a non-AP QSTA as the first frame in a polled TXOP when that TXOP was conveyed in
44 a QoS Data+CF-Poll(+CF-Ack) and the outgoing frame is directed to the HC's QSTA address. QoS
45 (Data+)CF-Poll+CF-Ack frames are only useful if the HC wants to grant another TXOP to the same non-AP
46 QSTA a SIFS after receiving the final transmission of that non-AP QSTA's previous TXOP.

1 HCF contention-based channel access shall not be used to transmit MSDUs belonging to an established traffic
 2 stream (with the HC's acceptance of the associated TSPEC), unless the granted TSPEC indicates it is
 3 permitted to do so when the Access Policy subfield of the TS Info field is set to "Both controlled and
 4 contention-based channel access.

5 When frames associated with a TSPEC are transmitted over contention-based channel access, QSTAs shall
 6 use the AC associated with the UP specified in the TSPEC.

7 If an HC receives a frame from a QSTA with duration/ID covering only the response frame, the HC shall
 8 assume that the QSTA is terminating its TXOP and the HC may initiate other transmissions, or allow the
 9 channel to go into the contention period.

10 **9.9.2.3.1 TXOP requests**

11 Non-AP QSTAs may send TXOP requests during polled TXOPs or EDCA TXOPs using the QoS Control
 12 field in the QoS Data, or QoS Null frame directed to the HC, with the TXOP request duration or queue size
 13 and TID value indicated to the HC. QAPs indicate whether they process TXOP request or queue size in the
 14 QoS Capability Information Element in the Beacon, probe response and (re)association response frames.
 15 QAPs shall process requests in at least one format. Non-AP QSTAs shall only use the request format that the
 16 QAP indicates that it can process.

17 If the value of TXOP duration requested, in a QoS Data type frame, is zero, the HC shall continue to poll
 18 according to the negotiated schedule.

19 **9.9.2.3.2 Use of RTS/CTS**

20 QSTAs may send an RTS as the first frame of any frame exchange sequence for which improved NAV
 21 protection is desired, during either the CP or CFP, and without regard for dot11RTSThreshold.¹⁶

22 If a QSTA sends an RTS and does not receive an expected CTS, then the recovery rules are as specified in
 23 9.9.2.1.3.

24 If NAV protection is desired for a transmission to the QAP in response to a QoS data frame with a subtype
 25 that includes CF-Poll, the polled non-AP QSTA is allowed to send a CTS frame (as a CTS frame is shorter
 26 than a QoS data frame and can be sent with a higher probability of it receiving by other STAs) with the RA
 27 containing its own MAC address in order to set the NAV in its own vicinity without the extra time to send an
 28 RTS¹⁷.

29 **9.9.3 Admission Control at the HC**

30 An IEEE 802.11[®] network may use admission control to administer policy or regulate the available
 31 bandwidth resources. Admission control is also required when a QSTA desires guarantee on the amount of
 32 time that it can access the channel. The HC, which is in the QAP, is used to administer admission control in
 33 the network. As the QoS facility supports two access mechanisms, there are two distinct admission control
 34 mechanisms – one for contention-based access and another for controlled-access mechanism.

¹⁶ The sending of RTS during the CFP is usually unnecessary, but may be used to ensure that the addressed recipient QSTA is within range and awake, and to elicit a CTS response that sets the NAV at STAs in the vicinity of the addressed recipient. This is useful when there are nearby STAs that are members of other BSSs and are out of range to receive beacons from this BSS. Sending an RTS during the CFP is only useful when the recipient is a QSTA, because a non-QoS STA in the same BSS has its NAV set to protect the CFP, which renders those non-QoS STAs unable to respond. Using the same duration calculation during the CFP as specified for the CP is directly applicable for all cases except when the RTS is sent by the HC, and the following frame includes a QoS (+)CF-Poll.

¹⁷ This is unnecessary because the NAVs in the vicinity of the QAP were set by the QoS (+)CF-Poll frame.

Admission control, in general, depends on vendors' implementation of the scheduler, available channel capacity, link conditions, retransmission limits, and the scheduling requirements of a given stream. All of these criteria affect the admissibility of a given stream; any stream may be rejected. If the HC has admitted no streams that require polling, it may not find it necessary to perform the scheduler or related HC functions.

9.9.3.1 Contention-based Admission Control Procedures

A non-AP QSTA may support admission control procedures. QAPs shall support admission control procedures, at least to the minimal extent of advertising that admission is not mandatory on its ACs.

The QAP uses the ACM (admission control mandatory) subfields advertised in the EDCA Parameter Set element to indicate whether admission control is enabled for each of the ACs. While the CWmin, CWmax, AIFS, TXOP limit parameters may be adjusted over time by the QAP, the ACM bit shall be static for the duration of the lifetime of the BSS. An ADDTS request shall be transmitted by a non-AP QSTA to the HC in order to request admission of traffic in any direction (uplink, downlink, direct, or bidirectional) employing an AC that requires admission control. The ADDTS request shall contain the user priority associated with the traffic and shall indicate EDCA as the access policy. The QAP shall associate the received user priority of the ADDTS request with the appropriate AC as per the UP to AC mappings described elsewhere in this document. The non-AP QSTA may transmit unadmitted traffic for those ACs for which the QAP does not require admission control. If a QSTA desires to send data without admission control using an AC that mandates admission control, the QSTA shall use an AC that corresponds to a lower priority and does not require admission control. All ACs with priority higher than that of an AC with ACM flag equal to 1 should have ACM flag set to 1. If an ADDTS request has not been accepted for downlink traffic with a UP that maps to an AC with ACM equal to 1, the QAP should lower the UP of such traffic so that it is transmitted in an AC that does not require admission control.

9.9.3.1.1 Procedures at the QAP

The QAP shall respond to requests for admission conveyed in the ADDTS requests for those AC with the ACM subfield set to 1. On receipt of an ADDTS request conveyed in either ADDTS request frame or Reassociation Request frame from an associated non-AP QSTA, the QAP shall make a determination as to whether to

- a) accept the request
- b) deny the request

The QAP may use any algorithm in making such a determination. If the QAP decides to accept the request, the QAP shall also derive the Medium Time from the information conveyed in the TSPEC element in the ADDTS request. The QAP may use any algorithm in deriving the Medium Time, but subclause H.1.2 provides a procedure that may be used. Having made such a determination, the QAP shall transmit a TSPEC element to the requesting non-AP QSTA contained in an ADDTS response frame or (re)association frame. If the QAP is accepting the request, the Medium Time field shall be specified.

The QAP may choose at any time to tear down an explicit EDCA-policy admission by sending a DELTS frame.

9.9.3.1.2 Procedure at non-AP QSTAs

Each channel access function shall maintain two variables. The first of these is the admitted time, and the second is the used time.

The admitted time and used time shall be set to zero at the time of association. The non-AP QSTA may subsequently decide, to explicitly request admission amount for the AC, which is associated with the specified priority.

In order to make such a request, the non-AP QSTA shall transmit a TSPEC element contained in an ADDTS request management frame with the following fields specified (i.e. non-zero): Nominal MSDU Size, Mean

1 Data Rate, Minimum PHY Rate, and Surplus Bandwidth Allowance. The Medium Time field is not used in
2 the request frame and shall be set to zero.

3 On receipt of a TSPEC element, contained in a ADDTS response frame, indicating that the request has been
4 accepted, the non-AP QSTA shall set the admitted time variable for the specified channel access function to
5 the value contained in the Medium Time field of that element.

6 The non-AP QSTA may choose to tear down the explicit request at any time. For the tear down of an explicit
7 admission, the non-AP QSTA shall transmit a DELTS frame containing the TSPEC element to the QAP.

8 If the non-AP QSTA receives a DELTS frame with the action field set to DELTS it shall disable the use of
9 the specified AC.

10 If an explicit admission is torn down, the non-AP QSTA shall disable the use of the AC. The admitted time
11 and the used time for the AC shall both be set to zero.

12 The non-AP QSTA shall update the value of used time:

13 a) at one second intervals

14 $\text{used_time} = \max((\text{used_time} - \text{admitted_time}), 0)$

15 b) after each successful or unsuccessful MPDU (re)transmission attempt,

16 $\text{used_time} = \text{used_time} + \text{MPDUExchangeTime}$

17 where MPDUExchangeTime corresponds to the just completed MPDU exchange.
18

19 The MPDUExchangeTime equals the time required to transmit the MPDU sequence. For the case of an
20 MPDU transmitted with normal acknowledgement policy and without RTS-CTS protection, this equals the
21 time required to transmit the MPDU plus the time required to transmit the expected response frame plus one
22 SIFS, plus one aSlotTime if the transmission was unsuccessful. If the used time reaches or exceeds the
23 admitted time value, the corresponding channel access function shall no longer transmit using the EDCA
24 parameters for that AC as specified in the QoS parameter set element. However, a non-AP QSTA may choose
25 to temporarily replace the EDCA parameters for that channel access function with those specified for an
26 access category of lower priority, if no admission control is required for those access categories.

27 If, for example, a non-AP QSTA has made and had accepted an explicit admission for a AC, and the channel
28 conditions subsequently worsen, possibly including a change in PHY layer data rate, such that it requires
29 more time to send the same data, the non-AP QSTA may make a request for more admitted time to the QAP,
30 and at the same time downgrade the EDCA parameters for that AC for short intervals in order to send some of
31 the traffic at the admitted priority, and some at the unadmitted priority, whilst waiting for a response to the
32 admission request.

33 9.9.3.2 Controlled-access Admission Control

34 This subclause describes the schedule management of the admitted HCCA streams by the HC. When the HC
35 provides controlled channel access to non-AP QSTAs, it is responsible for granting or denying polling service
36 to a TS based on the parameters in the associated TSPEC. If the TS is admitted, the HC is responsible for
37 scheduling channel access to this TS based on the negotiated TSPEC parameters. The HC should not initiate a
38 modification of TSPEC parameters of an admitted TS unless requested by the STA. The HC should not tear
39 down a TS unless explicitly requested by the STA or at the expiry of the inactivity timer. The polling service
40 based on admitted TS provides a “guaranteed channel access” from the scheduler in order to have its QoS
41 requirements met. This is an achievable goal when the wireless medium operates free of external interference
42 (such as operation within the channel by other technologies and co-channel overlapping BSS interference).
43 The nature of wireless communications may preclude absolute guarantees to satisfy QoS requirements.
44 However, in a controlled environment (e.g. no interference), the behavior of the scheduler can be observed
45 and verified to be compliant to meet the service schedule.

46 The normative behavior of the scheduler is as follows:

- 47 — The scheduler shall be implemented such that, under controlled operating conditions, all stations with
48 admitted TS are offered TXOPs that satisfy the service schedule.

— Specifically, if a TS is admitted by the HC, then the scheduler shall service the non-AP QSTA during a service period (SP). An SP is a contiguous time during which a set of one or more downlink frames and/or one or more polled TXOPs are granted to the QSTA. An SP starts at fixed intervals of time specified in Service Interval field. The first SP starts when the lower order 4 bytes of the TSF timer equals the value specified in Service Start Time field¹⁸. An SP ends after; 1) expiration of the time duration specified in Max Service Duration field; or 2) if the non-AP QSTA receives a frame with EOSP field set to 1 in QoS control field during an SP. Additionally, the Minimum TXOP duration shall be at least the time to transmit one Maximum MSDU Size successfully at the minimum PHY rate specified in the TSPEC. If Maximum MSDU Size is not specified in the TSPEC then the minimum TXOP duration shall be at least the time to transmit one Nominal MSDU Size successfully at the minimum PHY rate. The vendors are free to implement any optimized algorithms, such as reducing the polling overheads, increasing the TXOP duration etc, within the parameters of the transmitted schedule.

The HC may aggregate admitted HCCA TS for a single non-AP QSTA and establishes a Service Schedule for the non-AP QSTA. When the HC aggregates the admitted TS, it shall set the Aggregation field in the granted TSPEC and Service Schedule to 1. The HC shall provide an aggregate Service Schedule if the non-AP QSTA sets the Aggregation field in its TSPEC request. If the QAP establishes an aggregate Service Schedule for a non-AP QSTA, it shall aggregate all HCCA streams for the QSTA. The Service Schedule is communicated to the non-AP QSTA in a Schedule element contained in an ADDTS response.

The HC may update the Service Schedule at any time by sending a Schedule element in a Schedule frame. The updated schedule is in effect when the HC receives the acknowledgement frame for the Schedule frame. The modified Service Start Time shall not exceed the requested Service Start Time by more than one Minimum Service Interval. The Service Start Time in the Schedule Element in the Schedule frame shall not exceed the beginning of the immediately previous SP by more than the maximum service interval. The Service Start Time shall not precede the beginning of the immediately previous SP by more than the minimum service interval.

A non-AP QSTA may affect the Service Schedule by modifying or deleting its existing TS as specified in clause 11.4.

Clause H.3.1 provides guidelines for deriving an aggregate Service Schedule for a single non-AP QSTA from the non-AP QSTA's admitted TS. The Schedule shall meet the QoS requirements specified in the TSPEC.

During any time interval $[t1, t2]$ including the interval that is greater than specification interval, the cumulative TXOP duration shall be greater than the time required to transmit all MSDUs (of Nominal MSDU Size) arriving at the mean data rate for the stream, over the period $[t1, t2-D]$. The parameter D is set to the specified Maximum Service Interval in the TSPEC. If Maximum Service Interval is not specified, then D is set to the Delay Bound in the TSPEC.

The HC shall use Minimum PHY rate in calculating TXOPs if Minimum PHY rate is present in the TSPEC field in the ADDTS response. Otherwise, the HC may use an observed PHY rate in calculating TXOPs.

A non-AP QSTA operating in power-save mode shall be in Awake state before the start of an SP and it shall remain in active mode until the SP ends. The end of SP is indicated to a non-AP QSTA by the HC by setting EOSP bit in a QoS frame. This bit indicates that the current SP ends after the end of current frame exchange sequence. In absence of indication by EOSP bit, a SP ends after the Max Service Duration from the scheduled SP start time. The indication of the end of an SP facilitates power management. It may also be used as input to uplink traffic stream management functions like scheduling and queue management. These functions are implementation dependent and are outside the scope of this standard.

¹⁸ The low order 4 bytes of the TSF timer cover a span of around 71 minutes. Due to TSF timer wrap over, and due to possibility of receiving the schedule frame after the indicated start time timer, ambiguity may occur. This ambiguity is resolved by using the nearest absolute TSF timer value in past or future when the four low order bytes match the Start Time field in the Schedule Element.

A minimum set of TSPEC parameters shall be specified during the TSPEC negotiation. The specification of a minimum set of parameters is required so that the scheduler can determine a schedule for the stream that is to be admitted. These parameters are Mean Data Rate, Nominal MSDU Size and at least one of Maximum Service Interval and Delay Bound in the ADDTS Request. In the ADDTS Response, these parameters are Mean Data Rate, Nominal MSDU Size and Maximum Service Interval and shall be non-zero when a stream is admitted.

If any of the elements in the minimum set of parameters do not have the required non-zero values as specified above in the ADDTS Request, the HC may reject the stream or admit it with an alternative minimum set of parameters. If the HC admits the stream with the alternative set of TSPEC parameters, these parameters are indicated to the non-AP QSTA through the ADDTS response. If both Maximum Service Interval and Delay Bound are specified, the HC may use only the Maximum Service Interval. If any other parameter is specified in the TSPEC element, the scheduler may use it when calculating the schedule for the stream. The HC may also use the UP value in the TS Info field for admission control or scheduling purposes, however this decision is outside the scope of the standard. The mandatory set of parameters can be set by any higher layer entity or may be generated autonomously by the MAC.

If a STA specifies a non-zero Minimum Service Interval, and if the TS is admitted, the HC shall generate a schedule that conforms to the specified Minimum Service Interval.

A reference Design for an example Scheduler and Admission Control Unit is provided in Annex H. An example use of the TSPEC for admission control is also described in Annex H.

9.10 Block acknowledgment

9.10.1 Introduction (Informative)

The Block Acknowledgement (Block Ack) mechanism allows a block of QoS Data MPDUs to be transmitted, each separated by a SIFS period, between two QSTAs. The mechanism is for improving the channel efficiency by aggregating several acknowledgements into one frame. There are two types of Block Ack mechanisms: immediate and delayed. Immediate Block Ack is suitable for high-bandwidth, low latency traffic while the delayed Block Ack is suitable for applications that tolerate moderate latency.¹⁹ In this clause, the QSTA with data to send using the Block Ack mechanism is referred to as the originator and the receiver of that data as the recipient.

The Block Ack mechanism is initialized by an exchange of ADDBA request/response frames. After initialization, blocks of QoS data type frames can be transmitted from the originator to the recipient. A block may be started within a polled TXOP or by winning EDCA contention. The number of frames in the block is limited, and the amount of state that must be kept by the recipient is bounded. The MPDUs within the block of frames are acknowledged by a BlockAck control frame, which is requested by a BlockAckReq control frame.

The Block Ack mechanism does not require setting up of a TS; however QSTAs using the TS facility may choose to signal their intention to use Block Ack mechanism for the scheduler's consideration in assigning TXOPs. Acknowledgements of frames belonging to the same TID, but transmitted during multiple TXOPs, may also be combined into a single Block Ack frame. This mechanism allows the originator to have flexibility regarding the transmission of Data MPDUs. The originator may split the block of frames across TXOPs, separate the data transfer and the block acknowledgement exchange and interleave blocks of MSDUs for different TIDs or RAs.

Figure 62.8 illustrates the message sequence chart for the set up, data and block acknowledgement transfer and the tear down of Block Ack mechanism and are discussed in detail in the following subclauses.

¹⁹ The delayed Block Ack mechanism is primarily intended to allow existing implementations to use this feature with minimal hardware changes and also to allow inexpensive implementations that would use the processing power on the host.

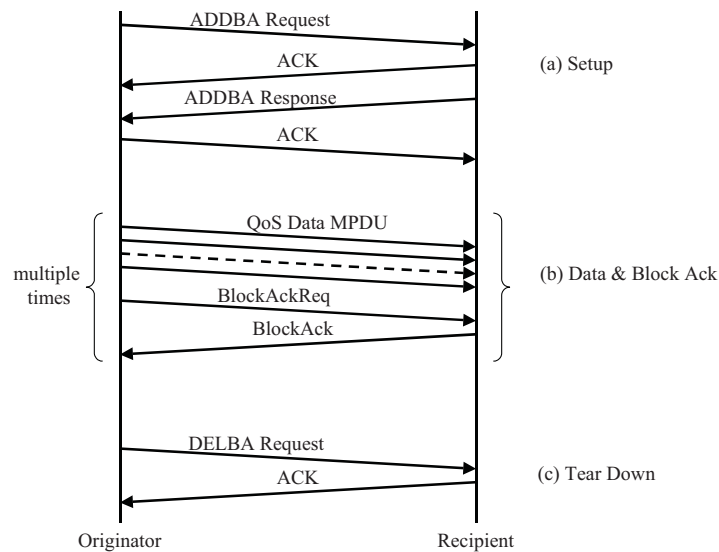


Figure 62.8 – Message Sequence Chart for Block Ack Mechanism: (a) Set up, (b) Data and acknowledgement transfer and (c) Tear down

9.10.2 Set up and modification of the Block Ack parameters

A QSTA that intends to use the Block Ack mechanism for the transmission of QoS Data frames to a peer should first check if the intended peer QSTA is capable of participating in Block Ack mechanism by discovering and examining its Block Ack capability bit. If the intended peer QSTA is capable of participating, the originator sends an ADDBA request frame indicating the TID for which the Block Ack is being set up. The Block Ack policy and the Buffer Size fields in the ADDBA Request frame are advisory and may be changed by the recipient. The receiving QSTA shall respond by an ADDBA response frame. The receiving QSTA, which is the intended peer, has the option of accepting or rejecting the request. When the QSTA accepts, it indicates the type of Block Ack and the number of buffers that it shall allocate for the support of this block. If the receiving QSTA rejects the request, then the originator shall not use the Block Ack mechanism.

If the Block Ack mechanism is being set up for a TS, bandwidth negotiation (using ADDTS request and response QoS Action management frames) should precede the set up of the Block Ack mechanism.

Once the Block Ack exchange has been set up, Data and acknowledgements are transferred using the procedure described in clause 9.10.3.

9.10.3 Data and acknowledgement transfer

After setting up for the Block exchange following the procedure in 9.10.2, the originator may transmit a Block of QoS data type frames separated by SIFS period, with the total number of frames not exceeding the Buffer Size subfield in the associated ADDBA response frame. Each of the frames shall have the Ack policy subfield in the QoS Control set to “Block Acknowledgement”. The RA field of the frames shall be the recipient's *unicast* address. The originator requests acknowledgement of outstanding QoS Data type frames by sending a BlockAckReq frame. The recipient shall maintain a Block Ack record for the block.

Subject to any constraints herein about permitted use of TXOP according to the channel access mechanism used: the originator may separate the Block and the BlockAckReq into separate TXOPs; the originator may

split a Block across multiple TXOPs; the originator may sequence frames with different TIDs in the same TXOP; the originator may interleave MPDUs with different TIDs within the same TXOP; the originator may sequence or interleave MPDUs for different RAs within a TXOP. A protective mechanism (such as transmitting using HCCA, using RTS/CTS or the mechanism described in 9.10) should be used to reduce the probability of other stations transmitting during the TXOP. If no protective mechanism is used, then the first frame that is sent as a block shall have a response frame and shall have the duration field set such that the NAVs are set to appropriate values at all STAs in the QBSS.

The originator shall use the Block Ack Starting Sequence Control to signal the first MPDU in the block for which an acknowledgement is expected. MPDUs in the recipient's buffer with a sequence control value that precedes the Starting Sequence Control value are called "preceding" MPDUs. The recipient shall reassemble any complete MSDUs from buffered preceding MPDUs and indicate these to its higher layer. The recipient shall then release any buffers held by preceding MPDUs. The range of the outstanding MPDUs (i.e. the reorder buffer) must begin on an MSDU boundary. The total number of frames that can be sent depends on the total number of MPDUs in all the outstanding MSDUs. The total number of MPDUs in these MSDUs may not exceed the reorder buffer size in the receiver.

The recipient shall maintain a block acknowledgement record consisting of originator address, TID, and a bitmap of the size of Re-ordering Buffer Size indexed by received MPDU sequence control value, where the MPDU sequence control value is defined as (Sequence Number * 16 + Fragment number). This record holds the acknowledgement state of the data frames received from the originator.

If the immediate Block Ack policy is used, the recipient shall respond to a BlockAckReq, with a BlockAck frame. If the recipient sends the BlockAck frame, the originator updates its own record and retries any frames that are not acknowledged in the BlockAck frame, either in another block or individually.

If the delayed Block Ack policy is used, the recipient shall respond to BlockAckReq with an ACK frame. The recipient shall then send its BlockAck response in a subsequently obtained TXOP. Once the contents of the BlockAck frame have been prepared, the recipient shall send this frame in the earliest possible TXOP using the highest priority AC. The originator shall respond with an ACK frame upon receipt of the BlockAck frame.

The BlockAck contains acknowledgements for the MPDUs of up to 64 previous MSDUs. If the BlockAck indicates that an MPDU was not received correctly, the originator shall retry that MPDU subject to that MPDU's appropriate retry limit.

A typical Block Ack frame exchange sequence using the immediate Block Ack, for a single TID is shown in figure 62.9.

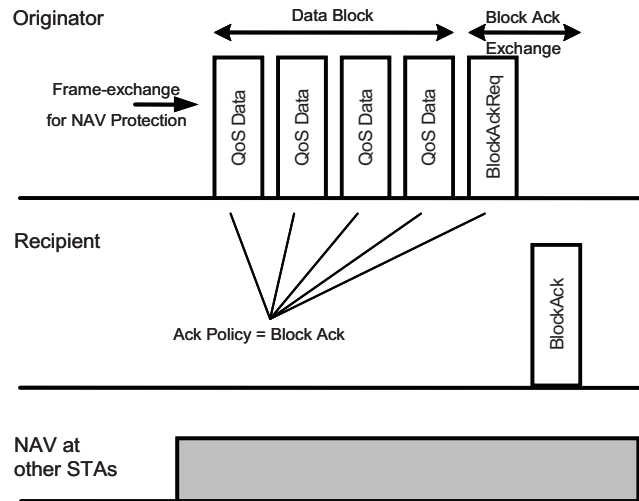


Figure 62.9 – A typical Block Ack sequence when immediate policy is used

A typical Block Ack sequence using the delayed Block Ack is shown in Fig 62.10.

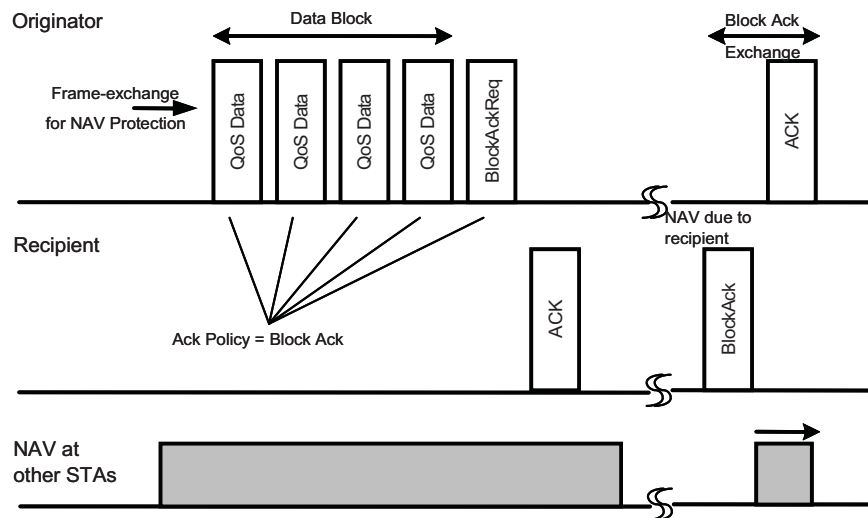


Figure 62.10 – A typical Block Ack sequence when delayed policy is used

If there is no response (i.e., neither a BlockAck nor an ACK) to the BlockAckReq frame, the originator may retransmit the BlockAckReq within the current TXOP (if time permits) or within a subsequent TXOP. If the BlockAckReq frame is associated with MPDUs belonging to a TC, then the retransmission of the BlockAckReq frame is subject to the dot11ShortRetryLimit. If the BlockAckReq frame is associated with MPDUs belonging to a TS, then the frame shall be discarded if any MSDUs referenced by this BlockAckReq frame have been discarded from the transmit buffer due to expiry of their lifetime limit.

The BlockAckReq shall be discarded if all MSDUs referenced by this BlockAckReq have been discarded from the transmit buffer due to expiry of their lifetime limit.

In order to improve efficiency, originators using the Block Ack facility may send MPDU frames with Ack policy in QoS Control frames set to "Normal Acknowledgement" if only a few MPDUs are available for

transmission. When there are sufficient number of MPDUs, the originator may switch back to the use of Block Ack. No Block Ack parameters, however, shall be changed without renegotiation.

The frame exchange sequences are provided in 9.12.

9.10.4 Receive buffer operation

The recipient flushes received MSDUs from its receive buffer as described in this subclause.

If a BlockAckReq is received, all complete MSDUs with lower sequence numbers than the starting sequence number contained in the BlockAckReq shall be indicated to the MAC client using the MA-UNIDATA.indication primitive.

If, after an MPDU is received, the receive buffer is full, the complete MSDU with the earliest sequence number shall be indicated to the MAC client using the MA-UNIDATA.indication primitive.

All comparisons of sequence numbers are performed circularly modulo 2^{12} .

The recipient shall always indicate the reception of MSDU to its MAC Client in order of increasing sequence number.

9.10.5 Tear down of the Block Ack mechanism

When the originator has no data to send and the final Block Ack exchange has completed, it shall signal the end of its use of the Block Ack mechanism by sending the DELBA frame to its recipient. There is no management response frame from the recipient.²⁰ The recipient of the DELBA frame shall release all resources allocated for the Block Ack transfer.

9.10.6 Error recovery upon a peer failure

An originator or a recipient shall assume that there is a peer failure, if its peer fails to respond within Block Ack Timeout Value. An originator failure is detected if there is no Data or BlockAckReq MPDU received from it using the TID for the duration of the timeout. A recipient failure is detected if there is no BlockAck MPDU received from it using the TID for the duration of the timeout.

When a timeout is detected, the QSTA shall act as though a DELBA frame has been received.

If the timeout is detected at the recipient and the originator subsequently attempts to continue using the BlockAck mechanism with this TID, then the recipient shall ignore any Data MPDUs sent using this TID and shall send a DELBA frame. The originator may attempt to set up the use of Block Ack or may send the MPDUs using an alternative acknowledgement mechanism.

9.11 No acknowledgement

The usage of No Acknowledgement is determined by the policy at the QSTA. When No Acknowledgement policy is used, there is no MAC-level recovery and the reliability of this traffic is reduced, relative to the reliability of traffic with other acknowledgement policies, due to the increased probability of lost frames from interference, collisions, or time-varying channel parameters. A protective mechanism (such as transmitting using HCCA, using RTS/CTS or the mechanism described in 9.10) should be used to reduce the probability of other stations transmitting during the TXOP.

²⁰ Normal acknowledgement rules apply.

1 *Insert the contents of 9.7 of the IEEE 802.11[®], 1999 (Reaffirm) standard here and renumber the subclause*
2 *as shown, renumbering tables and figures as necessary:*

3 **9.12 Frame exchange sequences**

4 *Change the text as shown*

5 The allowable frame exchange sequences are summarized in Table 21 ~~and Table 22, 22.1, 22.2, 22.3 and~~
6 22.4. A legend applicable to ~~both~~ all the tables follows Table 22.4. The frame exchange sequences are also
7 illustrated in Figures 62.11-62.16 in the form of HMSCs.

Insert the following entries at the end of Table 21 as shown:

<HCF-Sequence>	1 or more	Start of frame sequences initiated by a QSTA
----------------	-----------	--

Insert the following entries at the end of Table 22 as shown:

<HCF-Sequence>	1 or more	Start of frame sequences initiated by a QSTA
----------------	-----------	--

Move the Legend behind Table XX+1:

Insert the following tables into the subclause:

Table 22.1 – <HCF sequence>

HCF frame Sequence	Frames in Sequence	Usage
{CTS-}[Data (bc/mc) =] Data(bc/mc)	1 or more	Broadcast or multicast MSDU sent by a QAP
{CTS-}[QoS Data (bc/mc) =] QoS Data (bc/mc)	1 or more	Broadcast or multicast MSDU sent by a QAP as QoS Data frames
{CTS-}[Mgmt (bc) =] Mgmt(bc)	1 or more	Broadcast MMPDU sent by a QAP
{CTS-}<TXOP-Sequence> [– <TXOP-Sequence>]	1 or more	Start of a TXOP sequence using contention
QoS CF-Poll self	1	QoS CF-Poll frame sent by the HC with the destination address same as the HC's MAC address.
<CF-Ack-Piggybacked-QoS-Poll-Sequence>	2 or more	Poll started by a QoS CF-Poll onto which an acknowledgement is piggybacked.
{RTS – CTS – } <Non-CF-Ack-Piggybacked-QoS-Poll-Sequence>	2 or more	Poll started by a QoS CF-Poll onto which an acknowledgement is not piggybacked.

Table 22.2 – <Poll sequence>

Poll Sequence (If Piggybacked CF-Ack is present it is a CF-Ack-Piggybacked-QoS-Poll-Sequence else it is a Non-CF-Ack-Piggybacked-QoS-Poll-Sequence)	Frames in Sequence	Usage
QoS CF-Poll {+CF-Ack } (no data) { – CTS self } – <TXOP-Sequence> [– <TXOP-Sequence>] { – QoS Null(dir, normal ack)(no data) – ACK }	2 or more	Start of a TXOP sequence as initiated with a separate poll from HC. (*1), (*3), (*4)
QoS CF-Poll {+CF-Ack } (no data) – QoS Null(dir, normal ack)(no data) – ACK	3	Separate QoS CF-Poll from an HC to a QSTA with empty queue, insufficient time for queued MPDU, or too little time remaining before a dwell or medium occupancy boundary to send a queued frame (*1)

QoS Data(dir,normal ack)+CF-Poll{+CF-Ack} – <CF-Ack-Piggybacked-QoS-Data-Sequence>	3 or more	Start of a TXOP sequence as initiated with a poll from HC that is piggybacked onto a data frame. The ack policy of the data frame is set to normal ack. The response frame is an ack piggybacked onto the data frame with ack policy set to normal acknowledgement. (*1), (*2), (*3)
QoS Data(dir,normal ack)+CF-Poll{+CF-Ack} – ACK – <TXOP-Sequence> [– <TXOP-Sequence>] { – QoS Null(dir, normal ack)(no data) – ACK }	2 or more	Start of a TXOP sequence as initiated with a poll from HC that is piggybacked onto a data frame. The ack policy of the data frame is set to normal ack. The response frame is an ACK frame. (*1), (*3)
QoS Data(dir,normal ack)+CF-Poll{+CF-Ack} – ACK – QoS Null(dir, normal ack)(no data) – ACK	4	QoS Poll piggybacked onto a QoS Data frame with ack policy set to normal ack, from the HC to a QSTA with empty queue, insufficient time for queued MPDU, or too little time remaining before a dwell or medium occupancy boundary to send a queued frame (*1)
QoS Data(dir,block ack no ack)+CF-Poll{+CF-Ack} { – CTS self } – <TXOP-Sequence> [– <TXOP-Sequence>] { – QoS Null(dir, normal ack)(no data) – ACK }	2 or more	Start of a TXOP sequence as initiated with a poll from the HC with CF-Ack with ack policy set to block ack or no ack. (*1), (*3), (*4)
QoS Data(dir,block ack no ack)+CF-Poll{+CF-Ack} – QoS Null(dir, normal ack)(no data) – ACK	3	QoS Poll from the HC to a QSTA with empty queue, insufficient time for queued MPDU, or too little time remaining before a dwell or medium occupancy boundary to send a queued frame (*1)

1

2

Table 22.3 – <TXOP sequence>

TXOP frame Sequence	Frames in Sequence	Usage
{RTS – CTS – } Mgmt(dir) – ACK	2	Directed MMPDU
{RTS – CTS – } Data(dir) – ACK	2	Directed MPDU, transmitted by the QAP only
{RTS – CTS – } QoS Data(dir, normal ack) – ACK	2	Directed MPDU with normal acknowledgement policy responded by an ACK
{RTS – CTS – } QoS Data(dir, normal ack) – <CF-Ack-Piggybacked-QoS-Data-Sequence>	2 or more	Directed MPDU with normal acknowledgement policy responded with a new TXOP
{RTS – CTS – } QoS Data(dir, normal ack) – <CF-Ack-Piggybacked-QoS-Poll-Sequence>	2 or more	Directed MPDU with normal acknowledgement policy responded with a new TXOP
{RTS – CTS – } QoS Data(dir, normal ack) – Data+CF-Ack	2	Directed MPDU with normal acknowledgement policy responded with a Data+CF-Ack
{RTS – CTS – } QoS Data(dir, no ack)	1	Directed MPDU with no acknowledgement policy

{RTS – CTS – } QoS Data(dir, block ack)	1	Directed MPDU with block acknowledgement policy
{RTS – CTS – } QoS Null(dir, normal ack) (no data) – ACK	2	QoS Null frame to indicate no data or queue size
{RTS – CTS – } QoS Null(dir, normal ack) (no data) – <CF-Ack-Piggybacked-QoS-Data-Sequence>	2 or more	QoS Null with normal acknowledgement policy responded with a new TXOP
{RTS – CTS – } QoS Null(dir, normal ack) (no data) – <CF-Ack-Piggybacked-QoS-Poll-Sequence>	2 or more	QoS Null with normal acknowledgement policy responded with a new TXOP
{RTS – CTS – } QoS Null(dir, normal ack) – Data+CF-Ack	2	QoS Null with normal acknowledgement policy responded with a Data+CF-Ack
{RTS – CTS – } BAR – BA	2	Immediate Block Ack Request and its response frame.
{RTS – CTS – } BAR – ACK	2	Delayed Block Ack Request and a response frame to it.
{RTS – CTS – } BA – ACK	2	Delayed Block Ack response

Table 22.4 – <CF-Ack Piggybacked QoS Data sequence>

CF-Ack Piggybacked QoS Data Sequence	Frames in Sequence	Usage
QoS Data(dir, normal ack)+CF-Ack – ACK [– TXOP-Sequence] {QoS Null(dir, normal ack)(no data) – ACK}	2 or more	Start of a TXOP sequence with a QoS Data frame piggybacked onto the acknowledgement to the data frame sent in the previous sequence. (*1), (*2), (*3)
QoS Data(dir,normal ack)+CF-Ack – <CF-Ack-QoS-Piggybacked Poll Sequence>	3 or more	Start of a TXOP sequence with a QoS Data frame piggybacked onto the acknowledgement to the data frame sent in the previous sequence. The response frame would start a new piggybacked Poll Sequence. (*1), (*2), (*3) (*5)
QoS Data(dir,normal ack)+CF-Ack – <CF-Ack-Piggybacked-QoS-Data-Sequence>	3 or more	Start of a TXOP sequence with a QoS Data frame piggybacked onto the acknowledgement to the data frame sent in the previous sequence. The response frame would start a piggybacked non-poll data frame Sequence. (*1), (*2), (*3) (*5)
QoS Data(dir,normal ack)+CF-Ack – Data+CF-Ack	2	Start of a TXOP sequence with a QoS Data frame with ack policy set to normal ack and piggybacked onto the acknowledgement to the data frame sent in the previous sequence. The response frame is a Data+Ack, addressed to a non-QoS STA. (*1), (*2), (*3) (*5)
QoS Data(dir,block ack no ack)+CF-Ack [– <TXOP-Sequence>] { – QoS Null(dir, normal ack)(no data) – ACK}	1 or more	Start of a TXOP sequence with a QoS Data frame with ack policy set to block ack or no ack and piggybacked onto the acknowledgement to the data frame sent in the previous sequence. (*1), (*2), (*3)

Change the title of legend as follows:

1 LEGEND (For Table 21, ~~and~~ Table 22, Table 22.1, Table 22.2, Table 22.3 and Table 22.4)

2 *Insert the following definitions at the end of the legend*

3 24–A “|” between two items implies that one or the other item, but not both, is used.

4 25–“QoS Data(dir,normal ack)” represents any data frame of subtype QoS Data, that has no piggybacking of
5 acknowledgement or polling, with an individual address in the Address1 field and the acknowledgement
6 policy set to normal ack.

7 26–“QoS Data(dir,no ack)” represents any data frame of subtype QoS Data, that has no piggybacking of
8 acknowledgement or polling, with an individual address in the Address1 field and the acknowledgement
9 policy set to no ack.

10 27–“QoS Data(dir,block ack)” represents any data frame of subtype QoS Data, that has no piggybacking of
11 acknowledgement or polling, with an individual address in the Address1 field and the acknowledgement
12 policy set to block ack.

13 28–“BAR” represents any Block Ack Request frame.

14 29–“BA” represents any Block Ack frame.

15 30–“QoS CF-Poll(no data)” represents a data frame of subtype QoS CF-Poll(no data).

16 31–(*1) implies that the frame is transmitted within a SIFS of the previously transmitted MPDU.

17 32–(*2) implies that piggybacking of acknowledgement to data frame is allowed only in the first response
18 frame.

19 32–(*3) implies that the final QoS Null sequence shall be inserted if the TXOP holder has insufficient time
20 for queued MPDU, or too little time remaining before a dwell or medium occupancy boundary to send a
21 queued frame, and if the remaining TXOP is big enough to send the QoS Null sequence, or the TXOP holder
22 has no queued MPDU.

23 33–(*4) implies CTS to self is allowed if the non-AP QSTA intends to set the NAV at its vicinity. The frame
24 that follows CTS is sent by the QSTA that transmits the CTS.

25 34–(*5) Piggybacked poll sequence or a new sequence is started only when the previously received frame has
26 the duration field set to cover only the response frame.

27 35–An isolated equals sign “=” represents a PIFS separating the pair of frames.

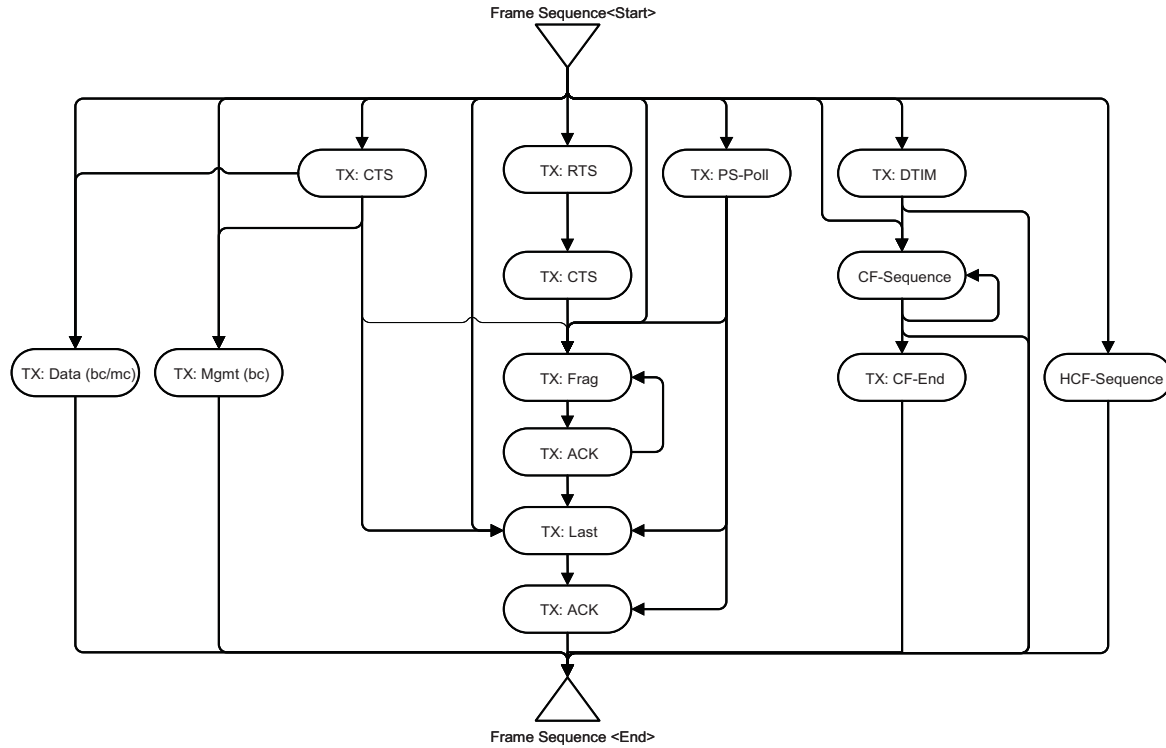


Figure 62.11 - Frame Sequence

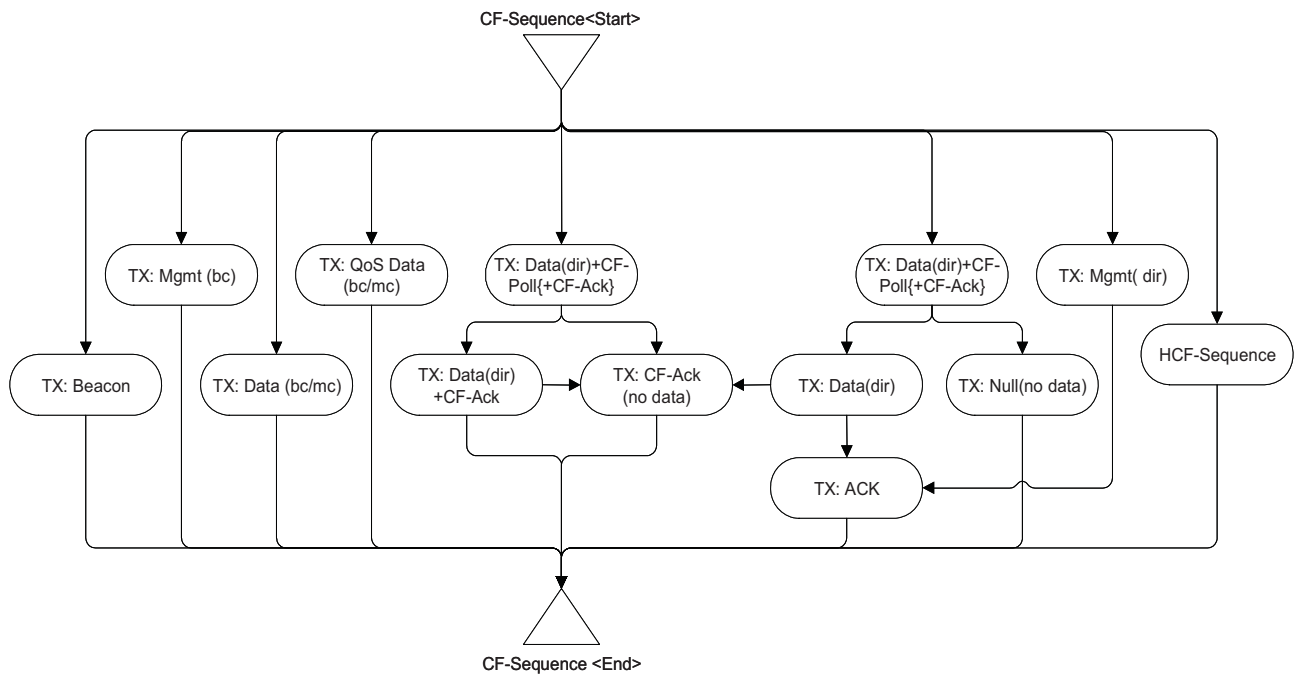


Figure 62.12 – CF Frame Sequence

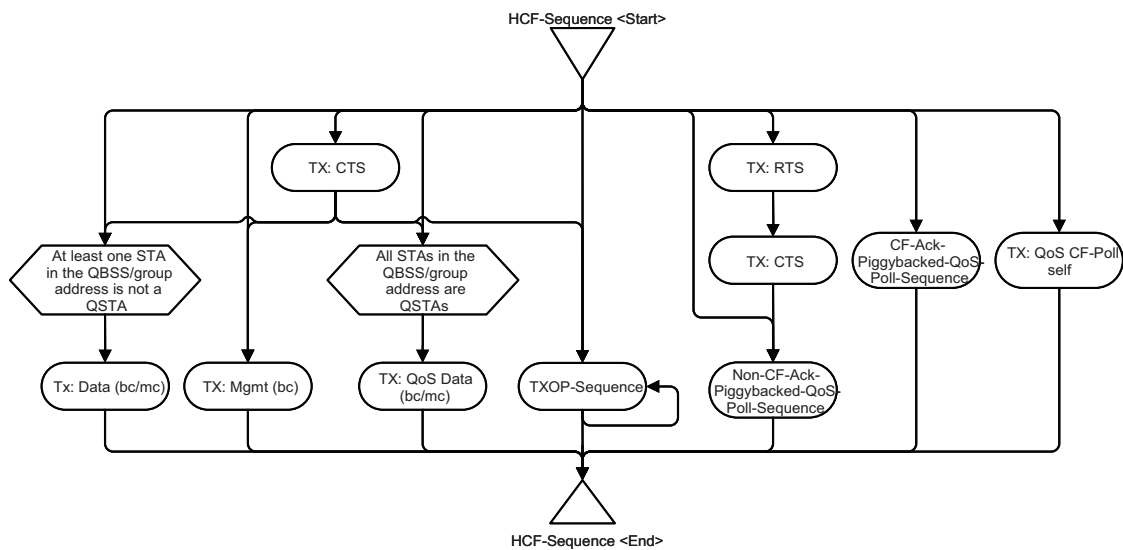
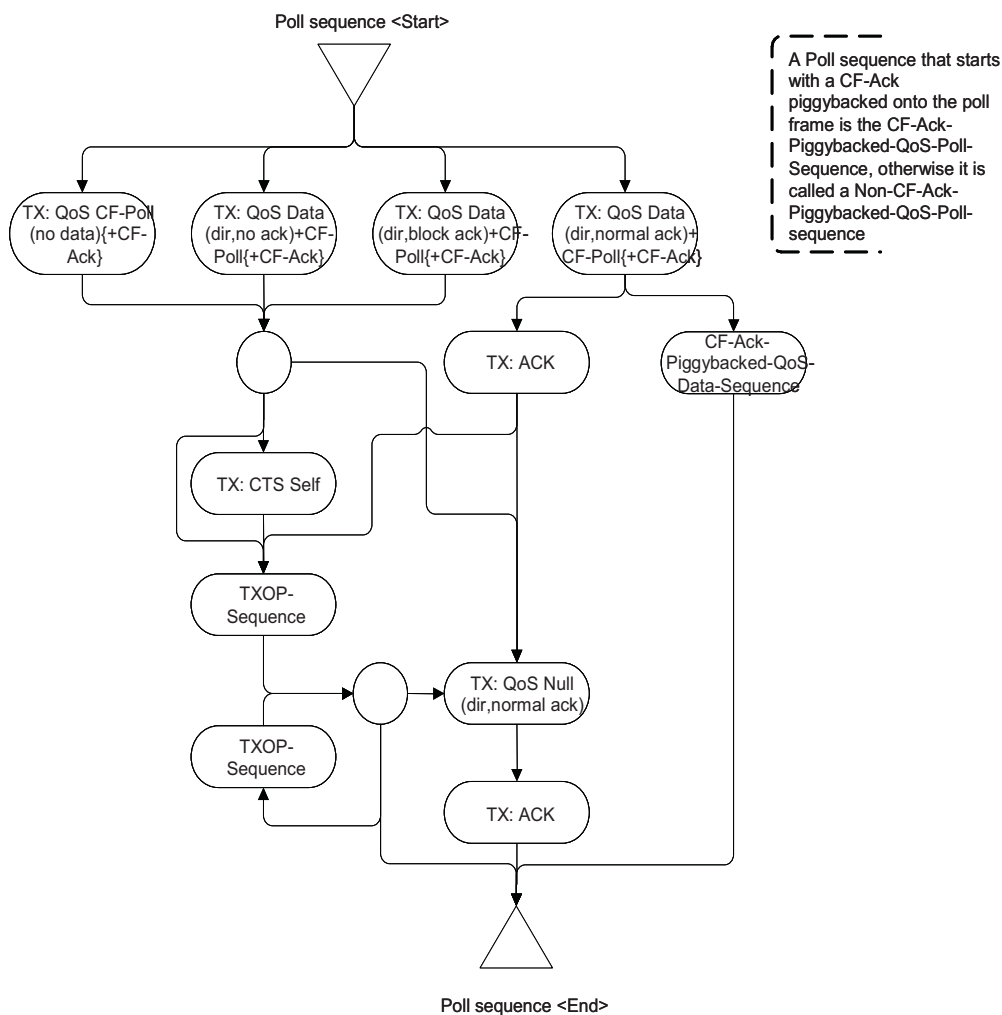
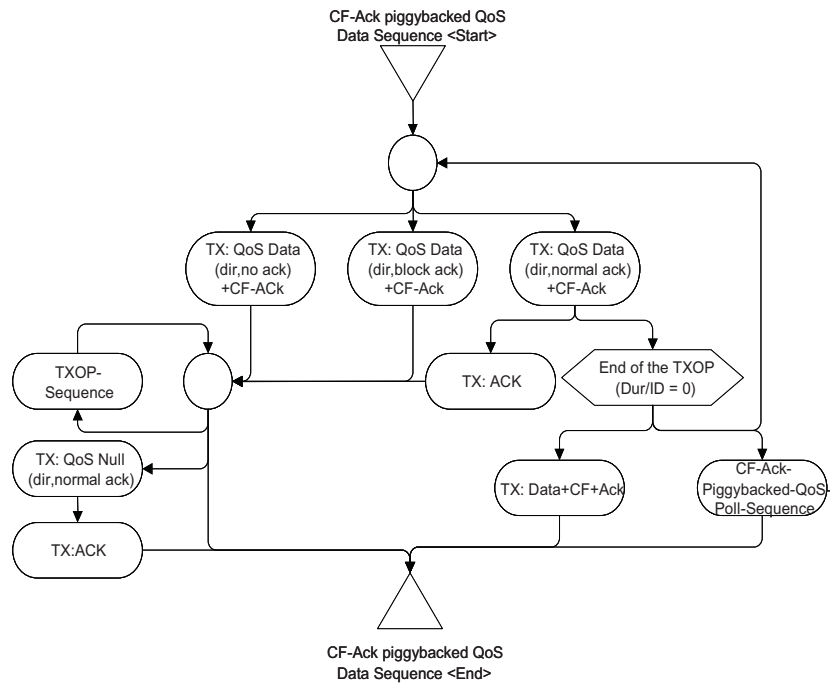


Figure 62.13 – HCF Sequence



1

Figure 62.14 – Poll Sequence

2

3

Figure 62.15 – CF-Ack Piggybacked QoS Data Sequence

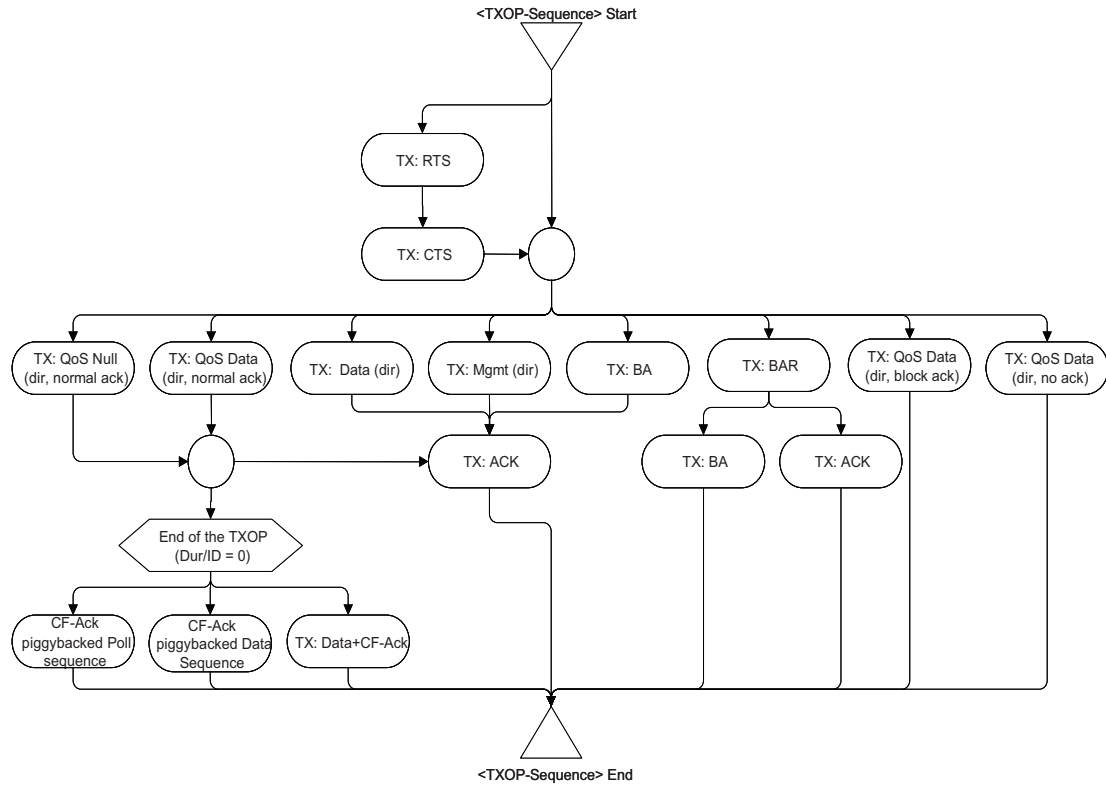


Figure 62.16 – TXOP Sequence

10. Layer management

10.3 MLME SAP Interface

10.3.2 Scan

10.3.2.2 MLME-SCAN.confirm

10.3.2.2.2 Semantics of the service primitive

Insert the following entries into the table consisting BSSDescription in the subclause as shown:

Name	Type	Valid Range	Description
QBSSLoad	As defined in Frame Format	As defined in Frame Format	The values from the QBSS Load information element if such an element was present in the probe response or beacon, else null.
EDCAParameterSet	As defined in Frame Format	As defined in Frame Format	The values from the EDCA Parameter Set information element if such an element was present in the probe response or beacon, else null.
QoSCapability	As defined in frame format	As defined in frame format	The values from the QoS Capability information element if such an element was present in the probe response or beacon, else null.

10.3.6 Associate

10.3.6.1 MLME-ASSOCIATE.request

10.3.6.1.2 Semantics of the service primitive

Insert “APSD” and “QoSCapability” as parameters to the primitive MLME-ASSOCIATE.request behind “Listen Interval”.

Insert the following entries to the table in the subclause:

Name	Type	Valid Range	Description
<u>QoSAction</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the QoS action frame that is to be transported. The only allowed action is ADDTS Request. The parameter shall be present only if the MIB attribute dot11QoSOptionImplemented is true.</u>
<u>QoSCapability</u>	<u>As defined in frame format</u>	<u>As defined in frame format</u>	<u>Specifies the parameters within the QoS Capability information element that are supported by the MAC entity. The</u>

			parameter shall be present only if the MIB attribute <u>dot11QoSOptionImplemented</u> is <u>true</u> .
--	--	--	--

10.3.6.2 MLME-ASSOCIATE.confirm

10.3.6.2.2 Semantics of the service primitive

Insert “CapabilityInformation”, “SupportedRates” and “QoSCapability” as parameters to the primitive MLME-ASSOCIATE.confirm behind “ResultCode”.

Insert the following entries to the table in the subclause:

Name	Type	Valid Range	Description
<u>CapabilityInformation</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the operational capability definitions to be used by the peer MAC entity (AP).</u>
<u>SupportedRates</u>	<u>Set of Integers</u>	<u>1-127 inclusive (for each integer in the set)</u>	<u>The set of data rates that are supported by the peer MAC entity (AP).</u>
<u>EDCAPParameterSet</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the EDCA Parameter Set that the QSTA should use. The parameter shall be present only if the MIB attribute <u>dot11QoSOptionImplemented</u> is true.</u>
<u>QoSAction</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the QoS action frame that has been transported. The only possible action is ADDTS Response. The parameter shall be present only if the MIB attribute <u>dot11QoSOptionImplemented</u> is true.</u>

10.3.6.3 MLME-ASSOCIATE.indication

10.3.6.3.2 Semantics of the service primitive

Insert “CapabilityInformation”, “SupportedRates” and “QoSCapability” as parameters to the primitive MLME-ASSOCIATE.indication behind “PeerSTAAddress”.

Insert the following entries to the table in the subclause:

Name	Type	Valid Range	Description
<u>CapabilityInformation</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the operational capability definitions to be used by the peer MAC entity.</u>
<u>SupportedRates</u>	<u>Set of Integers</u>	<u>1-127 inclusive (for each integer in the set)</u>	<u>The set of data rates that are supported by the peer MAC entity.</u>
<u>QoSAction</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the QoS action frame that has been transported. The only possible action is ADDTS Request. The parameter shall be present only if the MIB attribute <u>dot11QoSOptionImplemented</u> is true.</u>
<u>QoSCapability</u>	<u>As defined in frame format</u>	<u>As defined in frame format</u>	<u>Specifies the parameters within the QoS Capability that are supported by the peer</u>

			MAC entity. The parameter shall be present only if the MIB attribute dot11QoSOptionImplemented is true.
--	--	--	---

Change 10.3.7.1 as shown below:

10.3.7 Reassociate

10.3.7.1 MLME-REASSOCIATE.request

10.3.7.1.2 Semantics of the service primitive

Insert “APSD” and “QoSCapability” as parameters to the primitive MLME-REASSOCIATE.request behind “Listen Interval”. Insert the following entries to the table in the subclause:

Name	Type	Valid Range	Description
<u>QoSAction</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the QoS action frame that is to be transported. The only allowed action is ADDTS Request. The parameter shall be present only if the MIB attribute dot11QoSOptionImplemented is true.</u>
<u>QoSCapability</u>	<u>As defined in frame format</u>	<u>As defined in frame format</u>	<u>Specifies the parameters within the QoS Capability information element that are supported by the MAC entity. The parameter shall be present only if the MIB attribute dot11QoSOptionImplemented is true.</u>

10.3.7.2 MLME-REASSOCIATE.confirm

10.3.7.2.2 Semantics of the service primitive

Insert “CapabilityInformation”, “SupportedRates” and “QoSCapability” as parameters to the primitive MLME-REASSOCIATE.confirm behind “ResultCode”.

Insert the following entries to the table in the subclause:

Name	Type	Valid Range	Description
<u>CapabilityInformation</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the operational capability definitions to be used by the peer MAC entity (AP).</u>
<u>SupportedRates</u>	<u>Set of Integers</u>	<u>1-127 inclusive (for each integer in the set)</u>	<u>The set of data rates that are supported by the peer MAC entity (AP).</u>
<u>EDCAParameterSet</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the EDCA Parameter Set that the QSTA should use. The only possible action is ADDTS Response. The parameter shall be present only if the MIB attribute dot11QoSOptionImplemented is true.</u>
<u>QoSAction</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the QoS action frame that has been transported. The parameter shall be present only if the MIB attribute</u>

			<u>dot11QoSOptionImplemented is true.</u>
--	--	--	---

10.3.7.3 MLME-REASSOCIATE.indication

10.3.7.3.2 Semantics of the service primitive

Insert “CapabilityInformation”, “SupportedRates” and “QoSCapability” as parameters to the primitive MLME-ASSOCIATE.indication behind “PeerSTAAddress”.

Insert the following entries to the table in the subclause:

Name	Type	Valid Range	Description
<u>CapabilityInformation</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the operational capability definitions to be used by the peer MAC entity.</u>
<u>SupportedRates</u>	<u>Set of Integers</u>	<u>1-127 inclusive (for each integer in the set)</u>	<u>The set of data rates that are supported by the peer MAC entity.</u>
<u>QoSAction</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>Specifies the QoS action frame that has been transported. The only possible action is ADDTS Request. The parameter shall be present only if the MIB attribute dot11QoSOptionImplemented is true.</u>
<u>QoSCapability</u>	<u>As defined in frame format</u>	<u>As defined in frame format</u>	<u>Specifies the parameters within the QoS Capability that are supported by the peer MAC entity. The parameter shall be present only if the MIB attribute dot11QoSOptionImplemented is true.</u>

Replace 10.3.10.1 with the updated subclause below:

10.3.10 Start

10.3.10.1 MLME-START.request

10.3.10.1.2 Semantics of the service primitive

Insert “EDCAParameterSet” as a parameter to the primitive MLME-START.request behind “OperationalRateSet”.

Insert the following entries to the table in the subclause:

Name	Type	Valid Range	Description
<u>EDCAParameterSet (QoS only)</u>	<u>As defined in Frame Format</u>	<u>As defined in Frame Format</u>	<u>The initial EDCA parameter Set values to be used in the QBSS. The parameter shall be present only if the MIB attribute dot11QoSOptionImplemented is true.</u>

Insert after 10.3.10.2.4 the following subclauses:

10.3.11 TS management interface

This mechanism supports the process of adding, modifying, or deleting a traffic stream in a QBSS using the procedures defined in 11.4.

The primitives used for this mechanism are called TS Management primitives, which include MLME-ADDTS.xxx and MLME-DELTS.xxx primitives, where xxx denotes request, confirm, indication, or response. Each primitive contains parameters that correspond to a QoS Action. Requests and responses may cause the transmission of either the corresponding QoS Action frames or the (Re)Association Request or Response frames. Confirms and indications are issued upon the receipt of the appropriate QoS action.

Table 22.1 defines which primitives are supported by which type of STA.

Table 22.1 – Supported TS Management Primitives

Primitive	Request	Confirm	Indication	Response
ADDTS	non-AP QSTA	non-AP QSTA	HC	HC
DELTS	non-AP QSTA & HC	non-AP QSTA & HC	non-AP QSTA & HC	-

10.3.11.1 MLME-ADDTS.request

10.3.11.1.1 Function

This primitive requests addition (or modification) of a traffic stream. It is valid at the non-AP QSTA, and requests the HC to admit the new or changed TS.

10.3.11.1.2 Semantics of the service primitive

The primitive parameters are as follows:

MLME-ADDTS.request (

- DialogToken,
- TSPEC,
- TCLAS.
- TCLASProcessing,
- ADDTSFailureTimeout

)

Name	Type	Valid Range	Description
DialogToken	Integer	0-127	Specifies a number unique to the QoS action primitives and frames used in adding (or modifying) the traffic stream of concern
TSPEC	As defined in frame format with the exception of the Surplus Bandwidth Allowance, Minimum PHY rate, and Maximum	As defined in frame format with the exception of the Surplus Bandwidth Allowance, Minimum PHY rate, and Maximum	Specifies the source address, destination address, TSID, traffic characteristics and QoS requirements of the traffic stream of concern

	and Minimum Service Intervals, which are optionally specified	and Minimum Service Intervals, which are optionally specified	
TCLAS (0 or more)	As defined in frame format	As defined in frame format	Specifies the rules and parameters by which an MSDU may be classified to the specified TS
TCLASProcessing	As defined in frame format	As defined in frame format	Specifies how the TCLAS elements are to be processed when there are multiple TCLAS elements
ADDTSTimeout	Integer	Greater than or equal to 1	Specifies a time limit (in TU) after which the TS set up procedure will be terminated

10.3.11.1.3 When generated

This primitive is generated by the SME at a non-AP QSTA to request the addition of a new (or modification of an existing) traffic stream in order to support parameterized QoS transport of the MSDUs belonging to this traffic stream when a higher-layer protocol or mechanism signals the QSTA to initiate such an addition (or modification).

10.3.11.1.4 Effect of receipt

The non-AP QSTA operates according to the procedures defined in 11.4.

10.3.11.2 MLME-ADDTTS.confirm

10.3.11.2.1 Function

This primitive reports the results of a traffic stream addition (or modification) attempt.

10.3.11.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDTTS.confirm (
    ResultCode,
    DialogToken,
    TSDelay,
    TSPEC,
    Schedule Element,
    TCLAS,
    TCLASProcessing
)
```

Name	Type	Valid Range	Description
ResultCode	Enumeration	SUCCESS, INVALID-PARAMETERS, REJECTED_WITH_SUGGESTED_CHANGES,	Indicates the results of the corresponding MLME-ADDTTS.request

		REJECTED_FOR_DELAY_PERIOD, TIMEOUT, TRANSMISSION FAILURE	
DialogToken	Integer	As defined in the corresponding MLME-ADDTS.request	Specifies a number unique to the QoS action primitives and frames used in adding (or modifying) the TS
TSDelay	Integer	≥ 0	When ResultCode is "REJECTED_FOR_DELAY_PERIOD", provides the amount of time a QSTA should wait before attempting another ADDTS request.
TSPEC	As defined in frame format	As defined in frame format	Specifies the source address, destination address, TSInfo, traffic characteristics and QoS requirements of the TS
Schedule Element	As defined in frame format	As defined in frame format	Specifies the Minimum Service Interval, the Maximum Service Interval, the Minimum TXOP Duration, the Maximum TXOP Duration, and the Specification Interval
TCLAS (0 or more)	As defined in frame format	As defined in frame format	Specifies the rules and parameters by which an MSDU may be classified to the specified TS.
TCLASProcessing	As defined in frame format	As defined in frame format	Specifies how the TCLAS elements are to be processed when there are multiple TCLAS elements

For the ResultCode value of SUCCESS, the TSPEC and optional TCLAS parameters describe the characteristics of the traffic stream that has been created (or modified). In the case of SUCCESS, these parameters exactly match those of the matching MLME-ADDTS.request.

For other values of ResultCode, no new TS has been created. In the case of REJECTED_WITH_SUGGESTED_CHANGES, the TSPEC represents an alternative proposal by the HC. A TS is not created with this definition. If the suggested changes are acceptable to the non-AP QSTA, it is the responsibility of the non-AP QSTA to set up the TS with the suggested changes.

If this is the result of a modification of an existing TS, the status of that TS remains unchanged.

10.3.11.2.3 When generated

This primitive is generated by the MLME as a result of an MLME-ADDTS.request indicating the results of that request.

This primitive is generated when that MLME-ADDTS.request is found to contain invalid parameters, when a timeout occurs, or when the non-AP QSTA receives a response in the form of an ADDTS response in the corresponding QoS Action frame or in the (Re)Association Response from the HC.

10.3.11.2.4 Effect of receipt

The SME is notified of the results of the traffic stream addition (or modification) procedure.

The SME should operate according to the procedures defined in 11.4.

10.3.11.3 MLME-ADDTS.indication

10.3.11.3.1 Function

This primitive reports to the HC SME the request for adding (or modifying) a traffic stream.

10.3.11.3.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDTS.indication (
    DialogToken,
    Non-APQSTAAddress
    TSPEC,
    TCLAS,
    TCLASProcessing
)
```

Name	Type	Valid Range	Description
DialogToken	Integer	As defined in the received ADDTS request frame	Specifies a number unique to the QoS action primitives and frames used in adding (or modifying) the TS
Non-APQSTAAddress	MAC Address		Contains the MAC address of the non-AP QSTA that initiated the MLME-ADDTS.request
TSPEC	As defined in frame format with the exception of the Surplus Bandwidth Allowance, Minimum PHY rate, and Maximum and Minimum Service Intervals, which are optionally specified	As defined in the received ADDTS request frame with the exception of the Surplus Bandwidth Allowance, Minimum PHY rate, and Maximum and Minimum Service Intervals, which are optionally specified	Specifies the source address, destination address, TSID, traffic characteristics and QoS requirements of the TS
TCLAS (0 or more)	As defined in frame format	As defined frame format	Specifies the rules and parameters by which an MSDU may be classified to the specified TS
TCLASProcessing	As defined in frame format	As defined in frame format	Specifies how the TCLAS elements are to be processed when there are multiple TCLAS elements

The TCLAS is optional at the discretion of the non-AP QSTA that originated the request. An HC shall be capable of receiving an ADDTS request frame that contains a TCLAS element and generating an indication that contains this as a parameter.

10.3.11.3.3 When generated

This primitive is generated by the MLME as a result of receipt of a request to add (or modify) a traffic stream by a specified non-AP QSTA in the form of an Add TS request in the corresponding QoS Action frame or in the (re)association request frame.

10.3.11.3.4 Effect of receipt

The SME is notified of the request for a traffic stream addition (or modification) by a specified non-AP QSTA.

This primitive solicits an MLME-ADDTS.response from the SME that reflects the results of admission control at the HC on the traffic stream requested to be added (or modified).

The SME should operate according to the procedures defined in 11.4.

The SME shall generate an MLME-ADDTS.response within a dot11ADDTSResponseTimeout.

10.3.11.4 MLME-ADDTS.response

10.3.11.4.1 Function

This primitive responds to the request for a traffic stream addition (or modification) by a specified non-AP QSTA MAC entity.

10.3.11.4.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDTS.response (
    ResultCode,
    DialogToken,
    TSDelay,
    Non-APQSTAAddress,
    TSPEC,
    Schedule Element
    TCLAS,
    TCLASProcessing
)
```

Name	Type	Valid Range	Description
ResultCode	Enumeration	SUCCESS, INVALID- PARAMETERS, REJECTED_WITH _SUGGESTED_C HANGES, REJECTED_FOR_ DELAY_PERIOD	Indicates the results of the corresponding MLME-ADDTS.indication
DialogToken	Integer	As defined in the corresponding MLME-	DialogToken of the matching MLME-ADDTS.indication

		ADDTS.indication	
Non-APQSTAAAddress	MAC Address		non-AP QSTA Address of the matching MLME-ADDTS.indication
TSDelay	Integer	≥ 0	When ResultCode is "REJECTED_FOR_DELAY_PERIOD", provides the amount of time a QSTA should wait before attempting another ADDTS request.
TSPEC	As defined in frame format	As defined in frame format	Specifies QoS parameters of the TS.
Schedule Element	As defined in frame format	As defined in frame format	Specifies the Minimum Service Interval, the Maximum Service Interval, the Minimum TXOP Duration, the Maximum TXOP Duration, and the Specification Interval
TCLAS (0 or more)	As defined in frame format	As defined in frame format	Specifies the rules and parameters by which an MSDU may be classified to the specified TS
TCLASProcessing	As defined in frame format	As defined in frame format	Specifies how the TCLAS elements are to be processed when there are multiple TCLAS elements

The DialogToken and non-AP QSTA Address parameters shall contain the value from the matching MLME-ADDTS.indication.

If the ResultCode is SUCCESS, the TSPEC and (optional) TCLAS parameters shall contain the values from the matching MLME-ADDTS-indication.

If the ResultCode is REJECTED_WITH_SUGGESTED_CHANGES, the TSPEC and TCLAS parameters represent an alternative proposed TS. The TS, however, is not created. The TSID and Direction within the TSPEC shall be as in the matching MLME-ADDTS.indication. The difference may lie in the QoS (e.g., minimum data rate, mean data rate and delay bound) values, as a result of admission control performed at the SME of the HC on the traffic stream requested to be added (or modified) by the non-AP QSTA. If sufficient bandwidth is not available, the QoS values may be reduced—in one extreme, the minimum data rate, mean data rate, delay bound, and jitter bound may be all set to zero, indicating that no QoS is to be provided to this traffic stream.

10.3.11.4.3 When generated

This primitive is generated by the MLME at the HC as a result of an MLME-ADDTS.indication to initiate addition (or modification) of a traffic stream with a specified peer MAC entity or entities.

10.3.11.4.4 Effect of receipt

This primitive approves addition (or modification) of a traffic stream requested by a specified non-AP QSTA MAC entity, with or without altering the traffic specification.

This primitive shall cause the MAC entity at the HC to send an ADDTS Response in the corresponding QoS Action Management frame or (re)association response frame to the requesting non-AP QSTA containing the specified parameters.

10.3.11.5 MLME-DELTS.request

10.3.11.5.1 Function

This primitive requests deletion of a traffic stream with a specified peer MAC.

This primitive may be generated at either a non-AP QSTA or the HC.

10.3.11.5.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DELTS.request (
    Non-APQSTAAddress,
    TSInfo
)
```

Name	Type	Valid Range	Description
Non-APQSTAAddress (HC only)	MAC Address		(At the HC only) Specifies the MAC address of the non-AP QSTA that initiated this TS.
TSInfo	As defined in frame format	As defined in frame format	The TSInfo specifies the TS to be deleted.

10.3.11.5.3 When generated

This primitive is generated by the SME at a QSTA to initiate deletion of a traffic stream TAs when a higher-layer protocol or mechanism signals the QSTA to initiate such a deletion.

10.3.11.5.4 Effect of receipt

This primitive initiates a traffic stream deletion procedure. The MLME subsequently issues a MLME-DELTS.confirm that reflects the results.

This primitive shall cause the local MAC entity to send out a DELTS frame containing the specified parameters. If this primitive was generated at the HC, the frame is sent to the specified non-AP QSTA MAC address. If this primitive was generated at the non-AP QSTA, the frame is sent to its HC. In either case, the DELTS frame does not solicit a response from the recipient frame other than an acknowledgment to receipt of the frame.

10.3.11.6 MLME-DELTS.confirm

10.3.11.6.1 Function

This primitive reports the transmission status of a traffic stream deletion attempt with a specified peer MAC entity.

10.3.11.6.2 Semantics of the service primitive

The primitive parameters are as follows:

```

1      MLME-DELTS.confirm      (
2                                ResultCode,
3                                Non-APQSTAAddress,
4                                TSInfo
5                                )

```

Name	Type	Valid Range	Description
ResultCode	Enumeration	SUCCESS, INVALID- PARAMETERS, FAILURE	Indicates the results of the corresponding MLME-DELTS.request
Non-APQSTAAddress (HC only)	MAC address		(HC only) contains the non-AP QSTA MAC address of the matching request.
TSInfo	As defined in frame format	As defined in the corresponding MLME-DELTS.request	Contains the TSInfo of the matching request

10.3.11.6.3 When generated

This primitive is generated by the MLME as a result of an MLME-DELTS.request after the DELTS frame has been sent (or attempts to send it have failed) and any internal state regarding the use of the TS has been destroyed.

10.3.11.6.4 Effect of receipt

The SME is notified of the results of the traffic stream deletion procedure.

10.3.11.7 MLME-DELTS.indication

10.3.11.7.1 Function

This primitive reports the deletion of a traffic stream by a specified peer MAC entity or deletion of the traffic stream due to an inactivity timeout (HC only).

10.3.11.7.2 Semantics of the service primitive

The primitive parameters are as follows:

```

18      MLME-DELTS.indication(
19                                Reason,
20                                Non-APQSTAAddress,
21                                TSInfo
22                                )
23

```

Name	Type	Valid Range	Description
Reason	Enumeration	REQUESTED, TIMEOUT	Indicates the reason why the TS is being deleted.
Non-APQSTAAddress (HC only)	MAC address		(HC only) The MAC address of the non-AP QSTA for which the TS is being deleted.
TSInfo	As defined in frame format	As defined in the received DELTS	Specifies the TSInfo of the traffic stream of concern.

		frame	
--	--	-------	--

10.3.11.7.3 When generated

This primitive is generated by the MLME as a result of receipt of an initiation to delete a traffic stream by a specified peer MAC entity.

This primitive may also be generated by the MLME at the HC as a result of inactivity of a particular traffic stream. Inactivity results when a period equal to the Inactivity Interval in the Traffic Specification for the traffic stream elapses without arrival of an MSDU belonging to that traffic stream at the MAC entity of the HC via an MA-UNITDATA.request primitive in the case where the HC is the source station of that traffic stream, or without reception of an MSDU belonging to that traffic stream by the MAC entity of the HC in the case where a non-AP QSTA is the source station of that traffic stream.

This primitive is generated after any other state concerning the TSID/Direction within the MAC has been destroyed.

10.3.11.7.4 Effect of receipt

The SME is notified of the initiation of a traffic stream deletion by a specified peer MAC entity.

10.3.12 Management of Direct Links

This clause describes the management procedures associated with Direct Links.

10.3.12.1 MLME-DLP.request

10.3.12.1.1 Function

This primitive requests the set up of a Direct Link with a specified peer MAC entity.

10.3.12.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DLP.request    (
    PeerMACAddress,
    DLPTimeoutValue,
    DLPResponseTimeout
)
```

Name	Type	Valid Range	Description
PeerMACAddress	MACAddress	Any valid individual MAC address	Specifies the MAC address of the non-AP QSTA that is the intended immediate recipient of the data flow.
DLPTimeoutValue	Integer	≥ 0	Specifies a time limit (in seconds) after which the direct link is terminated, if there are no frame exchange sequences with the peer. A value of 0 implies that the direct link is never to be terminated based on a timeout.
DLPResponseTimeout	Integer	≥ 1	Specifies a time limit (in TU) after which the Direct Link set up procedure is terminated.

10.3.12.1.3 When generated

This primitive is generated by the SME at a non-AP QSTA to set up a Direct Link with another non-AP QSTA.

10.3.12.1.4 Effect of receipt

This primitive initiates a Direct Link set up procedure. The MLME subsequently issues an MLME-DLP.confirm that reflects the results.

10.3.12.2 MLME-DLP.confirm

10.3.12.2.1 Function

This primitive reports the results of a Direct Link set up attempt with a specified peer MAC entity.

10.3.12.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DLP.confirm (
    PeerMACAddress,
    ResultCode,
    CapabilityInformation,
    DLPTIMEOUTValue,
    SupportedRates
)
```

Name	Type	Valid Range	Description
PeerMACAddress	MACAddress	Any valid individual MAC address	Specifies the MAC address of the non-AP QSTA that is the intended immediate recipient of the data flow.
ResultCode	Enumeration	SUCCESS, INVALID_PARAMETERS, NOT_ALLOWED, NOT_PRESENT, NOT_QSTA, REFUSED, TIMEOUT	Indicates the results of the corresponding MLME-DLP.request.
CapabilityInformation	As defined in Frame Format	As defined in Frame Format	Specifies the operational capability definitions to be used by the peer MAC entity.
DLPTIMEOUTValue	Integer	≥ 1	Specifies a time limit (in seconds) after which the direct link is terminated, if there are no frame exchange sequences with the peer. A value of 0 implies that the direct link is never to be terminated based on a timeout.
SupportedRates	Set of Integers	1-127 inclusive (for each integer in the set)	The set of data rates that are supported by the peer MAC

			entity.
--	--	--	---------

10.3.12.2.3 When generated

This primitive is generated by the MLME as a result of an MLME-DLP.request to establish a Direct Link with a specified peer MAC entity.

10.3.12.2.4 Effect of receipt

The SME is notified of the results of the Direct Link set up procedure.

10.3.12.3 MLME-DLP.indication

10.3.12.3.1 Function

This primitive reports the establishment of a Direct Link with a specified peer MAC entity.

10.3.12.3.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DLP.indication (
    PeerMACAddress,
    CapabilityInformation,
    DLPTIMEOUTValue,
    SupportedRates
)
```

Name	Type	Valid Range	Description
PeerMACAddress	MACAddress	Any valid individual MAC address	Specifies the MAC address of the non-AP QSTA that is the sender of the data flow.
CapabilityInformation	As defined in Frame Format	As defined in Frame Format	Specifies the operational capability definitions to be used by the peer MAC entity.
DLPTIMEOUTValue	Integer	≥ 1	Specifies a time limit (in seconds) after which the direct link is terminated, if there are no frame exchange sequences with the peer.
SupportedRates	Set of Integers	1-127 inclusive (for each integer in the set)	The set of data rates that are supported by the peer MAC entity.

10.3.12.3.3 When generated

This primitive is generated by the MLME as result of the establishment of a Direct Link with a specific peer MAC entity that resulted from a Direct Link set up procedure that was initiated by that specific peer MAC entity.

10.3.12.3.4 Effect of receipt

The SME is notified of the establishment of the Direct Link set up.

10.3.12.4 MLME-DLPTeardown.request

10.3.12.4.1 Function

This primitive requests the teardown of the Direct Link with a specified peer MAC entity.

10.3.12.4.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DLPTeardown.request  (
                             PeerMACAddress
                             )
```

Name	Type	Valid Range	Description
PeerMACAddress	MACAddress	Any valid individual MAC address	Specifies the MAC address of the non-AP QSTA that is the intended immediate recipient of the data flow.

10.3.12.4.3 When generated

This primitive is generated by the SME at a non-AP QSTA for tearing down a Direct Link with another non-AP QSTA.

10.3.12.4.4 Effect of receipt

This primitive initiates a Direct Link teardown procedure. The MLME subsequently issues an MLME-DLPTeardown.confirm that reflects the results.

10.3.12.5 MLME-DLPTeardown.confirm

10.3.12.5.1 Function

This primitive reports the results of a Direct Link teardown attempt with a specified peer MAC entity.

10.3.12.5.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DLPTeardown.confirm  (
                             PeerMACAddress,
                             ResultCode,
                             )
```

Name	Type	Valid Range	Description
PeerMACAddress	MACAddress	Any valid individual MAC address	Specifies the MAC address of the non-AP QSTA that is the intended immediate recipient of the data flow.
ResultCode	Enumeration	SUCCESS, FAILURE	Indicates the results of the corresponding MLME-DLPTeardown.request

10.3.12.5.3 When generated

This primitive is generated by the MLME as a result of an MLME-DLPTeardown.request to teardown an established Direct Link with a specified peer MAC entity.

10.3.12.5.4 Effect of receipt

The SME is notified of the results of the Direct Link teardown procedure.

10.3.12.6 MLME-DLPTeardown.indication**10.3.12.6.1 Function**

This primitive indicates the teardown of an already established Direct Link with a specific peer MAC entity.

10.3.12.6.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DLPTeardown.indication (
    PeerMACAddress,
    Reason
)
```

Name	Type	Valid Range	Description
PeerMACAddress	MACAddress	Any valid individual MAC address	Specifies the MAC address of the non-AP QSTA that is the sender of the data flow.
Reason	Enumeration	REQUESTED, TIMEOUT	Indicates the reason why the direct link is being torn down.

10.3.12.6.3 When generated

This primitive is generated by the MLME as result of the teardown of a Direct Link with a specific peer MAC entity that resulted from a Direct Link teardown procedure that was initiated either by that specific peer MAC entity or by the local MAC entity.

10.3.12.6.4 Effect of receipt

The SME is notified of the teardown of the Direct Link set up.

10.3.13 Higher Layer Synchronization Support

This mechanism supports the process of synchronization among higher-layer protocol entities residing within different wireless STAs. The actual synchronization mechanism in the higher layer is out of the scope of this standard. In principle, the MLME indicates the transmission/reception of frames with a specific multicast address in the Address 1 field of an MPDU of type data.

10.3.13.1 MLME-HL-SYNC.request**10.3.13.1.1 Function**

This primitive requests activating the synchronization support mechanism.

10.3.13.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-HL-SYNC.request (
    GroupAddress
)
```

Name	Type	Valid Range	Description
GroupAddress	MACAddress	A multicast MAC address	Specifies the multicast MAC address to which the synchronization frames are addressed. A synchronization frame is a data type frame with a higher layer synchronization information.

10.3.13.1.3 When generated

This primitive is generated by the SME when a higher layer protocol initiates a synchronization process.

10.3.13.1.4 Effect of Receipt

This request activates the synchronization support mechanism at the STA. The MLME subsequently issues an MLME-HL-SYNC.confirm that reflects the results of the higher layer synchronization support request. If the request has been successful, and the higher layer synchronization support mechanism has been activated, the MLME shall issue an MLME-HL-SYNC.indication whenever a higher layer synchronization frame, which is a data type frame with the specified multicast address in Address 1 field, is received or transmitted.

10.3.13.2 MLME-HL-SYNC.confirm

10.3.13.2.1 Function

This primitive confirms the activation of the higher layer synchronization support mechanism.

10.3.13.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-HL-SYNC.confirm (
    ResultCode,
)
```

Name	Type	Valid Range	Description
ResultCode	Enumeration	SUCCESS, NOT_SUPPORTED	Indicates the result of the MLME-HL-SYNC.request.

10.3.13.2.3 When generated

This primitive is generated by the MLME as a result of an MLME-HL-SYNC.request to activate the higher layer synchronization support mechanism for a particular multicast address.

10.3.13.2.4 Effect of Receipt

The SME is notified of the activation of the higher layer synchronization support mechanism. The result code of NOT_SUPPORTED is issued if the MAC/MLME does not support the higher layer synchronization support mechanism or if the address provided by the MLME-HL-SYNC.request is not a multicast address.

10.3.13.3 MLME-HL-SYNC.indication

10.3.13.3.1 Function

This primitive indicates the last symbol on air of a higher layer synchronization frame, whether transmitted or received by the MAC.

10.3.13.3.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-HL-SYNC.indication (
    SourceAddress,
    SequenceNumber
)
```

Name	Type	Valid Range	Description
SourceAddress	MACAddress	Any valid individual MAC address	Specifies the Source Address of the STA that transmitted the higher layer synchronization frame.
SequenceNumber	Integer	As defined in the frame format	Specifies the sequence number of the higher layer synchronization frame received or transmitted.

10.3.13.3.3 When generated

This primitive is generated by the MLME when the successful reception or transmission of a higher layer synchronization frame is detected, as indicated by the PHY_RXEND.indication or PHY_TXEND.confirm primitives generated by the PHY layer. The higher layer synchronization frame is identified by the multicast MAC address registered by an earlier MLME-HL-SYNC.request primitive, in Address 1 field of a data type frame.

10.3.13.3.4 Effect of Receipt

The SME is notified of the reception or transmission of a higher layer synchronization frame.

10.3.14 Block Ack

This mechanism supports the initiation (or modification) and termination of Block Ack.

The primitives used for this mechanism are called Block Ack primitives, which include MLME-ADDBA.xxx and MLME-DELBA.xxx primitives, where xxx denotes request, confirm, indication, or response. Each primitive contains parameters that correspond to a Block Ack Action frame body. Requests and responses may cause these frames to be sent. Confirms and indications are emitted when an appropriate Block Ack action frame is received.

10.3.14.1 MLME-ADDBA.request

10.3.14.1.1 Function

This primitive requests the initiation (or modification) of Block Ack with a peer MAC entity.

10.3.14.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDBA.request (
    PeerQSTAAddress,
    DialogToken,
    TID,
    BlockAckPolicy
    BufferSize,
    BlockAckTimeout,
    ADDBAFailureTimeout
)
```

Name	Type	Valid Range	Description
PeerQSTAAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity with which to perform the Block Ack initiation (or modification).
DialogToken	Integer	0-255	Specifies a number unique to the Block Ack action primitives and frames used in defining the Block Ack.
TID	Integer	0-15	Specifies the TID of the data.
BlockAckPolicy	Enumeration	Immediate, Delayed	Specifies the Block Ack Policy
BufferSize	Integer	0-127	Specifies the number of MPDUs that can be held in its buffer
BlockAckTimeout	Integer	0-65535	Specifies the number of TUs without a frame exchange between peers after which the Block Ack is considered to be torn down.
ADDBAFailureTimeout	Integer	Greater than or equal to 1	Specifies a time limit (in TU) after which the BA set up procedure is terminated.

10.3.14.1.3 When generated

This primitive is generated by the SME at a QSTA to request initiation (or modification) of Block Ack with the specified peer MAC entity.

10.3.14.1.4 Effect of receipt

The QSTA shall send the ADDBA request frame to the specified peer MAC entity.

10.3.14.2 MLME-ADDBA.confirm

10.3.14.2.1 Function

The primitive reports the results of initiation (or modification) of the Block Ack attempt with the specified peer MAC entity

10.3.14.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDBA.confirm (
    PeerQSTAAAddress,
    DialogToken,
    TID,
    ResultCode,
    BlockAckPolicy,
    BufferSize,
    BlockAckTimeout
)
```

Name	Type	Valid Range	Description
PeerQSTAAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity with which the Block Ack initiation (or modification)

			was attempted. This value must match the PeerQSTAAAddress parameter specified in MLME-ADDBA.request.
DialogToken	Integer	0-255	Specifies a number unique to the Block Ack action primitives and frames used in defining the Block Ack. This value must match the Dialog Token parameter specified in MLME-ADDBA.request.
TID	Integer	0-15	Specifies the TID of the data. This value must match the TID specified in MLME-ADDBA.request
ResultCode	Enumeration	SUCCESS, REFUSED, TIMEOUT	Indicates the result of the corresponding MLME-ADDBA.request.
BlockAckPolicy	Enumeration	Immediate, Delayed	Specifies the Block Ack Policy
BufferSize	Integer	0-127	Specifies the maximum number of MPDUs in the block for the specified TID
BlockAckTimeout	Integer	0-65535	Specifies the number of TUs without a frame exchange between peers after which the Block Ack is considered to be torn down.

10.3.14.2.3 When generated

This primitive is generated by the MLME as a result of an MLME-ADDBA.request to indicate the results of that request.

10.3.14.2.4 Effect of receipt

The SME is notified of the results of the Block Ack initiation (or modification).

10.3.14.3 MLME-ADDBA.indication

10.3.14.3.1 Function

This primitive reports the initiation (or modification) of Block Ack by a peer MAC entity.

10.3.14.3.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDBA.indication (
    PeerQSTAAAddress,
    DialogToken,
    TID,
    BlockAckPolicy
    BufferSize,
    BlockAckTimeout
)
```

Name	Type	Valid Range	Description
PeerQSTAAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity that requested the Block Ack initiation (or modification).
DialogToken	Integer	0-255	Specifies a number unique to the Block Ack action primitives and frames used in defining

			the Block Ack.
TID	Integer	0-15	Specifies the TID of the data.
BlockAckPolicy	Enumeration	Immediate, Delayed	Specifies the Block Ack Policy
BufferSize	Integer	0-127	Specifies the number of MPDUs that can be held in peerQSTAAddress buffer
BlockAckTimeout	Integer	0-65535	Specifies the number of TUs without a frame exchange between peers after which the Block Ack is considered to be torn down.

10.3.14.3.3 When generated

This primitive is generated by the MLME as a result of receipt of a Block Ack initiation (or modification) by the specified peer MAC entity in the form of an ADDBA request frame.

10.3.14.3.4 Effect of receipt

The SME is notified of the initiation (or modification) of the Block Ack by the specified peer MAC entity.

10.3.14.4 MLME-ADDDBA.response

10.3.14.4.1 Function

The primitive responds to the initiation (or modification) by a specified peer MAC entity.

10.3.14.4.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDDBA.response(
    PeerQSTAAddress,
    DialogToken,
    TID,
    ResultCode,
    BlockAckPolicy,
    BufferSize,
    BlockAckTimeout
)
```

Name	Type	Valid Range	Description
PeerQSTAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity that attempted the Block Ack initiation (or modification). This value must match the PeerQSTAAddress parameter specified in MLME-ADDDBA.indication.
DialogToken	Integer	0-255	Specifies a number unique to the Block Ack action primitives and frames used in defining the Block Ack. This value must match the Dialog Token parameter specified in MLME-ADDDBA.indication.
TID	Integer	0-15	Specifies the TID of the data. This value must match the TID specified in MLME-ADDDBA.indication.
ResultCode	Enumeration	SUCCESS, REFUSED, INVALID	Indicates the result of the corresponding MLME-ADDDBA.indication.

		PARAMETERS, TIMEOUT	
BlockAckPolicy	Enumeration	Immediate, Delayed	Specifies the Block Ack Policy. Undefined when ResultCode is REFUSED.
BufferSize	Integer	0-127	Specifies the number of MPDUs that can be in Block for the specified TID. Undefined when ResultCode is REFUSED.
BlockAckTimeout	Integer	0-65535	Specifies the number of TUs without a frame exchange between peers after which the Block Ack is considered to be torn down.

10.3.14.4.3 When generated

This primitive is generated by the MLME as a result of an MLME-ADDBA.indication to initiate Block Ack by the specified peer MAC entity.

10.3.14.4.4 Effect of receipt

The primitive causes the MAC entity to send an ADDBA response frame to the specified peer MAC entity.

10.3.14.5 MLME-DELBA.request

10.3.14.5.1 Function

This primitive requests the deletion of Block Ack with a peer MAC entity.

10.3.14.5.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DELBA.request (
    PeerQSTAAAddress,
    Direction,
    TID
)
```

Name	Type	Valid Range	Description
PeerQSTAAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity with which to perform the Block Ack deletion.
Direction	Enumeration	Originator, recipient	Specifies if the MAC entity initiating the MLME-DELBA.request is the originator or the recipient of the data stream that uses the Block Ack.
TID	Integer	0-15	Specifies the TID of the MSDUs for which this Block Ack has been set up.

10.3.14.5.3 When generated

This primitive is generated by the SME at a QSTA to request deletion of Block Ack with the specified peer MAC entity.

10.3.14.5.4 Effect of receipt

The QSTA shall send the DELBA frame to the specified peer MAC entity.

10.3.14.6 MLME-DELBA.confirm**10.3.14.6.1 Function**

The primitive reports the transmission status of the Block Ack deletion attempt with a specified peer MAC entity.

10.3.14.6.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DELBA.confirm (
    PeerQSTAAAddress,
    Direction,
    TID,
    ResultCode,
)
```

Name	Type	Valid Range	Description
PeerQSTAAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity with which the Block Ack deletion was attempted. This value must match the PeerQSTAAAddress parameter specified in MLME-DELBA.request.
Direction	Enumeration	Originator, recipient	Specifies if the MAC entity initiating the MLME-DELBA.request is the originator or the recipient of the data stream that uses the Block Ack.
TID	Integer	0-15	Specifies the TID of the MSDUs for which this Block Ack has been set up. This value must match the TID specified in MLME-DELBA.request
ResultCode	Enumeration	SUCCESS, INVALID-PARAMETERS, FAILURE	Indicates the result of the corresponding MLME-DELBA.request.

10.3.14.6.3 When generated

This primitive is generated by the MLME as a result of an MLME-DELBA.request to indicate the results of that request.

10.3.14.6.4 Effect of receipt

The SME is notified of the results of the Block Ack deletion.

10.3.14.7 MLME-DELBA.indication**10.3.14.7.1 Function**

This primitive reports the deletion of Block Ack with by a peer MAC entity.

10.3.14.7.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DELBA.indication (
    PeerQSTAAAddress,
```

Direction,
TID
)

Name	Type	Valid Range	Description
PeerQSTAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity with which to perform the Block Ack deletion.
Direction	Enumeration	Originator, Recipient	Specifies if the MAC entity initiating the MLME-DELBA.request is the originator or the recipient of the data stream that uses the Block Ack.
TID	Integer	0-15	Specifies the TID of the MSDUs for which this Block Ack has been set up.
Reason	Enumeration	REQUESTED, TIMEOUT	Indicates the reason why the BA is being deleted.

10.3.14.7.3 When generated

This primitive is generated by the MLME as a result of receipt of a Block Ack deletion by the specified peer MAC entity in the form of a DELBA frame.

10.3.14.7.4 Effect of receipt

The SME is notified of the deletion of the Block Ack by the specified peer MAC entity.

10.3.15 Schedule Element Management

This clause describes the management procedures associated with the QoS Schedule Element.

The primitives defined are MLME-SCHEDULE.request, MLME-SCHEDULE.confirm, and MLME-SCHEDULE.indication.

10.3.15.1 MLME-SCHEDULE.request

10.3.15.1.1 Function

This primitive requests transmission of a Schedule frame. It is valid at the HC.

10.3.15.1.2 Semantics of the service primitive

The primitive parameters are as follows:

MLME-SCHEDULE.request (
Non-AP QSTA Address,
Schedule Element
)

Name	Type	Valid Range	Description
Non-AP QSTA Address	MAC Address	Any valid individual address	MAC Address of the non-AP QSTA to which the Schedule frame shall be sent
Schedule Element	As defined in frame format	As defined in frame format	Specifies the schedule for the non-AP QSTA, including the Service Interval (min and max) TXOP duration (min and max) and Specification Interval

10.3.15.1.3 When generated

This primitive is generated by the SME at the HC to send the Schedule information, in the form of a Schedule frame, to a specified non-AP QSTA when the Schedule Information for the non-AP QSTA is changed.

10.3.15.1.4 Effect of receipt

This primitive shall cause the MAC entity at the HC to send a Schedule frame to the non-AP QSTA specified in the primitive containing the specified Schedule parameters.

10.3.15.2 MLME-SCHEDULE.confirm**10.3.15.2.1 Function**

This primitive reports the transmission status of a MLME-SCHEDULE.request.

10.3.15.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-SCHEDULE.confirm    (
                           ResultCode,
                           )
```

Name	Type	Valid Range	Description
ResultCode	Enumeration	SUCCESS, INVALID PARAMETERS, UNESPECIFIED FAILURE	Indicates the results of the corresponding MLME-SCHEDULE.request

10.3.15.2.3 When generated

This primitive is generated by the MLME as a result of an MLME-SCHEDULE.request when the action completes.

10.3.15.2.4 Effect of receipt

The SME is notified of the result of the MLME-SCHEDULE.request. If the result is SUCCESS, the Schedule Element has been correctly sent by the HC to the non-AP QSTA in the Schedule Frame.

10.3.15.3 MLME-SCHEDULE.indication**10.3.15.3.1 Function**

This primitive reports the reception of a new Schedule by the non-AP QSTA in the form of a Schedule frame. It is valid at the non-AP QSTA.

10.3.15.3.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-SCHEDULE.indication (
```

Schedule Element
)

Name	Type	Valid Range	Description
Schedule Element	As defined in frame format	As defined in frame format	Specifies the schedule for the non-AP QSTA, including the Service Interval (min and max) TXOP duration (min and max) and Specification Interval

10.3.15.3.3 When generated

This primitive is generated by the MLME as a result of receipt of a new Schedule in the form of a Schedule frame.

10.3.15.3.4 Effect of receipt

The SME is notified of the receipt of QoS Schedule in the form of a Schedule frame. The new Schedule Element parameters shall overwrite previously stored values.

11. MAC sublayer management

11.2 Power Management

11.2.1 Power management in an infrastructure network

Insert the following paragraph at the end of subclause, above 11.2.1.1:

A QSTA may be in power-save mode before the set up of DLP or Block Ack. Once DLP is set up with another QSTA, the QSTA suspends the power save mode and shall always be awake. When a station enters normal (non-APSD) PS mode, any downlink Block Ack agreement without an associated schedule is suspended for the duration of this PS mode. MSDUs for TID without a schedule are sent using normal acknowledgement following a PS-poll as described in rest of this clause. Uplink Block ack, Block Acks for any TSD with a schedule, and any Block Acks to APSD QSTA continue to operate normally.

Insert the following subclause after 11.2.3 renumbering all the following subclauses:

11.2.1.4 Power management with APSD

QAPs capable of supporting automatic power-save delivery (APSD) shall signal this capability through the use of the APSD subfield in the Capability Information Field in Association Request management frame.

Non-AP QSTAs operating in a QBSS wishing to utilize the automatic power-save delivery mechanism shall inform the AP by setting the APSD subfield of the TS info field in the TSPEC element to 1. The request may be sent in an ADDTS request frame or in (re) association frames. The request could also be sent for ACs for which ACM subfield is set to 0. The QAP shall not arbitrarily transmit MSDUs to non-AP QSTAs that use APSD, but shall buffer MSDUs and only transmit them during the service period.

Non-AP QSTAs use the power management field in the frame control field of a frame to indicate whether it is in active or power-save mode. As APSD is a mechanism for the delivery of downlink frames to power-saving stations, and thus the frames of a non-AP QSTA using APSD shall have the power management bit in the frame control field set to 1 for buffering to take place at the QAP.

The QAP uses the power-save delivery mechanism currently in effect for a non-AP QSTA to deliver frames to the non-AP QSTA when it is operating in power-save mode. A non-AP QSTA may transition out of the power-save mode, and then back in, by changing the value of power management bit accordingly. A non-AP QSTA with admitted TS having the APSD subfield in the TSInfo field set to 1, but not in power-save mode, will be in active state until it returns to power-save mode.

APSD defines two different types of delivery mechanism, depending upon the SP used. An SP can be either unscheduled or scheduled.

An unscheduled SP begins when the QAP receives any frame from the non-AP QSTA and ends after the QAP has attempted to transmit all frames destined for the non-AP QSTA.

A scheduled SP starts at fixed intervals of time specified in Service Interval field. The first scheduled SP starts when the low order 4 bytes of the TSF timer equals the value specified in Service Start Time field. A non-AP QSTA using scheduled SP shall first wake up to receive downlink frames buffered and/or polls from the AP/HC. The station shall wake up subsequently at a fixed time interval equal to the Service Interval. The QAP may modify the Service Start Time by indicating so in the Schedule element in ADDTS response and Schedule frames.

A scheduled service period begins at the scheduled wakeup time that corresponds to the Service Interval and the Service Start Time indicated in the Schedule element sent in response to a TSPEC. The station shall wake up at a subsequent time whenever

$$(TSF - Service Start Time) \bmod Minimum Service Interval = 0.$$

A non-AP QSTA may signal its desire to utilize APSD as the power-save mode delivery method for individual traffic streams by setting the APSD subfield of the TS info field contained in the TSPEC element

to 1. When a QSTA sets the APSD subfield for any TSPEC to 1, all traffic to the station shall be delivered using APSD. The non-AP QSTA shall request scheduled service periods by sending a TSPEC element that contains a TS Info Field with its Schedule bit set to 1. If the APSD mechanism is supported by the QAP, the QAP shall respond with a Schedule element indicating whether the requested wakeup time can be accommodated by the QAP, and if not, a modified wakeup schedule shall be indicated by the Service Interval and the Service Start Time. If the APSD mechanism is supported by the QAP, the non-AP QSTA shall request unscheduled service periods by sending a TSPEC element that contains a TS Info Field with its Schedule bit set to 0; in this case the QAP shall not return a Schedule element in the response. The Scheduled and Unscheduled Service Period set up through the ADDTS request for power save delivery may be declined by the QAP

APSD shall only be used to deliver unicast frames to a QSTA. Broadcast/multicast frame delivery shall follow the frame delivery rules defined for broadcast/multicast frames as defined in clause 11.2.1.4.

11.2.1.45 AP operation during CP

Change the text in the subclause as shown:

APs shall maintain a Power Management status for each currently associated STA that indicates in which Power Management mode the STA is currently operating. QAPs that implement and signal their support of APSD shall maintain an automatic power-save delivery (APSD) and an access policy status for each currently associated non-AP QSTA that indicates whether the non-AP QSTA is presently using APSD, and the schedule (if any) for the non-AP QSTA. An AP shall, depending on the Power Management mode of the STA, temporarily buffer the MSDU or management frame destined to the STA. A QAP implementing APSD shall, if a non-AP QSTA is using APSD, temporarily buffer the MSDU or management frames destined to that non-AP QSTA. No MSDUs or management frames received for STAs operating in the Active mode or have not indicated their usage of APSD shall be buffered for power management reasons.

- a) MSDUs, or management frames destined for PS STAs, shall be temporarily buffered in the AP. MSDUs, or management frames, destined for APSD capable non-AP QSTAs shall be temporarily buffered in the QAP when requested by the non-AP QSTA. The algorithm to manage this buffering is beyond the scope of this standard, with the exception that if the AP is QoS-enabled, it must preserve the order of arrival of frames per TID, per station basis. The algorithm to manage this buffering shall not favor the transmission of buffered frames destined for non-AP QSTAs using APSD mechanism over the transmission of equal or higher-priority frames destined for non-AP QSTAs not using APSD.
- b) MSDUs, or management frames destined for STAs in the Active mode or have not indicated their usage of APSD, shall be directly transmitted.
- c) At every beacon interval, the AP shall assemble the partial virtual bitmap containing the buffer status per destination for STAs in the PS mode, and shall send this out in the TIM field of the beacon. At every beacon interval, the QAP shall assemble the partial virtual bitmap containing the buffer status per destination for non-AP QSTAs using APSD, and shall send this out in the TIM field of the beacon.²¹ If a non-AP QSTA set up a scheduled SP, then the QSTA automatically transitions to active mode at each SP. Therefore, the QAP shall transmit all frames buffered for the non-AP QSTA following a scheduled SP. If the non-AP QSTA has set up unscheduled SPs, the QAP shall buffer frames until it has received a frame from the non-AP QSTA, which indicates the start of an unscheduled SP. The bit for AID 0 (zero) shall be set when broadcast or multicast traffic is buffered.
- d) All broadcast/multicast MSDUs, with the Order bit in the Frame Control field clear, shall be buffered if any associated STAs are in PS mode.
- e) Immediately after every DTIM, the AP shall transmit all buffered broadcast/multicast MSDUs. The More Data field of each broadcast/multicast frame shall be set to indicate the presence of further

²¹ While sending this indication in the beacon is not essential for the operation of APSD, it simplifies the construction of TIM field at the AP.

- 1 buffered broadcast/multicast MSDUs. If the AP is unable to transmit all of the buffered broadcast/
 2 multicast MSDUs before the TBTT following the DTIM, the AP shall indicate that it will continue to
 3 deliver the broadcast/multicast MSDUs by setting the bit for AID 0 (zero) of the TIM element of
 4 every Beacon frame, until all buffered broadcast/multicast frames have been transmitted.
- 5 f) A single buffered MSDU or management frame for a STA in the PS mode shall be forwarded to the
 6 STA after a PS-Poll has been received from that STA. The More Data field shall be set to indicate
 7 the presence of further buffered MSDUs or management frames for the polling STA. A QAP may set
 8 the More Data bit in a QoS +CF-Ack frame to 1, in response to a QoS Data frame to indicate that it
 9 has one or more pending frames buffered for the PS station identified by the RA address in the
 10 QoS+CF-Ack. A QAP may also set the More Data bit in ACK frame in response to a QoS Data
 11 frame to indicate that it has one or more pending frames buffered for the PS station identified by the
 12 RA address in the ACK frame, if that PS Station has set the "More Data Ack" subfield in the QoS
 13 Capability Information Element to 1. Further PS-Poll frames from the same STA shall be
 14 acknowledged and ignored until the MSDU or management frame has either been successfully
 15 delivered, or presumed failed due to maximum retries being exceeded. This prevents a retried PS-
 16 Poll from being treated as a new request to deliver a buffered frame.
- 17 g) At each APSD SP for a non-AP QSTA, the QAP shall attempt to transmit all frames destined for the
 18 non-AP QSTA. The More Data bit of the directed data or management frame shall be set to 1 to
 19 indicate the presence of more frames that are destined for that non-AP QSTA. For all frames except
 20 for the final frame of the SP, the EOSP subfield of the QoS Control field of the QoS data frame shall
 21 be set to 0 to indicate the continuation of the SP. A QAP may also set the More Data bit to 1 in a
 22 QoS +CF-Ack frame in response to a QoS Data frame to indicate that it has one or more pending
 23 frames buffered for the target station identified by the RA address in the QoS +CF-Ack. The QAP
 24 considers APSD QSTA to be in Active state after it has sent a QoS +CF-Ack, with the More Data bit
 25 set, to the APSD QSTA. If necessary the QAP may generate an extra QoS Null frame, with the more
 26 data subfield set to 0 and the EOSP set to 1. When the QAP has transmitted a directed frame to the
 27 non-AP QSTA the EOSP subfield set to 1 during the SP, it shall not transmit any more frames using
 28 this mechanism until the next SP. The QAP shall use EOSP=1 to indicate the end of SP in APSD.
 29 The More Data Bit shall be used simply to indicate whether there is data still buffered in the AP at
 30 the end of the SP. The usage of More Data and EOSP subfields is summarized in table 26.1.
- 31 h) If the QAP does not receive an acknowledgement to a directed MPDU or management frame sent to
 32 a STA in power-save mode following receipt of a PS Poll from that STA, it may retransmit the frame
 33 fewer times than the Max Retry Limit before the next TIM, but it shall retransmit that frame at least
 34 once before the next TIM, time permitting – subject to the Max Retry Limit and the MSDULifetime
 35 limit. If an Ack to the retransmission is not received, it may wait until the next TIM to further
 36 retransmit that frame subject to the Max Retry Limit and the MSDULifetime limit.
- 37 i) If the QAP does not receive an acknowledgement to a directed MPDU or management frame sent
 38 with the More Data sub-field set to 0 or the EOSP set to 1, it may retransmit the frame fewer times
 39 than the Max Retry Limit within the same service period, but it shall retransmit that frame at least
 40 once within the same service period – subject to the Max Retry Limit and the MSDULifetime limit.
 41 If an Ack to the retransmission is not received, it may wait until the next service period to further
 42 retransmit that frame subject to the Max Retry Limit and the MSDULifetime limit.
- 43 gj) An AP shall have an aging function to delete pending traffic when it is buffered for an excessive
 44 time period. The QAP may base the aging function on the listen interval specified by the non-AP
 45 QSTA in the (re)association request.
- 46 hk) When an AP is informed that a STA changes to the Active mode, then the AP shall send buffered
 47 MSDUs and management frames (if any exist) to that STA without waiting for a PS-Poll. Whenever
 48 a QAP is informed that an APSD capable non-AP QSTA changes its power-save mode to not be
 49 APSD mode, then the QAP shall send buffered MSDUs and management frames (if any exist) to that
 50 non-AP QSTA according to the rules corresponding to the current power-save mode of the non-AP
 51 QSTA.

Insert the following table at the end of subclause 11.2.1.5.

Table 26.1 – Usage of More Data and EOSP subfields

More Data	EOSP	Description	AP Behavior	STA Behavior
0	0	AP has no more buffered frames SP does not end. Not to be used for non-AP QSTAs using unscheduled APSD.	QAP engages in follow-up actions (e.g. sending Poll) during the remaining SP	Non-AP QSTA stays awake.
1	0	QAP has buffered frames SP does not end	QAP continues delivering buffered DL frames	Non-AP QSTA stays awake to receive DL frames
0	1	QAP has no more buffered frames SP ends	QAP transfers no more frames to the non-AP QSTA until next SP	Non-AP QSTA returns to Doze mode
1	1	QAP has buffered frames SP ends	QAP transfers no more frames to the non-AP QSTA until next SP	Non-AP QSTA may do one of the following: Non-AP QSTA returns to Doze mode immediately and waits until next SP to receive remaining buffered frames Non-AP QSTA remains awake and may use PS-Poll to receive remaining frames

11.2.1.67 Receive operation for STAs in PS mode during the CP

Change item marked “e)” as shown:

- e) When ReceivedDTIMs is true, the STA shall wake up early enough to be able to receive every DTIM. A STA receiving broadcast/multicast MSDUs shall remain awake until the More Data field of the broadcast/multicast MSDUs indicates there are no further buffered broadcast/multicast MSDUs, or until a TIM is received indicating there are no more buffered broadcast/multicast MSDUs. If a non-AP QSTA receives a QoS+CF-Ack from its QAP, with the More Data bit set to 1 or a TIM with the bit for its AID set, then the QSTA must operate exactly as if it received a TIM with its AID bit set. If a non-AP QSTA has set the “More Data Ack” subfield in QoS Capability Information Element to 1, then if it receives an ACK frame from its QAP, with the More Data bit set to 1, the QSTA shall operate exactly as if it received a TIM with its AID bit set. For example, a QSTA that is using the PS-Poll delivery method shall issue a PS-Poll frame to retrieve a buffered frame.

The following subclause heading is shown only to indicate the new number. There are no changes to the text in the subclause:

11.2.1.8 Receive operation for STAs in PS mode during the CFP

Insert the following subclause after subclause 11.2.1.8 and further renumber the following subclauses:

11.2.1.9 Receive operation for non-AP QSTAs using APSD

A non-AP QSTA using APSD shall operate as follows to receive an MSDU or management frame from the QAP:

- a) If a scheduled SP has been set up, the non-AP QSTA transitions to Active mode at its scheduled Start Time. (The non-AP QSTA shall wake up early enough to receive transmissions at the scheduled SP).
- b) If the non-AP QSTA is initiating an unscheduled SP, the non-AP QSTA transitions to Active mode and transmits a frame to the QAP.
- c) The non-AP QSTA shall remain awake until it receives a QoS data type frame addressed to it, with the EOSP subfield in QoS Control field set to 1.

The following two subclause headings are shown only to indicate the new numbers. There are no changes to the text in the subclauses:

11.2.1.10 STAs operating in the Active mode

11.2.1.11 AP aging function

11.2.2 Power management in an IBSS

Insert the following paragraph at the end of the subclause, above 11.2.2.1:

In a QIBSS, a QSTA shall not operate in power save mode while it has Block Ack set up with another QSTA. A QSTA may be in power-save mode before the set up of Block Ack. Once Block Ack is set up with another QSTA, the QSTA suspends the power save mode and shall always remain awake.

Insert the following subclauses after subclause 11.3.4 and renumber 11.4 as 11.8:

11.4 Traffic Stream operation

11.4.1 Introduction (Informative)

A TSPEC describes the traffic characteristics and the QoS requirements of a traffic stream (TS). The main purpose of the TSPEC is to reserve resources within the HC and modify the HC's scheduling behavior. It also allows other parameters to be specified that are associated with the traffic stream, such as a traffic classifier and Ack policy.

A TS may have one or more TCLAS (within the discretion of the QSTA that sets up the stream) associated with it. The QAP uses the parameters in the TCLAS elements to filter the MSDUs belonging to this TS so that they can be delivered with the QoS parameters that have been set up for the TS.

TSPEC and the optional TCLAS elements are transported on the air by the ADDTS, in the corresponding QoS Action frame or (Re)Association Request or Response and across the MLME SAP by the MLME-ADDTS primitives.

Following a successful negotiation, a traffic stream is created, identified within the non-AP QSTA by its TSID and Direction, and identified within the HC by a combination of TSID, Direction and non-AP QSTA address.

1 It is always the responsibility of the non-AP QSTA to initiate the creation of a TS regardless of its direction.

2 In the direct link case, it is the responsibility of the non-AP QSTA that is going to send the Data to create the
3 TS. In this case, the non-AP QSTA negotiates with the HC to gain TXOPs that it uses to send the Data.
4 There is no negotiation between the originator and recipient non-AP QSTAs concerning the TS – the
5 originator can discover the capabilities of the recipient (rates, BlockAck) using the DLP protocol. It does not
6 negotiate the use of these on a per-TS basis with the intended recipient.

7 In the case of traffic relayed by a QAP, sending and receiving non-AP QSTA may both create individual TS
8 for the traffic. Any traffic classifier created for the downlink TS applies equally regardless of whether the
9 source is in the same BSS or reached through the DS.

10 A non-AP QSTA may simultaneously support up to 8 traffic streams from the HC to itself and up to 8 traffic
11 streams from itself to other QSTAs, including the HC. The actual number it supports may be less due to
12 implementation restrictions.

13 A HC may simultaneously support up to 8 downlink traffic streams and up to 8 uplink traffic streams per
14 associated non-AP QSTA. The actual number it supports may be less due to implementation restrictions.

15 The ability to transmit traffic for admitted TSPECs over the EDCA is controlled by the Access Policy
16 subfield of the TS Info field. If the non-AP QSTA associated with the TSPEC requests it, and the HC grants
17 it, then it is permitted for frames associated with that stream to be transmitted over the EDCA, within the
18 constraints specified in subclause 9.9.2.3.

19 **11.4.2 TSPEC Construction (informative)**

20 TSPECs are constructed at the SME, from application requirements supplied via the SME, and with
21 information specific to the MAC layer. There are no normative requirements on how any TSPEC is to be
22 generated. However, in Annex H.3.2, a description is given on how and where certain parameters may be
23 chosen.

24 **11.4.3 TS lifecycle**

25 Figure 68.1 summarizes the TS lifecycle. (Using the HMSC syntax defined in ITU Z.120).

26 Initially a TS is inactive. A QSTA shall not transmit any QoS Data frames using an inactive TS.

27 Following a successful TS Setup initiated by the non-AP QSTA, the TS becomes active, and either the non-
28 AP QSTA or HC may transmit QoS Data frames using this TSID (according to the Direction field).

29 While the TS is active, the parameters of the TSPEC characterizing the TS can be re-negotiated, initiated by
30 the non-AP QSTA. This negotiation can succeed – resulting in a change to the TSPEC, or can fail, resulting
31 in no change to the TSPEC.

32 An active TS becomes inactive following a TS deletion process initiated at either non-AP QSTA or HC. It
33 also becomes inactive following a TS timeout detected at the HC.

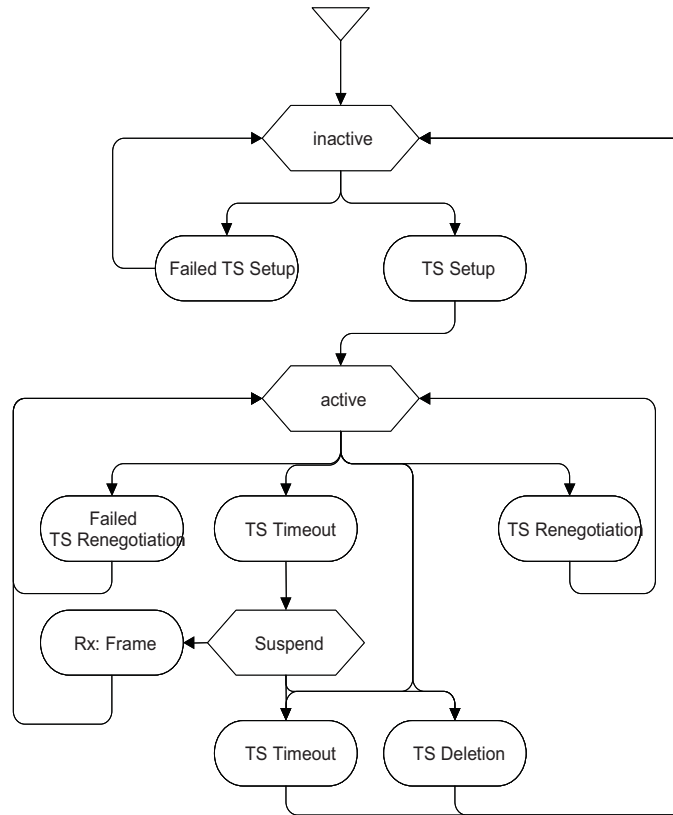


Figure 68.1 – TS Lifecycle (Informative)

11.4.4 TS setup

Figure 68.2 shows the sequence of messages occurring at a TS setup. This message sequence in this figure and in the subsequent figures does not show the ACK.

The non-AP QSTA SME decides that a TS needs to be created. How it does this, and how it selects the TSPEC parameters, is beyond the scope of this specification. The SME generates an MLME-ADDTS.request containing a TSPEC. A TSPEC may also be generated autonomously by the MAC without any initiation by the SME. However, if a TSPEC is generated subsequently by the SME, the TSPEC containing the same TSID generated autonomously by the MAC shall be overridden. If one or more TSPECs are initiated by the SME, the autonomous TSPEC shall be terminated.

The non-AP QSTA MAC transmits the TSPEC in an ADDTS request in the corresponding QoS Action frame or the (re)association request frame to the HC and starts a response timer called ADDTS timer of duration `dot11ADDTSResponseTimeout`.

The HC MAC receives this management frame and generates an MLME-ADDTS.indication primitive to its SME containing the TSPEC.

The SME in the HC decides whether to admit the TSPEC as specified, or refuse the TSPEC, or not admit but suggest an alternative TSPEC and generates an MLME-ADDTS.response primitive containing the TSPEC and a ResultCode value. The contents of the TSPEC and status field contain values specified in 10.3.11.4.2.

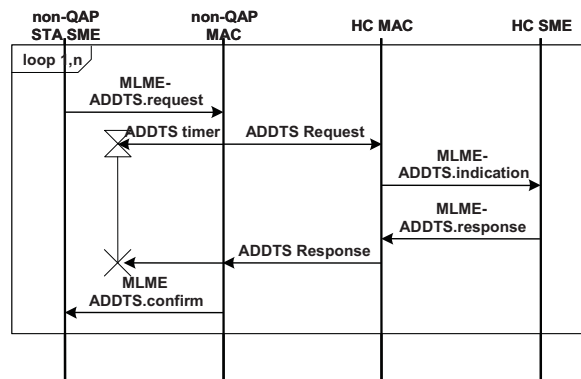
The HC MAC transmits an ADDTS response in the corresponding QoS Action frame or (re) association response containing this TSPEC and status. The encoding of the ResultCode values to Status Code field values are defined in Table 26.2.

Table 26.2 – Encoding of ResultCode to Status Code field value

ResultCode	Status Code
SUCCESS	0
INVALID-PARAMETERS	38
REJECTED_WITH_SUGGESTED_CHANGES	39
REJECTED_FOR_DELAY_PERIOD	40

The non-AP QSTA MAC receives this management frame and cancels its ADDTS timer. It generates an MLME-ADDTS.confirm to its SME containing the TSPEC and status.

The non-AP QSTA SME decides whether the response meets its needs or not. How it does this is beyond the scope of this specification. If the ResultCode is “SUCCESS”, the TS enters into an active state. Otherwise, the whole process can be repeated using the same TSID and Direction and a modified TSPEC until the non-AP QSTA SME decides that the granted TXOP is adequate or inadequate and cannot be improved.

**Figure 68.2 – TS Setup**

The parameters that are set for a TS can be modified in a similar manner, when such a request is generated by the SME through ADDTS.request. When a request for the modification of the TS parameters is accepted by the HC, it shall reset both the suspension interval and the inactivity interval timers.

If the AP/HC grants a TXOP for an ADDTS request with Ack Policy set to “Block Acknowledgement” and Direction set to either “Downlink” or “Bidirectional” then it shall initiate a Block Ack negotiation by sending an ADDBA request to the STA which originated the ADDTS request. If a non-AP QSTA is granted a TXOP for an ADDTS request with Ack Policy set to “Block Acknowledgement” and Direction set to either “Uplink” or “Bidirectional” then it shall initiate a Block Ack negotiation by sending an ADDBA request to the AP which granted the TXOP.

11.4.5 Failed TS setup

There are two possible types of Failed TS setup:

1. The transmission of ADDTS request frame failed
2. No ADDTS response frame is received from the HC (for example because of delay due to congestion, or because the response frame cannot be transmitted)

Figure 68.3 summarizes the remaining two cases. The MLME shall issue a MLME-ADDTS.confirm, with ResultCode “TRANSMISSION_FAILURE” in the former case and with the ResultCode “TIMEOUT” in the

latter case. In both cases, the non-AP QSTA MAC shall send a DELTS to the HC specifying the TSID and Direction of the failed request just in case the HC had received the request and it was the response that was lost.

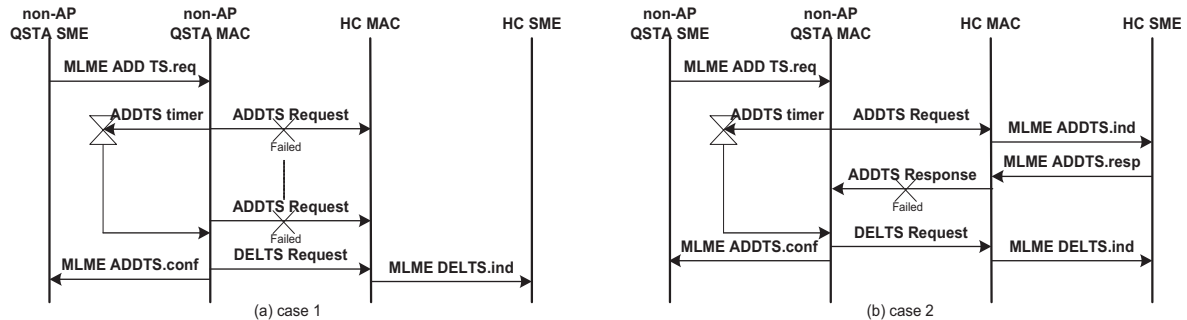


Figure 68.3 – Failed TS Setup detected within non-AP STA MAC

11.4.6 Data transfer

After the set up of a TSPEC, MSDUs are classified above the MAC and are sent to the MAC through the MAC-SAP using the service primitive MA-UNITDATA.request with the priority parameter encoded to the TSID. The MAC delivers the MSDUs based on a schedule using frames of subtype QoS data. In the case of a non-AP QSTA, the MSDUs are transmitted using frames of subtype QoS data, when the non-AP QSTA is polled by the HC.

The generation of the associated TSID is done by a classifier above the MAC and it may use the associated TCLAS elements if any are present. When there are multiple TCLAS and if the Processing subfield of TCLAS Processing information element is set to 0, the priority parameter in the associated MA-UNITDATA.request primitive is set to TSID if the classifier can match the parameters in all the TCLAS elements associated with the TS. When there are multiple TCLAS and if the Processing subfield of the TCLAS Processing information element is set to 1, the priority parameter in the associated MA-UNITDATA.request primitive is set to TSID if the classifier can match the parameters in at least one of the TCLAS elements associated with the TS.

When an MSDU arrives at the MAC with a TSID for which there is no associated TSPEC, a default TSPEC should be generated by the MAC to support the traffic flow where possible. The default set of mandatory parameters for a TSPEC are shown in Table 26.3. The user priority for the TSPEC is set to the three least significant bits of the TSID i.e., treat as if the three least significant bits were 802.1D user priority. The Dialog Token within the TSPEC generated within the MAC is restricted to values from 128-255. If no TSPECs are available or the TSPEC is rejected by the HC, the MSDUs should be sent using EDCA with AC set to AC_BE. The transmitted frames should encode the TID field of the QoS Control with the priority parameter supplied by the service primitive MA-UNITDATA.request.

Table 26.3 – Default values for mandatory TSPEC parameters

UP	Mean Data Rate (bps)	Nominal MSDU Size (octets)	Service Interval (microseconds)
0	0	0	0
1	0	0	0
2	0	0	0
3	0	0	100,000
4	0	0	50,000

5	0	0	20,000
6	0	0	10,000
7	0	0	5,000

11.4.7 TS deletion

There are two types of TS deletion: non-AP QSTA-initiated and HC-initiated. In both cases, the SME entity above the MAC generates an MLME-DELTS.request specifying the TSID and Direction of the TS to be deleted. This causes the MAC to send a DELTS management action frame.

The TS is considered inactive within the initiating MAC when the ACK frame to the management action frame is received. No management action frame response is generated.

Figure 68.4 shows the case of TS deletion initiated by the non-AP QSTA. The case of HC-initiated TS deletion is the same with the non-AP QSTA and HC labels swapped over.

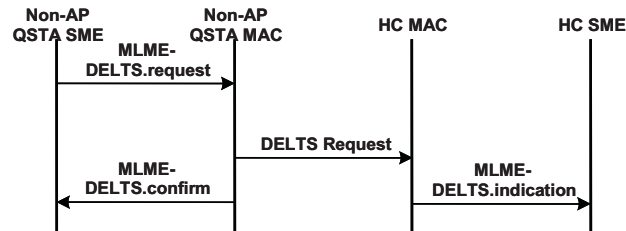


Figure 68.4 TS Deletion (non-AP STA-initiated)

An HC should not delete a TSPEC without a request from the SME except due to inactivity (see 11.4.8).

11.4.8 TS timeout

TS timeout is detected within the HC MAC when no traffic is detected on the TS within the inactivity timeout specified when the TS was created.

For an uplink TS, the timeout is based on the arrival of correctly received MSDUs within the MAC after any decryption and reassembly.

For a downlink TS, the timeout is based on the arrival of valid MA-UNITDATA.request primitives using this TS at the MAC SAP.

For a direct link TS, inactivity is considered to have happened if one of the two following happen:

- returning QoS Null immediately after SIFS interval that contains a zero queue size field in the QoS control field in response to a QoS CF-Poll
- no QoS Null frame indicating the queue size for related TSID within a TXOP.

Any other use of a polled TXOP delivered to the non-AP STA is considered to be activity on all direct link TS associated with that non-AP QSTA. Detection of inactivity of this type is optional.

In response to an inactivity timeout, the HC shall send a DELTS to the non-AP QSTA, and inform its SME using the MLME-DELTS.indication.

The case of uplink TS timeout is shown in figure 68.5.

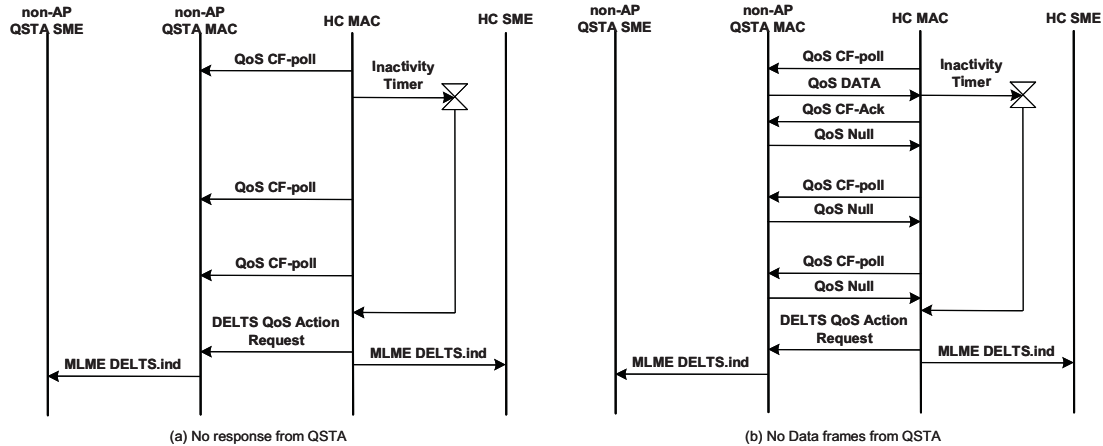


Figure 68.5 TS Timeout

11.4.9 TS suspension

TS suspension occurs within the HC MAC when no traffic is detected on the TS within the suspension interval specified when the TS was created. The generation of successive QoS (+)CF-Poll is stopped for this TS, but the TS is considered active.

11.4.10 TS Reinstatement

TS reinstatement can occur within the HC MAC when either a QoS-data or a QoS –null frame is received from a non-AP QSTA containing the TID of a suspended TS. The generation of successive QoS (+) CF-Polls is resumed for such a TS upon reinstatement. A non-AP QSTA, which has a TS suspended due to inactivity can initiate reinstatement of that TS by sending either a QoS-data or a QoS-null frame containing the corresponding TID. A QoS-null frame sent to reinstate suspended TS may access the medium using the top user priority.

11.5 Block Ack operation

Block Ack may be set up at the MAC (see 9.10.2) or by the initiation of SME. The set up and deletion of Block Ack at the initiation of the SME is described in this subclause.

11.5.1 Set up and modification of the Block Ack parameters

The procedures for setting up and modifying the Block Ack parameters are described in 11.5.1.1 and 11.5.1.2 respectively and illustrated in Figure 68.6.

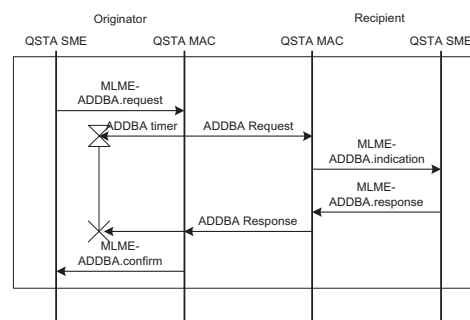


Figure 68.6 – Block Ack set up

11.5.1.1 Procedure at the originator

Upon receipt of MLME-ADDBA.request, an initiating QSTA that intends to QoS type Data frames under the Block Ack mechanism shall set up the Block Ack using the following procedure.

- a) Check if the intended peer QSTA is capable of participating in the Block Ack mechanism by discovering and examining its “Block Ack” capability bit. If the recipient is capable of participating, the originator sends an ADDBA frame indicating the TID and the Buffer Size.
- b) If an ADDBA response frame is received with the matching Dialog Token and the TID, and with a Status Code set to a value of 0, the QSTA has established Block Ack mechanism with the recipient QSTA and the MLME shall issue an MLME-ADDBA.confirm indicating the successful completion of the Block Ack set up.
- c) If an ADDBA response frame is received with the matching Dialog Token and the TID, and with a Status Code set to a value other than 0, the QSTA has not established Block Ack mechanism with the recipient QSTA and the MLME shall issue an MLME-ADDBA.confirm indicating the failure of the Block Ack set up.
- d) If there is no response from the recipient within dot11ADDBAFailureTimeout, the QSTA has not established Block Ack mechanism with the recipient QSTA and the MLME shall issue an MLME-ADDBA.confirm with ResultCode “TIMEOUT”.

11.5.1.2 Procedure at the recipient

A recipient shall operate as follows in order to support Block Ack initialization and modification.

- a) Whenever an ADDBA request frame is received from another QSTA, the MLME shall issue an MLME-ADDBA.indication.
- b) Upon receipt of the MLME-ADDBA.response, the QSTA shall respond by an ADDBA response frame with a Result code as defined in 7.4.4.2.
 - 1) If the ResultCode is “SUCCESS” the BlockAck is considered to be established with the originator. Contained in the frame are the type of Block Ack and the number of buffers that have been allocated for the support of this Block.
 - 2) If the ResultCode is “REFUSED” the Block Ack is not considered to have been established.

The encoding of the ResultCode values to Status Code field values are defined in Table 26.4.

Table 26.4 – Encoding of ResultCode to Status Code field value

ResultCode	Status Code
SUCCESS	0
REFUSED	37
INVALID-PARAMETERS	38

11.5.2 Tear down of the Block Ack mechanism

The procedure at the two QSTAs is described in 11.5.2.1 and 11.5.2.2, and illustrated in Figure 68.7.

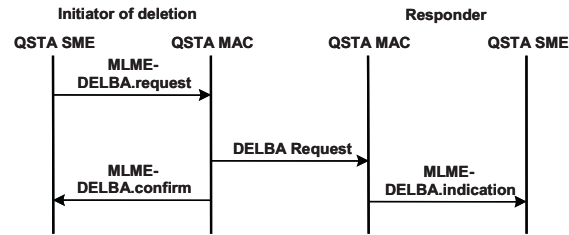


Figure 68.7 – Block Ack Deletion

11.5.2.1 Procedure at the initiator of the Block Ack teardown

Upon receipt of an MLME-DELBA.request, the QSTA MAC/MLME shall tear down the Block Ack using the following procedure:

- The QSTA shall transmit a DELBA frame.
- Upon the receipt of an acknowledgement to the DELBA frame, the MLME issues a MLME-DELBA.confirm.

11.5.2.2 Procedure at the recipient of the DELBA frame

A QSTA shall issue a MLME-DELBA.indication with the parameter Reason having a value of "REQUESTED" when a DELBA frame is received.

11.5.3 Error recovery upon a peer failure

Every station shall maintain an inactivity timer for every negotiated Block Ack set up. The inactivity timer is reset whenever MPDUs corresponding to the TID for which the Block Ack policy is set are received, regardless of the ack policy that is used for the transmission of that frame. The inactivity timer is not reset whenever MPDUs corresponding to other TIDs are received. When a timeout of BlockAckTimeout is detected, the QSTA shall send a DELBA frame to the peer QSTA, and shall issue a MLME-DELBA.indication with the Reason parameter having a value of "TIMEOUT". The procedure is illustrated in Figure 68.8.

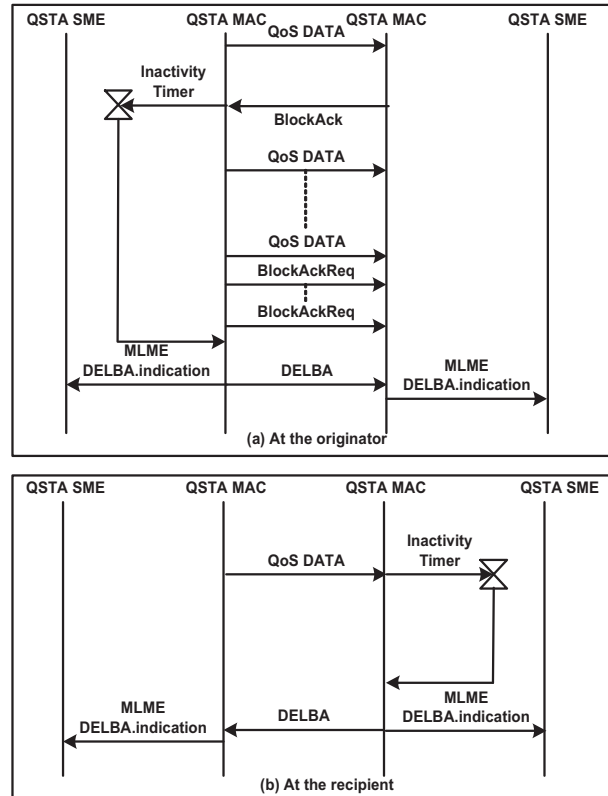


Figure 68.8 – Error Recovery by the Receiver upon a peer failure

11.6 Higher-layer timer synchronization

11.6.1 Introduction (Informative)

Higher-layer timer synchronization is beyond the scope of this standard. However explanation as to how the constructs in this standard can be used to support such capabilities may be useful to the designer.

One way to accomplish synchronization across a BSS is by multicasting synchronization (Sync) packets from the higher layers containing a time stamp and a sequence number. These packets would be opaque to the MAC and would be transported in the same way as any other MSDU (most likely addressed to the multicast address). Sync packets would be treated as a type of management packet by the higher layers. The time stamp in the synchronization packet would contain the higher layer clock value at the time when the previous Sync packet was transmitted. The sequence number parameter has a value equal to the sequence number of the MSDU in which the time stamp is provided.

The reason why the packet would contain the time stamp for the previous Sync packet (rather than the current packet) is that hardware and layering constraints would prohibit the ability to provide a time stamp for the exact instant the current packet is transmitted within that packet. However, a MLME-HL-SYNC.indication allows the transmitting QSTA to know exactly when each Sync packet is transmitted. A receiving QSTA can similarly note the time when each Sync packet is received as well as the sequence number for that frame. The receiving QSTA would save this receive time indication for each packet along with its sequence number, and compare the indication of the previously received Sync packet to the time stamp in the most currently received packet. This comparison allows the station to compute an offset, which can be used to adjust its time reference to match that of the synchronization source. The sequence number would ensure that the correct packet is being used for time stamp comparison. It is possible a packet is lost; in this case the received time stamp for the lost packet should be discarded.

1 The last symbol of the Sync frame is indicated by the Physical Layer using the PHY-TXEND.confirm and
2 PHY-RXEND indication primitives in the transmitter and receiver of the Sync frame respectively. Practical
3 limits on the coincidence of this indication and the last symbol of the Sync frame are implementation
4 dependent. The accuracy of this technique also depends on the propagation delay between the source and
5 receiving channel. However, both the time difference between the Physical Layer indication and the last
6 symbol of the Sync frame, and the propagation delay can be considered as fixed-delay components and,
7 therefore, only contribute to the fixed time difference between the transmitter and receiver stations' clocks,
8 and do not contribute to jitter. An implementation-dependent scheme can be used to cancel or minimize this
9 fixed time difference.

10 The Sync frame may also be relayed through the AP. In this case, the non-AP QSTA that generates the time
11 stamps notes the reception of the multicast Sync frame from the AP as the indication to save the higher clock
12 value for the next Sync frame. Receiving QSTAs would also similarly note the time when each Sync packet
13 is received from the AP. The sequence number would include a value corresponding to the frame received
14 from the AP.

15 Other implementations (aside from what is described here) may also be supported.

16 11.6.2 Procedure at the QSTA

17 A MAC that supports the MLME-HL-SYNC primitives shall respond to a MLME-HL-SYNC.request with an
18 MLME-HL-SYNC.confirm containing a ResultCode of SUCCESS. This confirms to the requesting
19 application that the specified group address has been entered into a MAC table of group addresses for which
20 MLME-HL-SYNC.indication shall be provided.

21 In order to determine whether or not to provide an MLME-HL-SYNC.indication for a particular data frame, a
22 MAC that supports MLME-HL-SYNC primitives compares the Address 1 field in a data frame's MAC header
23 against a list of group addresses previously registered by an MLME-HL-SYNC.request. If the MAC and the
24 transmitter of the synchronization frame are co-located within the same station, the MLME-HL-
25 SYNC.indication shall occur when the last symbol of a matching data frame is transmitted. Otherwise, the
26 indication shall occur when the last symbol of the matching data frame is received. In both cases, the MLME-
27 HL-SYNC.indication provided is simultaneous (within implementation-dependent delay bounds) with the
28 indication provided to other stations within the BSS for the same data frame.

29 11.7 DLP operation

30 In general, STAs are not allowed to transmit frames directly to other STAs in a BSS and should always rely
31 on the AP for the delivery of the frames. However, STAs with QoS facility (i.e., QSTAs) may transmit frames
32 directly to another QSTA by setting up such data transfer using Direct Link Protocol (DLP). The need for this
33 protocol is motivated by the fact that the intended recipient may be in Power Save Mode, in which case it can
34 only be woken up by the QAP. The second feature of DLP is to exchange rate set and other information
35 between the sender and the receiver. Finally, DLP messages can be used to attach security information
36 elements. The security elements and the associated security handshake are not part of this standard.

37 This protocol prohibits the stations going into power-save for the duration of the Direct Stream as long as
38 there is an active direct link set up between the two stations.

39 DLP does not apply in a QIBSS, where frames are always sent directly from one STA to another.

40 The DLP set up handshake is illustrated in Figure 68.9 and involves 4 steps.

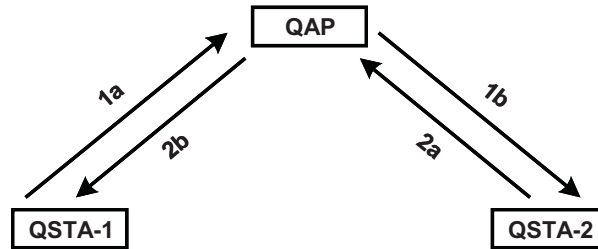


Figure 68.9 – The four steps that are involved in the Direct Link handshake.

- A station QSTA-1 that has data to send invokes DLP and sends a DLP Request frame to the AP (1a in Figure 68.9). This request contains the rate set, and capabilities of QSTA-1, as well as the MAC addresses of QSTA-1 and QSTA-2.
- If QSTA-2 is associated in the BSS, direct streams are allowed in the policy of the BSS and QSTA-2 is indeed a QSTA (i.e. compliant to this standard), the AP shall forward the DLP-request to the recipient, QSTA-2 (1b in Figure 68.9).
- If QSTA-2 accepts the direct stream, it shall send a DLP-response frame to the AP (2a in Figure 68.9), which contains the rate set, (extended) capabilities of QSTA-2 and the MAC addresses of QSTA-1 and QSTA-2.
- The AP shall forward the DLP-response to QSTA-1 (2b in Figure 68.9), after which the direct link becomes active and frames can be sent from QSTA-1 to QSTA-2 and from QSTA-2 to QSTA-1.

11.7.1 DLP set up

The DLP set up message flow is illustrated in figure 68.10.

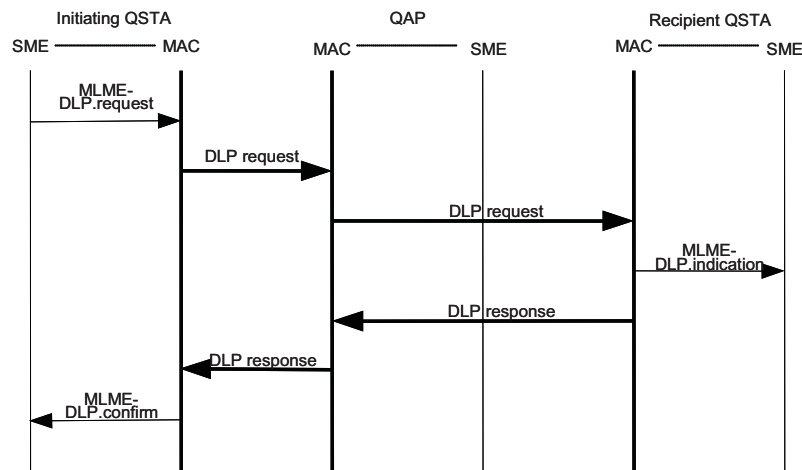


Figure 68.10 – Direct Link Protocol set up message flow

11.7.1.1 Set up procedure at the QSTA

All QSTAs shall maintain a list of QSTAs with which a direct link has been established.

Upon receipt of MLME-DLP.request from the SME, the QSTA

- shall issue an MLME-DLP.confirm with Result Code of “SUCCESS”, if the PeerMACAddress of MLME-DLP.request is in the list of QSTAs with which Direct Link has been established;
- else shall initiate the set up of the direct link by sending the DLP request frame to the AP (labeled as 1a in Figure 68.9). If the QSTA does not receive a DLP response frame within DLPResponseTimeout after sending the DLP request frame, the QSTA shall issue an MLME-DLP.confirm with the Result Code of “TIMEOUT”.

Upon receipt of the DLP request frame (from the QAP – labeled as 1b in Figure 68.9), the QSTA MAC shall send to the AP a DLP response frame (labeled as 2a in Figure 68.9) with a Result code of

- “SUCCESS”, if the QSTA is willing to participate in the Direct Link with the source QSTA. The QSTA shall also issue an MLME-DLP.indication to the SME and shall add the source QSTA to the list of the QSTAs, if already not present, with which Direct Link has been established.
- “Refused”, if the QSTA is not willing to participate in the Direct Link.

Upon receipt of the DLP response frame (from the QAP – labeled as 2b in Figure 68.9), the QSTA shall issue an MLME-DLP.confirm.

- If the Result code is “SUCCESS” the Direct Link is considered to be established with the Destination QSTA in the DLP Response frame, and the QSTA MAC shall add the destination QSTA to the list of QSTAs with which Direct Link has been established.
- If the Result code is “Refused” the Direct Links is not considered to have been established.

11.7.1.2 Set up procedure at the QAP

Upon receipt of the DLP request frame (labeled as 1a in Figure 68.9), the QAP shall

- send DLP Response frame to the QSTA that sent the DLP request with Result Code of “Not Allowed in the BSS”, if Direct Links are not allowed in the QBSS (labeled as 2b in Figure 68.9).
- send DLP Response frame to the QSTA that sent the DLP request with Result Code of “Not Present”, if the destination QSTA is not present in the QBSS (labeled as 2b in Figure 68.9).
- send DLP Response frame to the QSTA that sent the DLP request with Result Code of “Not a QSTA”, if the destination STA does not have QoS facility (labeled as 2b in Figure 68.9).
- send the DLP Request frame, with all fields having the same value as the DLP Request frame received by the AP, to the destination QSTA (labeled as 1b in Figure 68.9).

Upon receipt of the DLP response frame from a QSTA (labeled as 2a in Figure 68.9), the QAP shall send DLP Response frame to the source QSTA (labeled as 2b in Figure 68.9).

The mapping of the Status Code field values to the ResultCode parameter values in MLME-DLP.confirm are defined in Table 26.5.

Table 26.5 – Mapping of Status Code field value to ResultCode

ResultCode	Status Code
SUCCESS	0
REFUSED	37
INVALID-PARAMETERS	38
NOT_ALLOWED	41
NOT_PRESENT	42
NOT_QSTA	43

11.7.2 Data transfer after DLP set up

Both the QSTAs may use Direct Link for data transfers using any of the access mechanisms defined in this standard. QSTAs may also set up Block Ack if needed. If needed the QSTAs may set up traffic streams with the HC to ensure they have enough bandwidth or use polled TXOPs for data transfer. A protective mechanism (such as transmitting using HCCA, using RTS/CTS or the mechanism described in 9.10) should be used to reduce the probability of other stations interfering with the direct link transmissions.

11.7.3 DLP Teardown

The DLP tear down message flow is illustrated in figure 68.11.

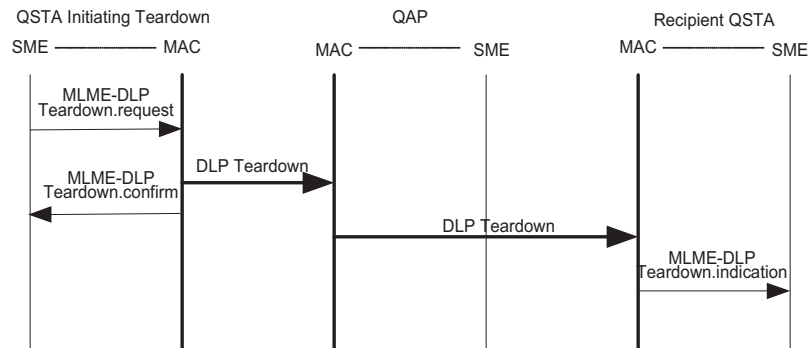


Figure 68.11 – Direct Link Protocol tear down message flow

11.7.3.1 Teardown procedure at the QSTA

Upon receipt of MLME-DLPTeardown.request from the SME, the QSTA shall initiate the teardown of the direct link by sending the DLP teardown to the QAP. If the transmission of the frame is successful, it shall issue an MLME-DLPTeardown.confirm with Result Code of “Success”. If the frame could not be transmitted, it shall issue an MLME-DLPTeardown.confirm with Result Code of “Failure”.

Upon receipt of the DLP teardown frame (from the QAP), the QSTA shall issue an MLME-DLPTeardown.indication to the SME and shall delete the QSTA from the list of the QSTAs with which Direct Link has been established.

11.7.3.2 Teardown procedure at the AP

Upon receipt of the DLP teardown frame from a QSTA, the QAP shall send DLP teardown frame to the destination QSTA.

11.7.4 Error recovery upon a peer failure

Every station shall maintain an inactivity timer for every negotiated DLP link (i.e., stations on both sides of the link maintain these timers). The DLP inactivity timer shall be reset for every successful frame transmission or reception for that DLP link. The direct link becomes inactive when no frames have been exchanged as part of the direct link for the duration of DLP Timeout Value, if the DLP Timeout Value is set to a non-zero value during the DL set up. When the direct link becomes inactive due to the timeout, the MAC issues an MLME-DLPTeardown.indication to the SME and sends a DLP Teardown frame to the QAP, with the peer MAC address as the Destination MAC address. All frames shall henceforth be sent via the QAP.

If there has been a traffic stream set up for the data transfer, it is the responsibility of the QSTAs to renegotiate the parameters of the TSPEC with the HC.

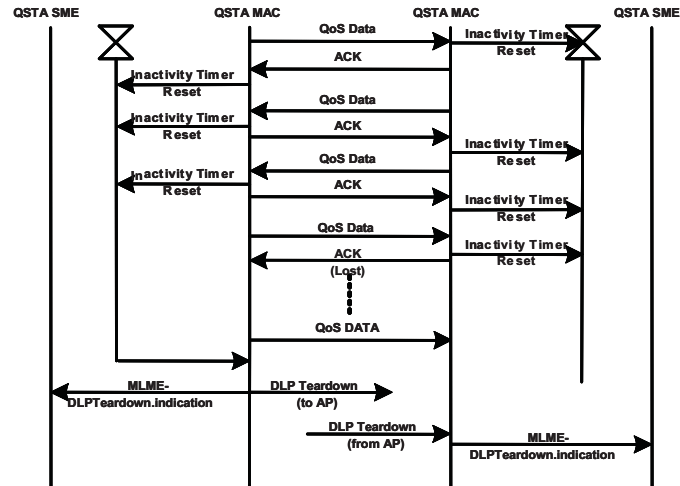


Figure 68.8 – Error recovery upon a peer failure

1 Annex A

2 *Add the following entries to the table in A.4.3:*

3 A.4.3 IUT configuration

Item	IUT configuration	References	Status	Support
CF12	QoS supported	9.9, 9.10	O	Yes, No

4
5 *Insert the following at the end of A.4:*

6 A.4.x QoS Base Functionality

Item	Protocol Capability	References	Status	Support
QB1	QoS frame format	7.2.1.1 – 7.2.1.3, 7.2.2, 7.2.3.1, 7.2.3.4 - 7.2.3.7, 7.2.3.9, 7.2.3.12,	CF12:M	Yes, No
QB2	Per TID duplicate detection	7.1.3.4, 7.1.3.5, 9.2.9	CF12:M	Yes, No
QB3	Decode of no-ack policy in QoS data frames	7.1.3.5.3, 9.9.1.4, 9.9.1.5, 9.9.3.1, 9.9.3.2	CF12:M	Yes, No
QB4	Block acknowledgements	7.2.1.7, 7.2.1.8, 7.4.4, 9.10, 11.5	CF12:O	Yes, No
QB5	Automatic Power Save Delivery	7.4.2, 11.2.3	CF12:O	Yes, No
QB6	Direct Link Protocol	7.3.2.20, 7.4.3, 10.3.12, 11.7	(CF1 AND CF12):M (CF2 AND CF12):O	Yes, No

1 A.4.y QoS Enhanced Distributed Channel Access

Item	Protocol Capability	References	Status	Support
QD1	Support for four transmit queues with a separate channel access entity associated with each	9.1.3.1, 9.9.1.1	CF12:M	Yes, No
QD2	Per-channel access function differentiated channel access	9.9.1.2, 9.9.1.3, 9.9.1.5	CF12:M	Yes, No
QD3	Continuation of EDCA TXOP support	9.9.1.4	CF12:M	Yes, No
QD4	Maintenance of within-queue ordering, exhaustive retransmission when sending non-QoS data frames	9.9.1.6	CF12:M	Yes, No
QD5	Interpretation of ACM bit in EDCA Parameter Set	7.3.2.14, 9.9.3.1	(CF2 & CF12):M	Yes, No
QD6	Contention-based admission control	9.9.3.1, 7.2.3.15, 7.2.3.16, 7.4.2.1 – 7.4.2.3, 11.4	(CF1 & CF12):O (CF2 & CF12): O	Yes, No

2 A.4.z QoS HCF Controlled Channel Access

Item	Protocol Capability	References	Status	Support
QP1	TSPEC and associated frame formats	7.4.2.	CF12:M	Yes, No
QP2	HCF controlled channel access rules	9.1.3.2, 9.9.2, 9.9.2.1 - 9.9.2.3	CF12:M	Yes, No
QP3	HCF controlled channel access schedule generation and management	9.9.3	(CF1 & CF12):M	Yes, No
QP4	HCF frame exchange sequences	9.12	CF12:M	Yes, No
QP5	TS Management	11.4	CF12:M	Yes, No
QP6	Minimum TSPEC parameter set	9.9.3	CF12:M	Yes, No
QP7	Power management	11.2.1.10	CF12:M	Yes, No

1 **Annex C**

2 *Insert the following text at the end of the text portion of the text portions introducing Annex C.3 and*
3 *Annex C.4:*

4 This annex does not describe the behavior of a STA with QoS facility.

Annex D

In “Major Sections” of Annex D, insert the following text to the end of MAC attributes:

```
-- dot11EDCATable                ::= {dot11mac 4}
-- dot11QAPEDCATable             ::= {dot11mac 5}
-- dot11QosCountersTable         ::= {dot11mac 6}
```

In “SMT Station Config Table” of Annex D, insert the following text to the end of dot11StationConfigEntry sequence list:

```
dot11QosOptionImplemented        TruthValue,
dot11BlockAckOptionImplemented   TruthValue,
dot11DirectOptionImplemented     TruthValue,
dot11APSDOptionImplemented       TruthValue,
dot11QAckOptionImplemented       TruthValue,
dot11QBSSLoadOptionImplemented   TruthValue,
dot11QueueRequestOptionImpelemented TruthValue,
dot11TXOPRequestOptionImplemented TruthValue,
dot11MoreDataAckOptionImpelemented TruthValue,
dot11AssociateinQBSS            TruthValue
```

Insert the following elements to the end of dot11StationConfigEntry element definitions behind dot11CountryString:

```
dot11QosOptionImplemented OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "This attribute, when TRUE, indicates that the station
        implementation is capable of supporting QoS. The capability is
        disabled, otherwise. The default value of this attribute is FALSE."
    ::= { dot11StationConfigEntry 24 }

dot11BlockAckOptionImplemented OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "This attribute, when TRUE, indicates that the station
        implementation is capable of supporting Block Ack. The capability is
        disabled, otherwise. The default value of this attribute is FALSE."
    ::= { dot11StationConfigEntry 25 }

dot11DirectOptionImplemented OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "This attribute, when TRUE, indicates that the station
        implementation is capable of sending and receiving frames from a
        non-AP QSTA in the QBSS. The capability is disabled, otherwise. The
        default value of this attribute is FALSE."
    ::= { dot11StationConfigEntry 26 }

dot11APSDOptionImplemented OBJECT-TYPE
    SYNTAX TruthValue
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "This attribute is available only at a QAP. When TRUE, this
        attribute indicates that the QAP implementation is capable of
        delivering data and polls to stations using APSD."
    ::= { dot11StationConfigEntry 27 }
```

```

1
2 dot11QAckOptionImplemented OBJECT-TYPE
3     SYNTAX TruthValue
4     MAX-ACCESS read-only
5     STATUS current
6     DESCRIPTION
7         "This attribute, when TRUE, indicates that the station
8         implementation is capable of interpreting the CF-Ack bit in a
9         received frame of type data even if the frame is not directed to the
10        QoS station. The capability is disabled, otherwise. A QSTA is
11        capable of interpreting the CF-Ack bit in a received data frame if
12        that station is the recipient of the data frame, regardless of the
13        value of this MIB variable. The default value of this attribute is
14        FALSE."
15    ::= { dot11StationConfigEntry 28 }
16
17 dot11QBSSLoadOptionImplemented OBJECT-TYPE
18     SYNTAX TruthValue
19     MAX-ACCESS read-only
20     STATUS current
21     DESCRIPTION
22         "This attribute is available only at a QAP. This attribute, when
23         TRUE, indicates that the QAP implementation is capable of generating
24         and transmitting the QBSS load element in the Beacon and Probe
25         Response frames. The capability is disabled, otherwise. The default
26         value of this attribute is FALSE."
27    ::= { dot11StationConfigEntry 29 }
28
29 dot11QueueRequestOptionImplemented OBJECT-TYPE
30     SYNTAX TruthValue
31     MAX-ACCESS read-only
32     STATUS current
33     DESCRIPTION
34         "This attribute is available only at a QAP. This attribute, when
35         TRUE, indicates that the QAP is capable of processing the Queue Size
36         field in QoS Control field of QoS Data type frames. The capability
37         is disabled, otherwise. The default value of this attribute is
38         FALSE."
39    ::= { dot11StationConfigEntry 30 }
40
41 dot11TXOPRequestOptionImplemented OBJECT-TYPE
42     SYNTAX TruthValue
43     MAX-ACCESS read-only
44     STATUS current
45     DESCRIPTION
46         "This attribute is available only at a QAP. This attribute, when
47         TRUE, indicates that the QAP is capable of processing the TXOP
48         Duration requested field in QoS Control field of QoS Data type
49         frames. The capability is disabled, otherwise. The default value of
50         this attribute is FALSE."
51    ::= { dot11StationConfigEntry 31 }
52
53 dot11MoreDataAckOptionImplemented OBJECT-TYPE
54     SYNTAX TruthValue
55     MAX-ACCESS read-only
56     STATUS current
57     DESCRIPTION
58         "This attribute, when TRUE, indicates that the station
59         implementation is capable of interpreting the More Data bit in the
60         ACK frames. The capability is disabled, otherwise. The default value
61         of this attribute is FALSE."
62    ::= { dot11StationConfigEntry 32 }
63
64 dot11AssociateinQBSS OBJECT-TYPE
65     SYNTAX TruthValue
66     MAX-ACCESS read-write
67     STATUS current
68     DESCRIPTION
69         "This attribute, when TRUE, indicates that the station associates in
70         an nQBSS. When FALSE, this attribute indicates that the station

```

associates in a QBSS. The capability is disabled, otherwise. The default value of this attribute is TRUE."
 ::= { dot11StationConfigEntry 33 }

In “dot11OperationTable” of Annex D, insert the following text to the end of dot11OperationEntry sequence list:

dot11CAPLimit	INTEGER,
dot11HCCWmin	INTEGER,
dot11HCCWmax	INTEGER,
dot11HCAIFSN	INTEGER,
dot11ADDBAResponseTimeout	INTEGER,
dot11ADDTSResponseTimeout	INTEGER,
dot11ChannelUtilizationBeaconInterval	INTEGER,
dot11ScheduleTimeout	INTEGER,
dot11DLPResponseTimeout	INTEGER

Insert the following elements to the end of dot11OperationEntry element definitions behind dot11ProductID:

dot11CAPLimit OBJECT-TYPE
 SYNTAX INTEGER
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "This attribute shall specify the maximum number of TUs a Controlled access phase(CAP) can last."
 ::= { dot11OperationEntry 10 }

dot11HCCWmin OBJECT-TYPE
 SYNTAX INTEGER (0..aCWmin)
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "This attribute shall specify the value of the minimum size of the window that shall be used by the HC for generating a random number for the backoff. The value of this attribute shall be such that it could always be expressed in the form of $2^X - 1$, where X is an integer. The default value of this attribute shall be 0."
 ::= { dot11OperationEntry 11 }

dot11HCCWmax OBJECT-TYPE
 SYNTAX INTEGER (0..aCWmax)
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "This attribute shall specify the value of the maximum size of the window that shall be used by the HC for generating a random number for the backoff. The value of this attribute shall be such that it could always be expressed in the form of $2^X - 1$, where X is an integer. The default value of this attribute shall be 0."
 ::= { dot11OperationEntry 12 }

dot11HCAIFSN OBJECT-TYPE
 SYNTAX INTEGER (1..15)
 MAX-ACCESS read-write
 STATUS current
 DESCRIPTION
 "This attribute shall specify the number of slots, after a SIFS duration, that the HC shall sense the medium idle either before transmitting or executing a backoff. The default value of this attribute shall be 1."
 ::= { dot11OperationEntry 13 }

dot11ADDBAResponseTimeout OBJECT-TYPE
 SYNTAX INTEGER (1..65535)

```

1      MAX-ACCESS read-write
2      STATUS current
3      DESCRIPTION
4          "This attribute shall specify the maximum number of seconds a BA is
5          to be responded. The default value of this attribute shall be 1."
6      ::= { dot11OperationEntry 14 }
7
8      dot11ADDSResponseTimeout OBJECT-TYPE
9          SYNTAX INTEGER (1..65535)
10         MAX-ACCESS read-write
11         STATUS current
12         DESCRIPTION
13             "This attribute shall specify the maximum number of seconds an ADDTS
14             request is to be responded. The default value of this attribute
15             shall be 1."
16         ::= { dot11OperationEntry 15 }
17
18         dot11ChannelUtilizationBeaconInterval OBJECT-TYPE
19             SYNTAX INTEGER (0..100)
20             MAX-ACCESS read-write
21             STATUS current
22             DESCRIPTION
23                 "This attribute shall indicate the number of beacon intervals over
24                 which the channel busy time should be averaged. The default value
25                 for this parameter shall be 50."
26             ::= { dot11OperationEntry 16 }
27
28
29         dot11ScheduleTimeout OBJECT-TYPE
30             SYNTAX INTEGER (0..100)
31             MAX-ACCESS read-write
32             STATUS current
33             DESCRIPTION
34                 "This attribute shall indicate the duration in TUs after which a STA
35                 could go into power-save mode. The default value for this parameter
36                 shall be 10."
37             ::= { dot11OperationEntry 17 }
38
39         dot11DLPResponseTimeout OBJECT-TYPE
40             SYNTAX INTEGER (1..65535)
41             MAX-ACCESS read-write
42             STATUS current
43             DESCRIPTION
44                 "This attribute shall specify the maximum number of seconds a Direct
45                 Link request is to be responded. The default value of this attribute
46                 shall be 10."
47             ::= { dot11OperationEntry 18 }
48
49
50

```

In "dot11Counters Table" of Annex D, insert the following text to the end of Dot11CountersEntry sequence list:

dot11QoSdiscardedFragmentCount	Counter32,
dot11AssociatedStationCount	Counter32,
dot11QoSCFPollsReceivedCount	Counter32,
dot11QoSCFPollsUnusedCount	Counter32,
dot11QoSCFPollsUnusableCount	Counter32,

Insert the following elements to the end of Dot11CountersEntry element definitions behind dot11WEPUndecryptableCount:

```

61      dot11QoSdiscardedFragmentCount OBJECT-TYPE
62          SYNTAX Counter32
63          MAX-ACCESS read-only
64          STATUS current
65          DESCRIPTION
66
67              "This counter shall increment for each QoS Data MPDU that has been
68              discarded."

```

```

1
2      ::= { dot11CountersEntry 15 }
3
4  dot11AssociatedStationCount OBJECT-TYPE
5      SYNTAX Counter32
6      MAX-ACCESS read-only
7      STATUS current
8      DESCRIPTION
9
10         "This counter, only available at QAP, shall increment when a station
11         associates or reassociates. This counter shall decrement when a
12         station disassociates."
13
14      ::= { dot11CountersEntry 16 }
15
16  dot11QoSFCFPollsReceivedCount OBJECT-TYPE
17      SYNTAX Counter32
18      MAX-ACCESS read-only
19      STATUS current
20      DESCRIPTION
21
22         "This counter shall increment for each QoS (+)CF-Poll that has been
23         received."
24
25      ::= { dot11CountersEntry 17 }
26
27  dot11QoSFCFPollsUnusedCount OBJECT-TYPE
28      SYNTAX Counter32
29      MAX-ACCESS read-only
30      STATUS current
31      DESCRIPTION
32
33         "This counter shall increment for each QoS (+)CF-Poll that has been
34         received but not used."
35
36      ::= { dot11CountersEntry 18 }
37
38  dot11QoSFCFPollsUnusableCount OBJECT-TYPE
39      SYNTAX Counter32
40      MAX-ACCESS read-only
41      STATUS current
42      DESCRIPTION
43
44         "This counter shall increment for each QoS (+)CF-Poll that has been
45         received but could not be used due to the TXOP size being smaller
46         than the time that is required for one frame exchange sequence."
47
48      ::= { dot11CountersEntry 19 }
49

```

Insert the following changes to the 802.11 MIB in Annex D, after the Group Addresses Table:

```

51  -- *****
52  -- * SMT EDCA Config TABLE
53  -- *****
54
55  dot11EDCATable OBJECT-TYPE
56      SYNTAX SEQUENCE OF Dot11EDCATableEntry
57      MAX-ACCESS not-accessible
58      STATUS current
59      DESCRIPTION
60          "Conceptual table for EDCA default parameter values at a non-AP
61          QSTA. This table shall contain the four entries of the EDCA
62          parameters corresponding to four possible ACs. Index 1 corresponds
63          to AC_BK, index 2 to AC_BE, index 3 to AC_VI, and index 4 to
64          AC_VO."
65      REFERENCE
66          "IEEE 802.11 Tge Draft Version, 9..1.3.1"
67      ::= { dot11mac 4}
68

```

```

1 dot11EDCATableEntry OBJECT-TYPE
2   SYNTAX Dot11EDCATableEntry
3   MAX-ACCESS not-accessible
4   STATUS current
5   DESCRIPTION
6     "An Entry (conceptual row) in the EDCA Table.
7
8     ifIndex - Each 802.11 interface is represented by an ifEntry.
9     Interface tables in this MIB module are indexed by ifIndex."
10
11   INDEX {ifIndex, dot11EDCATableIndex}
12 ::= { dot11EDCATable 1 }
13
14 Dot11EDCATableEntry ::=
15   SEQUENCE {
16     dot11EDCATableIndex          INTEGER,
17     dot11EDCATableCWmin          INTEGER,
18     dot11EDCATableCWmax          INTEGER,
19     dot11EDCATableAIFSN          INTEGER,
20     dot11EDCATableTXOPLimit      INTEGER,
21     dot11EDCATableMSDULifetime   INTEGER,
22     dot11EDCATableMandatory      INTEGER
23   }
24
25 dot11EDCATableIndex OBJECT-TYPE
26   SYNTAX INTEGER (1..4)
27   MAX-ACCESS not-accessible
28   STATUS current
29   DESCRIPTION
30     "The auxiliary variable used to identify instances of the columnar
31     objects in the EDCA Table. The value of this variable is
32     1) 1, if the value of the AC is AC_BK.
33     2) 2, if the value of the AC is AC_BE.
34     3) 3, if the value of the AC is AC_VI.
35     4) 4, if the value of the AC is AC_VO."
36 ::= { dot11EDCATableEntry 1 }
37
38 dot11EDCATableCWmin OBJECT-TYPE
39   SYNTAX INTEGER (0..255)
40   MAX-ACCESS read-write
41   STATUS current
42   DESCRIPTION
43     "This attribute shall specify the value of the minimum size of the
44     window that shall be used by a QSTA for a particular AC for
45     generating a random number for the backoff. The value of this
46     attribute shall be such that it could always be expressed in the
47     form of  $2^x - 1$ , where X is an integer. The default value for this
48     attribute is
49     1) aCWmin, if dot11EDCATableIndex is 1 or 2.
50     2) (aCWmin+1)/2 - 1, if dot11EDCATableIndex is 3.
51     3) (aCWmin+1)/4 - 1, if dot11EDCATableIndex is 4."
52 ::= { dot11EDCATableEntry 2 }
53
54 dot11EDCATableCWmax OBJECT-TYPE
55   SYNTAX INTEGER (0..65535)
56   MAX-ACCESS read-write
57   STATUS current
58   DESCRIPTION
59     "This attribute shall specify the value of the maximum size of the
60     window that shall be used by a QSTA for a particular AC for
61     generating a random number for the backoff. The value of this
62     attribute shall be such that it could always be expressed in the
63     form of  $2^x - 1$ , where X is an integer. The default value for this
64     attribute is
65     1) aCWmax, if dot11EDCATableIndex is 1 or 2.
66     2) aCWmin, if dot11EDCATableIndex is 3.
67     3) (aCWmin+1)/2 - 1, if dot11EDCATableIndex is 4."
68 ::= { dot11EDCATableEntry 3 }
69
70
71

```

```

1 dot11EDCATableAIFSN OBJECT-TYPE
2   SYNTAX INTEGER (2..15)
3   MAX-ACCESS read-write
4   STATUS current
5   DESCRIPTION
6     "This attribute shall specify the number of slots, after a SIFS
7     duration, that the QSTA, for a particular AC, shall sense the medium
8     idle either before transmitting or executing a backoff. The default
9     value for this attribute is
10    1) 7, if dot11EDCATableIndex is 1,
11    2) 3, if dot11EDCATableIndex is 2
12    3) 2, otherwise."
13
14 ::= {dot11EDCATableEntry 4 }
15
16 dot11EDCATableTXOPLimit OBJECT-TYPE
17   SYNTAX INTEGER (0..65535)
18   MAX-ACCESS read-write
19   STATUS current
20   DESCRIPTION
21     "This attribute shall specify the maximum number of microseconds of
22     an EDCA TXOP for a given AC at a non-AP QSTA. The default value for
23     this attribute is
24     1) 0 for all PHYs, if dot11EDCATableIndex is 1 or 2; this implies
25        that the sender can send one MSDU in an EDCA TXOP,
26     2) 3008 microseconds for IEEE 802.11a/g PHY and 6016 microseconds
27        for IEEE 802.11b PHY, if dot11EDCATableIndex is 3,
28     3) 1504 microseconds for IEEE 802.11a/g PHY and 3008 microseconds
29        for IEEE 802.11b PHY, if dot11EDCATableIndex is 4."
30
31 ::= {dot11EDCATableEntry 5 }
32
33
34 dot11EDCATableMSDULifetime OBJECT-TYPE
35   SYNTAX INTEGER (0..500)
36   MAX-ACCESS read-write
37   STATUS current
38   DESCRIPTION
39     "This attribute shall specify (in TUs) the maximum duration an MSDU,
40     for a given AC, would be retained by the MAC before it is discarded.
41     The default value for this parameter shall be 500."
42
43 ::= {dot11EDCATableEntry 6 }
44
45 dot11EDCATableMandatory OBJECT-TYPE
46   SYNTAX TruthValue
47   MAX-ACCESS read-write
48   STATUS current
49   DESCRIPTION
50     "This attribute, when TRUE, indicates that admission control is
51     mandatory for the given AC. When False, this attribute indicates
52     that the admission control is not mandatory for the given AC. The
53     default value for this parameter shall be FALSE."
54
55 ::= {dot11EDCAMandatory 77 }
56
57 -- *****
58 -- * End of dot11EDCA TABLE
59 -- *****
60
61 -- *****
62 -- * SMT QAP EDCA Config TABLE
63 -- *****
64
65 dot11QAPEDCATable OBJECT-TYPE
66   SYNTAX SEQUENCE OF Dot11QAPEDCATableEntry
67   MAX-ACCESS not-accessible
68   STATUS current
69   DESCRIPTION
70     "Conceptual table for EDCA default parameter values at the QAP. This
71     table shall contain the four entries of the EDCA parameters

```

```

1      corresponding to four possible ACs.  Index 1 corresponds to AC_BK,
2      index 2 to AC_BE, index 3 to AC_VI, and index 4 to AC_VO."
3  REFERENCE
4      "IEEE 802.11 Tge Draft Version, 9.10.1.X (to be filled later)"
5  ::= { dot11mac 5 }
6
7  dot11QAPEDCatableEntry OBJECT-TYPE
8      SYNTAX Dot11QAPEDCatableEntry
9      MAX-ACCESS not-accessible
10     STATUS current
11     DESCRIPTION
12         "An Entry (conceptual row) in the EDCA Table.
13
14         ifIndex - Each 802.11 interface is represented by an ifEntry.
15         Interface tables in this MIB module are indexed by ifIndex."
16
17     INDEX {ifIndex, dot11QAPEDCatableIndex}
18     ::= { dot11QAPEDCatable 1 }
19
20 Dot11QAPEDCatableEntry ::=
21     SEQUENCE {
22         dot11QAPEDCatableIndex          INTEGER,
23         dot11QAPEDCatableCWmin          INTEGER,
24         dot11QAPEDCatableCWmax          INTEGER,
25         dot11QAPEDCatableAIFSN          INTEGER,
26         dot11QAPEDCatableTXOPLimit      INTEGER,
27         dot11QAPEDCatableMSDULifetime   INTEGER,
28         dot11QAPEDCatableMandatory      INTEGER
29     }
30
31 dot11QAPEDCatableIndex OBJECT-TYPE
32     SYNTAX INTEGER (1..4)
33     MAX-ACCESS not-accessible
34     STATUS current
35     DESCRIPTION
36         "The auxiliary variable used to identify instances of the columnar
37         objects in the EDCA Table. The value of this variable is
38         1) 1, if the value of the AC is AC_BK.
39         2) 2, if the value of the AC is AC_BE.
40         3) 3, if the value of the AC is AC_VI.
41         4) 4, if the value of the AC is AC_VO.."
42     ::= { dot11QAPEDCatableEntry 1 }
43
44 dot11QAPEDCatableCWmin OBJECT-TYPE
45     SYNTAX INTEGER (0..255)
46     MAX-ACCESS read-write
47     STATUS current
48     DESCRIPTION
49         "This attribute shall specify the value of the minimum size of the
50         window that shall be used by a QAP for a particular AC for
51         generating a random number for the backoff. The value of this
52         attribute shall be such that it could always be expressed in the
53         form of  $2^x - 1$ , where X is an integer. The default value for this
54         attribute is
55         1) aCWmin, if dot11QAPEDCatableIndex is 1 or 2.
56         2) (aCWmin+1)/2 - 1, if dot11QAPEDCatableIndex is 3.
57         3) (aCWmin+1)/4 - 1, if dot11QAPEDCatableIndex is 4."
58
59     ::= { dot11QAPEDCatableEntry 2 }
60
61 dot11QAPEDCatableCWmax OBJECT-TYPE
62     SYNTAX INTEGER (0..65535)
63     MAX-ACCESS read-write
64     STATUS current
65     DESCRIPTION
66         "This attribute shall specify the value of the maximum size of the
67         window that shall be used by a QAP for a particular AC for
68         generating a random number for the backoff. The value of this
69         attribute shall be such that it could always be expressed in the
70         form of  $2^x - 1$ , where X is an integer. The default value for this
71         attribute is

```

```

1      1) aCWmax, if dot11QAPEDCATableIndex is 1.
2      2) 4*(aCWmin+1) - 1, if dot11QAPEDCATableIndex is 2.
3      3) aCWmin, if dot11QAPEDCATableIndex is 3.
4      4) (aCWmin+1)/2 - 1, if dot11QAPEDCATableIndex is 4."
5
6      ::= {dot11QAPEDCATableEntry 3 }
7
8  dot11QAPEDCATableAIFSN OBJECT-TYPE
9      SYNTAX INTEGER (1..15)
10     MAX-ACCESS read-write
11     STATUS current
12     DESCRIPTION
13         "This attribute shall specify the number of slots, after a SIFS
14         duration, that the QAP, for a particular AC, shall sense the medium
15         idle either before transmitting or executing a backoff. The default
16         value for this attribute is
17         1) 7, if dot11QAPEDCATableIndex is 1
18         2) 3, if dot11QAPEDCATableIndex is 2,
19         3) 1, otherwise."
20
21     ::= {dot11QAPEDCATableEntry 4 }
22
23  dot11QAPEDCATableTXOPLimit OBJECT-TYPE
24     SYNTAX INTEGER (0..65535)
25     MAX-ACCESS read-write
26     STATUS current
27     DESCRIPTION
28         "This attribute shall specify the maximum number of microseconds of
29         an EDCA TXOP for a given AC at the QAP. The default value for this
30         attribute is
31         1) 0 for all PHYs, if dot11QAPEDCATableIndex is 1 or 2; this implies
32            that the sender can send one MSDU in an EDCA TXOP,
33         2) 3000 microseconds for IEEE 802.11a/g PHY and 6000 microseconds
34            for IEEE 802.11b PHY, if dot11QAPEDCATableIndex is 3,
35         3) 1500 microseconds for IEEE 802.11a/g PHY and 3000 microseconds
36            for IEEE 802.11b PHY, if dot11QAPEDCATableIndex is 4."
37
38     ::= {dot11QAPEDCATableEntry 5 }
39
40
41  dot11QAPEDCATableMSDULifetime OBJECT-TYPE
42     SYNTAX INTEGER (0..500)
43     MAX-ACCESS read-write
44     STATUS current
45     DESCRIPTION
46         "This attribute shall specify (in TUs) the maximum duration an MSDU,
47         for a given AC, would be retained by the MAC at the QAP before it is
48         discarded. The default value for this parameter shall be 500.
49
50     ::= {dot11QAPEDCATableEntry 6 }
51
52  dot11QAPEDCATableMandatory OBJECT-TYPE
53     SYNTAX TruthValue
54     MAX-ACCESS read-write
55     STATUS current
56     DESCRIPTION
57         "This attribute, when TRUE, indicates that admission control is
58         mandatory for the given AC. When False, this attribute indicates
59         that the admission control is not mandatory for the given AC. The
60         default value for this parameter shall be FALSE.
61
62     ::= {dot11QAPEDCAMandatory 7 }
63
64
65  -- *****
66  -- * End of dot11QAPEDCA TABLE
67  -- *****
68
69  -- *****
70  -- * dot11QosCounters TABLE
71  -- *****

```

```

1
2 dot11QosCountersTable OBJECT-TYPE
3     SYNTAX SEQUENCE OF Dot11QosCountersEntry
4     MAX-ACCESS not-accessible
5     STATUS current
6     DESCRIPTION
7
8         "Group containing attributes that are MAC counters implemented as a
9         table to allow for multiple instantiations on an agent."
10
11     ::= { dot11mac 6 }
12
13 dot11QosCountersEntry OBJECT-TYPE
14     SYNTAX Dot11QosCountersEntry
15     MAX-ACCESS not-accessible
16     STATUS current
17     DESCRIPTION
18         "An Entry (conceptual row) in the QoS Table.
19
20         ifIndex - Each 802.11 interface is represented by an ifEntry.
21         Interface tables in this MIB module are indexed by ifIndex."
22
23     INDEX {ifIndex, dot11QosCounterIndex}
24     ::= { dot11QosCountersTable 1 }
25
26 Dot11QosCountersEntry ::=
27     SEQUENCE {
28         dot11QosCounterIndex                INTEGER,
29         dot11QosTransmittedFragmentCount    Counter32,
30         dot11QosFailedCount                  Counter32,
31         dot11QosRetryCount                   Counter32,
32         dot11QosMultipleRetryCount           Counter32,
33         dot11QosFrameDuplicateCount          Counter32,
34         dot11QosRTSSuccessCount               Counter32,
35         dot11QosRTSFailureCount               Counter32,
36         dot11QosACKFailureCount               Counter32,
37         dot11QosReceivedFragmentCount        Counter32,
38         dot11QosTransmittedFrameCount        Counter32,
39         dot11QosDiscardedFrameCount          Counter32,
40         dot11QosMPDUsReceivedCount           Counter32,
41         dot11QosRetriesReceivedCount         Counter32 }
42
43 dot11QosCounterIndex OBJECT-TYPE
44     SYNTAX INTEGER (1..16)
45     MAX-ACCESS not-accessible
46     STATUS current
47     DESCRIPTION
48
49         "The auxiliary variable used to identify instances of the columnar
50         objects in the QOSCounter Table. The value of this variable is equal
51         to TID + 1."
52
53     ::= {dot11QosCountersEntry 1 }
54
55 dot11QosTransmittedFragmentCount OBJECT-TYPE
56     SYNTAX Counter32
57     MAX-ACCESS read-only
58     STATUS current
59     DESCRIPTION
60
61         "This counter shall be incremented for an acknowledged MPDU, for a
62         particular UP, with an individual address in the address 1 field or
63         an MPDU with a multicast address in the address 1 field, either
64         belonging to a particular TID. This counter has relevance only for
65         TIDs between 0 and 7."
66
67     ::= { dot11QosCountersEntry 2 }
68
69 dot11QosFailedCount OBJECT-TYPE
70     SYNTAX Counter32
71

```

```

1      MAX-ACCESS read-only
2      STATUS current
3      DESCRIPTION
4
5          "This counter shall increment when an MSDU, for a particular UP, is
6          not transmitted successfully due to the number of transmit attempts
7          exceeding either the dot11ShortRetryLimit or dot11LongRetryLimit.
8          This counter has relevance only for TIDs between 0 and 7."
9
10     ::= { dot11QosCountersEntry 3 }
11
12 dot11QoSsRetryCountTC[0..7] OBJECT-TYPE
13     SYNTAX Counter32
14     MAX-ACCESS read-only
15     STATUS current
16     DESCRIPTION
17
18         "This counter shall increment when an MSDU, of a particular UP, is
19         successfully transmitted after one or more retransmissions. This
20         counter has relevance only for TIDs between 0 and 7."
21
22     ::= { dot11QosCountersEntry 4 }
23
24 dot11QoSMultipleRetryCount OBJECT-TYPE
25     SYNTAX Counter32
26     MAX-ACCESS read-only
27     STATUS current
28     DESCRIPTION
29
30         "This counter shall increment when an MSDU, of a particular UP, is
31         successfully transmitted after more than one retransmission. This
32         counter has relevance only for TIDs between 0 and 7."
33
34     ::= { dot11QosCountersEntry 5 }
35
36 dot11QoSFrameDuplicateCount OBJECT-TYPE
37     SYNTAX Counter32
38     MAX-ACCESS read-only
39     STATUS current
40     DESCRIPTION
41
42         "This counter shall increment when a frame, of a particular UP, is
43         received that the Sequence Control field indicates is a duplicate.
44         This counter has relevance only for TIDs between 0 and 7."
45
46     ::= { dot11QosCountersEntry 6 }
47
48 dot11QoSRTSSuccessCount OBJECT-TYPE
49     SYNTAX Counter32
50     MAX-ACCESS read-only
51     STATUS current
52     DESCRIPTION
53
54         "This counter shall increment when a CTS is received in response to
55         an RTS that has been sent for the transmission of an MPDU of a
56         particular UP. This counter has relevance only for TIDs between 0
57         and 7."
58
59     ::= { dot11QosCountersEntry 7 }
60
61 dot11QoSRTSFailureCount OBJECT-TYPE
62     SYNTAX Counter32
63     MAX-ACCESS read-only
64     STATUS current
65     DESCRIPTION
66
67         "This counter shall increment when a CTS is not received in response
68         to an RTS that has been sent for the transmission of an MPDU of a
69         particular UP. This counter has relevance only for TIDs between 0
70         and 7."
71

```

```

1      ::= { dot11QosCountersEntry 8 }
2
3      dot11QosACKFailureCount OBJECT-TYPE
4          SYNTAX Counter32
5          MAX-ACCESS read-only
6          STATUS current
7          DESCRIPTION
8
9              "This counter shall increment when an ACK is not received in
10             response to an MPDU of a particular UP. This counter has relevance
11             only for TIDs between 0 and 7."
12
13      ::= { dot11QosCountersEntry 9 }
14
15      dot11QosReceivedFragmentCount OBJECT-TYPE
16          SYNTAX Counter32
17          MAX-ACCESS read-only
18          STATUS current
19          DESCRIPTION
20
21              "This counter shall be incremented for each successfully received
22              MPDU of type Data of a particular UP. This counter has relevance
23              only for TIDs between 0 and 7."
24
25      ::= { dot11QosCountersEntry 10 }
26
27      dot11QosTransmittedFrameCount OBJECT-TYPE
28          SYNTAX Counter32
29          MAX-ACCESS read-only
30          STATUS current
31          DESCRIPTION
32
33              "This counter shall increment for each successfully transmitted MSDU
34              of a particular UP. This counter has relevance only for TIDs between
35              0 and 7."
36
37      ::= { dot11QosCountersEntry 11 }
38
39      dot11QosDiscardedFrameCount OBJECT-TYPE
40          SYNTAX Counter32
41          MAX-ACCESS read-only
42          STATUS current
43          DESCRIPTION
44
45              "This counter shall increment for each Discarded MSDU of a
46              particular UP. This counter has relevance only for TIDs between 0
47              and 7."
48
49      ::= { dot11QosCountersEntry 12 }
50
51      dot11QosMPDUsReceivedCount OBJECT-TYPE
52          SYNTAX Counter32
53          MAX-ACCESS read-only
54          STATUS current
55          DESCRIPTION
56
57              "This counter shall increment for each received MPDU of a particular
58              TID."
59
60      ::= { dot11QosCountersEntry 13 }
61
62      dot11QosRetriesReceivedCount OBJECT-TYPE
63          SYNTAX Counter32
64          MAX-ACCESS read-only
65          STATUS current
66          DESCRIPTION
67
68              "This counter shall increment for each received MPDU of a particular
69              TID with the retry bit set to 1."
70
71      ::= { dot11QosCountersEntry 14 }

```

1
2
3
4
5

```
-- *****  
-- * End of dot11QosCounters TABLE  
-- *****
```

Annex E

(Informative)

Bibliography

E.1 General

Insert the following references into E.1 and renumber references as appropriate:

[B5] IEEE Std 802.1Q-1998, IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks.

[B6] IETF RFC 2215, General Characterization Parameters for Integrated Service Network Elements. S. Shenker, J. Wroclawski. September 1997.

[B7] IETF RFC 3290, An Informal Management Model for Diffserv Routers, Y. Bernet, S. Blake, D. Grossman and A. Smith, May 2002.

[B8] IETF RFC 2212, Specification of the Guaranteed Quality of Service, S. Shenker, C. Partridge and R. Guerin, September 1997.

[B9] IETF RFC 2474, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, K. Nichols, S. Blake, F. Baker, and D. Black, December 1998.

Annex H

(Informative)

H.1 Example use of TSPEC for admission control

Admission control, in general, depends on vendors' implementations of schedulers, available channel capacity, link conditions, retransmission limits, and the scheduling requirements of a given TSPEC. However, for any given channel capacity, link conditions, and retransmission limits, some TSPEC constructions might be categorically rejected because a scheduler cannot create a meaningful schedule for that TSPEC. There must, for example, be a minimum number of specified fields in the TSPEC in order for the admission control mechanism to create a valid TSPEC. Table H.1 below lists the valid TSPEC parameters that must be present for all admission control algorithms to admit a TSPEC. This represents a set of necessary parameters in order for TSPEC to be admitted – it is not sufficient in and of itself to guarantee TSPEC admittance, which depends upon channel conditions and other factors. Such TSPECs are said to be Admissible. In the Table S means specified, and X means unspecified, and DC means “Don't Care”.

Table H.1 – Admissible TSPECs

TSPEC Parameter	Continuous Time QoS Traffic (HCCA)	Controlled-access CBR Traffic (HCCA)	Bursty Traffic (HCCA)	Unspecified non-QoS Traffic (HCCA)	Contention Based CBR Traffic (EDCA)
Nominal MSDU Size	S	S	X	DC	S
Minimum Service Interval	S	Nominal MSDU size/ Mean Data Rate, if specified (VoIP typically uses this)	Mean Data Rate/Nominal MSDU size, if Mean Data Rate specified	DC	DC
Maximum Service Interval	S	Delay Bound/Number of retries (AV typically uses this)	Delay Bound/Number of retries, if Delay Bound present	DC	DC
Inactivity Interval	Always Specified				DC
Suspension Interval	DC				
Minimum Data Rate	Must be specified if Peak Data Rate is specified	Equal to Mean Data Rate	X	DC	DC
Mean Data Rate	S	S	DC	DC	S
Maximum Burst Size	X	X	S	DC	DC
Minimum PHY Rate	Always specified				
Peak Data Rate	Must be specified if Minimum Data Rate Specified DC	Equal to Mean Data Rate	DC	DC	DC
Delay Bound	S	S	DC	X	X

TSPEC Parameter	Continuous Time QoS Traffic (HCCA)	Controlled-access CBR Traffic (HCCA)	Bursty Traffic (HCCA)	Unspecified non-QoS Traffic (HCCA)	Contention Based CBR Traffic (EDCA)
Surplus Bandwidth Allowance	Must be specified if delay and jitter bounds present			DC	S
Medium Time	X (not specified by non-AP QSTA; only an output from the HC)				

H.2 Recommended Practices for Contention-based Admission Control

H.2.1 Use of Admission Control Mandatory (ACM) Subfield

It is recommended not to require Admission Control for the access categories AC_BE and AC_BK. The ACM subfield for these categories should be set to 0. The AC parameters chosen by the QAP should account for unadmitted traffic in these access categories.

H.2.2 Deriving Medium Time

It is recommended that the QAP use the following procedure to derive Medium Time in its ADDTS response.

There are two requirements to consider: 1) the traffic requirements of the application, and 2) the expected error performance of the medium. The application requirements are captured by two TSPEC parameters: Nominal MSDU Size and Mean Data Rate. The medium requirements are captured by two TSPEC parameters: Surplus Bandwidth Allowance and Minimum PHY Rate. The following formula describes how Medium Time may be calculated (assuming RTS/CTS protection is not used):

$$\text{Medium Time} = \text{Surplus Bandwidth Allowance} * \text{pps} * \text{MPDUEXchangeTime}$$

where:

$$\text{pps} = \text{ceiling}((\text{Mean Data Rate} / 8) / \text{Nominal MSDU Size});$$

$$\text{MPDUEXchangeTime} = \text{duration}(\text{Nominal MSDU Size}, \text{Minimum PHY Rate}) + \text{SIFS} + \text{ACK duration}$$

(also see the definition of MPDUEXchangeTime in 9.9.3.1.2;

duration() is the PLME-TXTIME primitive that returns the duration of a packet based on its payload size and the PHY data rate employed.

H.3 Guidelines and reference design for example scheduler and admission control unit

H.3.1 Guidelines for Deriving Service Schedule Parameters

The HC establishes the service interval for each admitted TS for a non-AP QSTA to derive the aggregate Minimum Service Interval contained in the non-AP QSTA's Service Schedule. The service interval for each TS is equal to a non-zero Minimum Service Interval contained in the TSPEC, if it exists; otherwise, it is the Nominal MSDU Size divided by the Mean Data Rate. The Service Interval contained in the Service Schedule is equal to the smallest service interval for any TSPEC.

The HC may use an aggregate "token bucket specification" to police a non-AP QSTA's admitted flows. The HC must derive the aggregate mean data rate and aggregate maximum burst size to establish the aggregate token bucket specification. The aggregate mean data rate is equal to the sum of the mean data rates of all of the non-AP QSTA's admitted TS. The aggregate maximum burst size is equal to the sum of the maximum burst size of all of the non-AP QSTA's admitted TS. An aggregate token bucket is initialized with the aggregate maximum burst size. Tokens are added to the token bucket at the aggregate mean data rate.

The Maximum Service Duration contained in a non-AP QSTA's Service Schedule is bounded by the aggregated Maximum Burst Size.

H.3.2 TSPEC Construction (informative)

TSPECs are constructed at the SME, from application requirements supplied via the SME, and with information specific to the MAC layer. There are no normative requirements on how any TSPEC is to be generated. However, in this subclause a description is given of how and where certain parameters may be chosen. The following parameters typically arise from the application: Nominal MSDU Size, Maximum MSDU Size, Minimum Service Interval, Maximum Service Interval, Inactivity Interval, Minimum Data Rate, Mean Data Rate, Maximum Burst Size, Peak Data Rate, and Delay Bound. The following parameters are generated locally within the MAC: Minimum PHY Rate and Surplus Bandwidth Allowance, although the Maximum Service Interval and Minimum Service Intervals may be generated within the MLME as well. This subclause describes how the parameters that are typically generated within the MAC may be derived.

Note that a TSPEC may also be generated autonomously by the MAC without any initiation by the station management entity. However, if a TSPEC is generated subsequently by the SME, the TSPEC generated autonomously by the MAC shall be overridden. If one or more TSPECs are initiated by the SME, the autonomous TSPEC, containing the same TSID, shall be terminated.

Typically, TSPEC parameters not determined by the application are built upon the assumptions that there exist:

- A probability p of not transmitting the frame (because it would have exceeded its Delay Bound)
- An MSDU length (which can be considered fixed for constant-bit-rate applications)
- Application throughput and delay requirements
- A channel model of error – in particular a channel error probability for the (fixed) frame length
- Possibly country-specific limits on TXOP limits

The Minimum Service Interval, if determined within the MAC, may typically be given as the Nominal MSDU size/Mean Data Rate.

The Maximum Service Interval, if determined within the MAC may be calculated as the Delay Bound/Number of Retries possible. This number should be greater than the Minimum Service Interval, when that is specified. The number of retries may be chosen (as below) to meet a particular probability of dropping a packet because it exceeds its Delay Bound. Note that for multiple streams, this Service Interval should be the aggregate of all service intervals requested, since the QSTA is assigned the TXOPs, not any particular stream.

Typically, it can be assumed that the scheduler would attempt to schedule TXOPs distributed throughout a small multiple of beacon intervals (if not a single beacon interval). In addition, TXOP limits would typically be chosen to be as short as possible (within the constraints of the minimum PHY rate, Ack policy, and so forth), consistent with the goal of maximizing throughput. In other words, because of overhead, not to mention the requirements for transmitting a single Poll, MPDU and possibly ACK, the TXOPs need to be at least of a certain duration.

The channel model implies an error rate and an assumption about dependency (joint probability distribution of channel errors sequentially – i.e. burst error probabilities).

For example, if the channel causes errors independently from frame to frame, and the error probability is the same for all frames of the same length at all times, this channel would be said to be an independent, identically distributed error channel. With p as the probability of dropping the frame, and p_e as the probability of the frame not being transmitted successfully (i.e., either the data frame or the ACK associated with it are in error), let N_p be the number of retries required to maintain the probability of dropping the frame to be p .

The probability of any given packet being dropped in such a channel after N_p retries is given by:

$$p_{drop} = (p_e)^{N_p+1}$$

For example, in such a channel, if $p_e = 0.1$, and $p_{drop} = 10^{-8}$ then up to 7 retries within the delay bound are required, and the scheduler should ensure that sufficient cumulative TXOP allocations are made to accommodate retransmissions.

The Surplus Bandwidth Allowance parameter ensures the requesting QSTA is allocated a minimum amount of excess time by the scheduler so that application dropped packet rates are bounded. For example, this parameter can be chosen to ensure that when there is a 10% PER for 1000 octet packets, that there is a dropped packet error rate (i.e., packets that fail to be received within the Delay Bound) of 10^{-8} . This parameter may be chosen based on an initial assumption regarding channel/source characteristics, and renegotiated by sending ADDTS request if required, based on actual transmission behavior. To understand how this parameter may be specified, consider the case, where there are only 100 PPDU's to be transmitted, and that delay is not an issue. The packet error rate, p_e is 10%, with the errors happening independently from packet to packet. To accomplish this, the number of packets transmitted in each beacon interval must exceed the 100 PPDU's by N_{excess} in order to avoid dropping packets with some fixed probability (denoted as p_{drop}). For example, if $p_{drop} = 10^{-8}$, then the number of retries N_{excess} must satisfy to send *only* 100 packets successfully (based on Bernoulli distributed error probabilities):

$$p_{drop} = \sum_{k=N_{excess}}^{N_{excess}+100} \binom{N_{excess}+100}{k} p_e^k (1-p_e)^{100+N_{excess}-k}$$

where $p_e = 0.1$. Solution of an equation such as this yields the total number of additional retries. This may be found, for this example using the fact that

$$\sum_{k=a}^b \binom{b}{k} p_e^k (1-p_e)^{b-k} = I_{p_e}(a, b-a+1)$$

where $I_{p_e}(a, b)$ is the Incomplete Beta function, and taking n sufficiently large (or invoking the law of large numbers). Solving this yields that, on the average, 38 additional MPDU's are required to keep the probability of dropping a packet to less than 10^{-8} to send *only* 100 packets. For this case the

Surplus Bandwidth Allowance = $\frac{100+N_{excess}}{100}$, which for this example would be 1.38.

This might represent an upper bound for the excess bandwidth for many applications: it presumes that the observation interval is $100 + N_{excess}$ frames. When the observation interval (or Delay Bound) is longer than the time it takes to transmit 100 frames, then it can be shown that the excess bandwidth required decreases. For example, if it were desired to send 100000 frames with 12000 frames excess, then the probability of a dropped frame becomes 1.6×10^{-15} .

On the other hand, suppose that an infinitely long stream was to be transmitted without any constraints on delay. In such a case, with an infinite number of retries, and with a 10% PER, 1.111 times the bandwidth required to send MPDU's without error are required (because the probability that n retries are required for any packet is given by $.1^n$).

In fact, assuming a finite delay bound, the above result ($1.111 \times$ Surplus Bandwidth Allowance) represents a lower bound on what the Surplus Bandwidth Allowance would be.

Typically, then the excess bandwidth required would be between a “send only $N +$ excess packets” scenario within a given delay, and “send an infinite number of packets with an infinite number of retries with no delay” scenario.

A more exact calculation can be done via simulation as follows: Suppose there are $N_{\text{allocated}}$ constant length PPDUs per beacon interval (these are actually transmitted on the air), and suppose there are N_{payload} constant length PPDUs to be transmitted. Suppose further that these transmitted and payload PPDUs arrive in a uniformly distributed manner in each beacon interval, then the Delay incurred in waiting for a packet to be transmitted may be inferred from examining the transmit queue length, and statistically may be inferred from examining how many retries within a certain period of time are required to keep the probability of a dropped packet below a certain amount. The number of retries (and consideration of interframe space, polls, etc.) then determines the Surplus Bandwidth allowance.

In general, then the Surplus Bandwidth Allowance may be given by: $\frac{N_{\text{allocated}}}{N_{\text{payload}}}$.

Note that for the case of a Block Acknowledgement a similar calculation applies, although the calculations for the excess bandwidth need to take into account the probability of failing to receive a Block Ack, and so forth.

H.3.3 Reference Design for an example Scheduler and Admission Control Unit

This subclause provides the guidelines, for the design of a simple scheduler and admission control unit (the unit that administers admission policy in the HC SME), that meet the minimum performance requirements as specified in clause 9.9.3.2. The scheduler and admission control unit use the minimum set of mandatory TSPEC parameters as specified in subclause 9.9.3.2.

H.3.3.1 An example scheduler

This subclause includes the reference design for an example Scheduler. This Scheduler uses the mandatory set of TSPEC parameters to generate a schedule: Mean Data Rate, Nominal MSDU Size and Maximum Service Interval or Delay Bound. If both Maximum Service Interval and Delay Bound are specified by the non-AP QSTA in the TSPEC, the Scheduler uses the Maximum Service Interval for the calculation of the schedule.

The schedule generated by the Scheduler meets the normative behavior as specified in clause 9.9.3.2. The schedule for an admitted stream is calculated in two steps. The first step is the calculation of the Scheduled Service Interval (SI). In the second step, the TXOP duration for a given SI is calculated for the stream.

The calculation of the Scheduled Service Interval is done as follows. First the scheduler calculates the minimum of all Maximum Service Intervals for all admitted streams. Let this minimum be “m”. Second, the scheduler chooses a number lower than “m” that is a submultiple of the beacon interval. This value is the Scheduled Service Interval for all non-AP QSTAs with admitted streams. See Figure H-1.

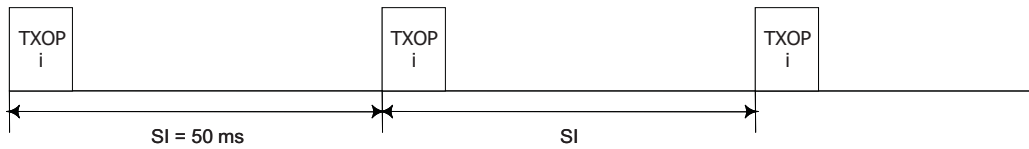
For the calculation of the TXOP duration for an admitted stream, the Scheduler uses the following parameters: Mean Data Rate (ρ) and Nominal MSDU Size (L) from the negotiated TSPEC, the Scheduled Service Interval (SI) calculated above, Physical Transmission Rate (R), Maximum allowable Size of MSDU, i.e., 2304 bytes (M) and Overheads in time units (O). The Physical Transmission Rate is the Minimum PHY Rate negotiated in the TSPEC. If Minimum PHY Rate is not committed in ADDTS response, the scheduler can use observed PHY rate as R . The Overheads in time includes interframe spaces, ACKs and CF-Polls. For simplicity, details for the overhead calculations are omitted in this description. The TXOP duration is calculated as follows. First the scheduler calculates the number of MSDUs that arrived at the Mean Data Rate during the SI:

$$N_i = \left\lceil \frac{SI \times \rho_i}{L_i} \right\rceil$$

- 1 Then the scheduler calculates the TXOP duration as the maximum of (1) time to transmit N_i frames at R_i and
 2 (2) time to transmit one maximum size MSDU at R_i (plus overheads):

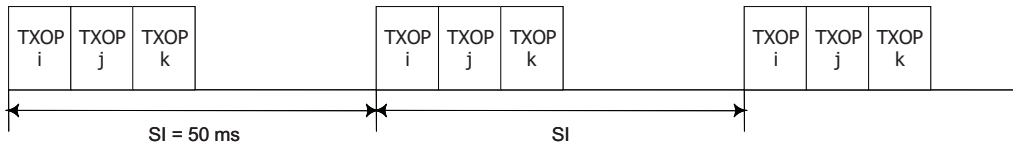
$$TXOP_i = \max \left(\frac{N_i \times L_i}{R_i} + O, \frac{M}{R_i} + O \right)$$

- 3 An example is shown in figure 62.5. Stream from QSTA “i” is admitted. The beacon interval is 100 ms and
 4 the Maximum Service Interval for the stream is 60 ms. The scheduler calculates a Scheduled Service Interval
 5 (SI) equal to 50 ms using the steps explained above.



6
 7 **Figure H-1 – Schedule for stream from QSTA “i”**

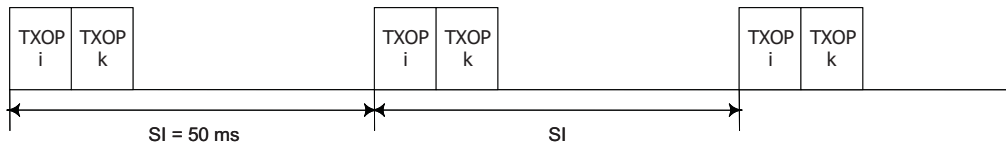
- 8 The same process is repeated continuously while the Maximum Service Interval for the admitted stream is
 9 larger than current SI. An example is shown in Figure H-2.



10
 11 **Figure H-2 – Schedule for streams from QSTAs “i” to “k”**

- 12 If a new stream is admitted with a Maximum Service Interval smaller than the current SI, the scheduler needs
 13 to change the current SI to a smaller number than the Maximum Service Interval of the newly admitted
 14 stream. Therefore the TXOP duration for the current admitted streams needs also to be recalculated with the
 15 new SI.

- 16 If a stream is dropped, the scheduler might use the time available to resume contention. The scheduler might
 17 also choose to move the TXOPs for the QSTAs following the QSTA dropped to use the unused time. An
 18 example is shown in figure H-3, when stream for QSTA j is removed. However, this option might require the
 19 announcement of a new schedule to all QSTAs.



20
 21 **Figure H-3 – Reallocation of TXOPs when a stream is dropped**

1 Different modifications can be implemented to improve the performance of the minimum scheduler. For
 2 example, a scheduler might generate different Scheduled Service Intervals (SI) for different QSTAs, and/or a
 3 scheduler might consider accommodating retransmissions while allocating TXOP durations.

4 **H.3.3.2 An Admission Control Unit**

5 This subclause describes a reference design for an Admission Control Unit (ACU) that administers admission
 6 of TS. The ACU uses the same set of parameters that the scheduler uses in H.3.3.1.

7 When a new stream requests admission, the admission control process is done in three steps. First, the ACU
 8 calculates the number of MSDUs that arrive at the Mean Data Rate during the Scheduled Service Interval.
 9 The Scheduled Service Interval (SI) is the one that the scheduler calculates for the stream as specified in
 10 clause H.3.3.1. For the calculation of number of MSDUs the ACU uses the equation for N_i shown in H.3.3.1.
 11 Second, the ACU calculates the TXOP duration that needs to be allocated for the stream. The ACU uses the
 12 equation for $TXOP_i$ shown in H.3.3.1. Finally, the ACU determines that the stream can be admitted when the
 13 following inequality is satisfied:

$$\frac{TXOP_{k+1}}{SI} + \sum_{i=1}^k \frac{TXOP_i}{SI} \leq \frac{T - T_{CP}}{T}$$

14 where k is the number of existing streams and k+1 is used as index for the newly arriving stream. T indicates
 15 the beacon interval and T_{CP} is the time used for EDCA traffic.

16 The ACU needs to ensure that it complies with the dot11CAPlimit, i.e., the scheduler does not allocate
 17 TXOPs that exceed dot11CAPlimit. The ACU might also consider additional time to allow for
 18 retransmissions.

19 The ACU ensures that all admitted streams have guaranteed access to the channel. Any modification can be
 20 implemented for the design of the ACU. For example, UP based ACU is possible by examining the UP Field
 21 in TSPEC to decide whether to admit, retain or drop a stream. If the UP is not specified, a default value of '0'
 22 is used. If a higher UP stream needs to be serviced, an ACU might drop lower UP streams.

23