Draft Supplement 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)

Sponsored by the
IEEE 802 Committee
of the
IEEE Computer Society

Introduction

(This introduction is not part of IEEE P802.11e, Draft Supplement 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))

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

Current Chair, Chair

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

Contents

Draft Supplement 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 supplement is based on the current edition of IEEE Std 802.11, 1999 Edition and the 802.11a, 802.11b and 802.11d supplements]

NOTE—The editing instructions contained in this supplement 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.

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 two citations at the appropriate locations 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 access point in 3.2 as follows:

3.2 access point (AP): Any entity that has station functionality and provides access to the distribution services, via the wireless medium (WM) for associated stations. As used in this standard, "AP" applies to all APs regardless of their capabilities.

Change the definition of basic service set in 3.7 as follows:

3.7 basic service set (BSS): A set of stations controlled by a single coordination function. As used in this standard, “BSS” applies to all BSS regardless of the availability of QoS facility.

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) or may have one point coordination function (PCF) and will have one distributed coordination function (DCF). The coordination function within a QoS basic service set (QBSS) will have one hybrid coordination function (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 pollable: A station able to (1) respond to a contention free 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. A QSTA need not be able to interpret ""piggyback"" acknowledgments to be contention free pollable.

Change the definition of distribution system in 3.20, and insert the informative footnote, as follows:

3.20 distribution system (DS): A system used to interconnect a set of basic service sets (BSSs) and integrated local area networks (LANs) to create an extended service set (ESS).

Change the definition of extended service set in 3.25 as follows:

3.25 extended service set (ESS): A set of one or more interconnected basic service sets (BSSs) and integrated local area networks (LANs) that appears as a single BSS to the logical link control sublayer at any station associated with one of those BSSs.

Change the definition of independent basic service set in 3.27 as follows:

3.27 independent basic service set (IBSS): A BSS that forms a self-contained network, and in which no access to a distribution system (DS) is available.

Change the definition of station in 3.42 as follows:

3.42 station (STA): Any device that contains an IEEE 802.11 conformant medium access control (MAC) and physical layer (PHY) interface to the wireless medium (WM). As used in this standard, "STA" applies to all stations regardless of their capabilities.

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

3.51 access category (AC): A variant of the DCF that contends for TXOPs using a set of EDCF channel access parameters from the QoS Parameter Set element in the beacon and Probe Response. Each QSTA has 4 ACs.

3.52 contention free burst (CFB): A set of frame exchange sequences, separated by SIFS, initiated by a single non-AP QSTA. A CFB is initiated by a non-AP QSTA after either receiving a QoS (+)CF-Poll frame, i.e., within a CAP in a CP, or on winning an EDCF contention. A CFB exists during a TXOP.

3.53 contention free period (CFP): A time period during operation of a basic service set (BSS) when a point coordination function (PCF) or hybrid coordination function (HCF) is active, and the right to transmit is assigned to stations by a point coordinator (PC) or a hybrid coordinator (HC), allowing frame exchanges to occur without contention between members of that BSS for the wireless medium (WM).

3.54 contention period (CP): A time period during operation of a basic service set (BSS) when a distributed coordination function (DCF) or hybrid coordination function (HCF) is active, and the right to transmit is either determined locally as stations with pending transfers contend for the WM using a carrier sense multiple access algorithm with collision avoidance (CSMA/CA), or is assigned to QoS enhanced stations (QSTAs) by the hybrid coordinator (HC).

3.55 controlled access phase (CAP): A time period during which a set of frame exchange sequences is started with a transmission by the HC. This time period may include one or more CFBs, initiated by QoS (+)CF-Polls transmitted from the HC. It may span multiple TXOPs.

3.56 differentiation of priorities, or differing priorities: Differentiation, or differing priorities, for the purposes of this standard, refers to a relative difference in the AC parameters that a non-AP QSTA uses to gain access to the medium.

3.57 direct link: A 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 are exchanged directly between the two non-AP QSTAs.

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

3.59 enhanced distributed coordination function (EDCF): A contention-based channel access used by all QSTAs in the QBSS. EDCF does not contain a coordination function; however, as the rules of operations for channel access are similar to that of DCF, the term EDCF is used.

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 in order to increase the probability of successful transfer across the WM and/or in order to use available TXOP duration limits efficiently in cases where the remaining TXOP duration is shorter than the time required to transmit the entire pending MSDU. The process of recombining a set of fragment MPDUs into an MSDU or MMPDU is known as defragmentation.

3.61 hidden station: A station with transmissions that are not detectable via carrier sense by a given transmitting station, but may interfere with a given receiving station.

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

[1] One contention-based channel access mechanism of the HCF is referred to as the "EDCF." It is closely related to the DCF channel access mechanism being viewed as an "Enhanced" version of that mechanism. Despite the presence of "CF" in its name, the EDCF is part of the HCF and is not a separate coordination function.

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

3.64 link: In relation to any IEEE 802.11 MAC entity, a physical, unidirectional path used to exchange MSDUs with a peer entity, including exactly one traversal of the wireless medium.

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

3.66 non-QoS AP: An AP that does not support the QoS facility, but compliant with IEEE Std. 802.11, 1999.

3.67 non-QoS BSS: A BSS that does not include the QoS facility.

3.68 non-QoS STA: A STA that does not support the QoS facility, but compliant with IEEE Std. 802.11, 1999.

3.69 QoS access point (QAP): An access point (AP) that implements the access point functionality specified for the QoS facility in this standard. A QAP differentiates among frames with different traffic identifiers within the traffic to/from each associated QSTA, and supports the hybrid coordination function (HCF). The functions of a QAP are a superset of the functions of a non-QoS AP, and thus a QAP is able to function as a non-QoS AP to non-QoS STAs.

3.70 QoS basic service set (QBSS): A basic service set (BSS) that supports LAN applications with quality of service (QoS) requirements by providing a QoS facility for communication via the wireless medium (WM) as specified in this standard.

3.71 QoS facility: The set of enhanced functions, formats, frame exchange sequences and managed objects to support the selective handling of 4 access categories, or 8 traffic streams per wireless link. The handling of MSDUs belonging to different Priorities may vary based on the Access Categories (AC) associated with the priority, as well as the values of other parameters that may be provided by an external management entity in a traffic specification for the particular traffic stream, and link.

3.72 QoS independent basic service set (QIBSS): An independent basic service set (IBSS) in which one or more of its stations support the QoS facility. A QSTA in a QIBSS may use only the HCF contention mechanism to provide differentiation of its data service using fixed AC parameters defined in its MIB.

3.73 QoS station (QSTA): An IEEE 802.11 station (STA) that implements the QoS facility and hybrid coordination function (HCF), and includes an IEEE 802.11-conformant physical (PHY) interface to the wireless medium (WM). A QSTA acts as a non-QoS STA when associated in non-QoS BSS.

3.74 superframe: The period between two successive beacons and is one beacon interval in length.. [2]

3.75 traffic category (TC): A traffic category is a set of MSDUs with a distinct priority, as viewed by higher-layer entities, relative to other MSDUs provided for delivery over the same link. Traffic categories are only meaningful to MAC entities that support quality of service (QoS) within the MAC data service. These MAC entities determine the priority for MSDUs belonging to a particular traffic category using the priority value provided with those MSDUs at the MAC SAP.

---

[2] The term superframe was used in the initial P802.11 drafts in 1994 to mean a beacon interval that included a CFP and a CP, but was superseded by "contention free repetition interval" in 1995. While superframe is occasionally used informally in material pertaining to IEEE Std 802.11-1999, the term does not appear in the normative text of that standard.

3.76 traffic classification (TCLAS): A traffic classification specifies certain parameter values to identify the MSDUs belonging to a particular traffic stream. The classification process, performed above the MAC SAP at a QoS station (QSTA), uses the parameter values of the traffic classification for a given TS to examine each incoming MAC service data units (MSDUs) and determine whether this MSDU belongs to that TS. The structure of the parameter list is beyond the scope of this standard. A detail of the classification process is also beyond the scope of the MAC.

3.77 traffic identifier (TID): Any of the identifiers usable for higher-layer entities to distinguish MSDUs to MAC entities that support quality of service (QoS) within the MAC data service. There are 16 possible TID values, 8 of which identify priorities and the other 8 of which identify a traffic streams (TSIDs).

3.78 traffic specification (TSPEC): A traffic specification describes the QoS characteristics of a traffic stream created by negotiation between a non-AP QSTA and an HC.

A traffic specification always includes a priority value, and optionally include quantitative objectives for, or limits on, traffic attributes such as MSDU sizes and arrival rates, traffic characteristics such as constant vs. variable data rate, maximum delivery delay, maximum delay variance (jitter), etc. and/or other MAC link options such as acknowledgement policy and minimum PHY rate needed. The MAC sublayer provides selective handling of MSDUs in a manner which attempts to honor the applicable traffic specifications. However, parameter values in traffic specifications are objectives, not guarantees, and it may be impossible, or may become impossible, for the MAC sublayer to provide the requested bandwidth and/or service quality, even in cases where the requested bandwidth had been indicated as being available and/or the requested service quality has previously been provided.

3.79 traffic stream (TS): A traffic stream is a set of MSDUs to be delivered subject to the QoS parameter values provided to the MAC in a particular traffic specification (TSPEC). Traffic streams are only meaningful to MAC entities that support quality of service (QoS) within the MAC data service. These MAC entities determine the TSPEC applicable for delivery of MSDUs belonging to a particular traffic stream using the traffic stream identifier (TSID) value provided with those MSDUs at the MAC SAP.

A non-AP QSTA may simultaneously support up to 8 traffic streams from HC to itself and up to 8 traffic streams from itself to other QSTAs, including the HC. The actual number it supports may be less than this due to implementation restrictions.

An HC may simultaneously support up to 8 downlink traffic streams and up to 8 uplink traffic streams per associated non-AP QSTA. The actual number it supports may be less than this due to implementation restrictions.

3.80 traffic stream identifier (TSID): Any of the identifiers usable by higher-layer entities to distinguish MSDUs to MAC entities for parameterized quality of service (QoS) service within the MAC data service.

3.81 transmission opportunity (TXOP): An interval of time when a particular QoS enhanced station (QSTA) has the right to initiate transmissions onto the wireless medium (WM). A TXOP is defined by a starting time and a maximum duration. During the contention period (CP), each TXOP begins either when the medium is determined to be available under the EDCF rules or when the QSTA receives a QoS (+)CF-Poll from the HC. The duration of an EDCF TXOP is limited by a QBSS-wide TXOP limit distributed in beacon frames, while the duration of a polled TXOP is specified in the frame header that includes the QoS (+)CF-Poll function. Within the limits of each TXOP, decisions regarding what to transmit are made locally by the MAC entity at the QSTA that has gained the TXOP.

3.82 uplink: A link from a non-AP STA to an AP.

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
AIFS	arbitration inter frame spacing
APSD	Automatic Power-Save Delivery
CA	collision avoidance
CAP	controlled access phase
CFB	contention free burst
CSMA	carrier sense multiple access
DLP	Direct Link Protocol
EDCF	enhanced distributed coordination function
HC	hybrid coordinator
HCF	hybrid coordination function
PSDU	physical (layer) service data unit
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
TC	traffic category
TCLAS	traffic classification
TID	traffic identifier
TS	traffic stream
TSID	traffic stream identifier
TSPEC	traffic specification
TXOP	transmission opportunity
WLAN	wireless LAN

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 clause b) of the indented list as follows:

b) ~~Are unprotected from outside signals~~Unprotected from non-802.11 conformant signals which share 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 last 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 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 is a single 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, despite the enormous differences in characteristics of the underlying media, the IEEE 802.11 QoS facility incorporates functionality that is not traditional for QoS support by 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 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 enhanced stations (QSTAs) associated with a QoS enhanced access point (QAP) in a QoS BSS (QBSS). A subset of the QoS enhancements may be available for use between QSTAs that are members of the same QoS IBSS (QIBSS). Because a QSTA implements a superset of STA functionality, the QSTA may associate with a non-QoS AP in a non-QoS BSS, 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 non-QoS APs comprise an integrated set of QoS-related formats and functions that 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. However, all service primitives, frame formats, coordination function and frame exchange rules, and management interface functions defined as part of the QoS facility are mandatory, with the exception of the Group Acknowledgement function defined in 9.11 which is an option separate from the core QoS facilities, and the presence of it is indicated by QSTAs separately from the core QoS facility.

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

The first mechanism, designated as EDCF, is based on differentiating priorities at which traffic is to be delivered. This differentiation is achieved through varying the amount of time a station would sense the channel to be idle, the length of the contention window during a backoff or the duration a station may transmit once it has the channel access. Details of this mechanism are provided in clause 9.10.1.

The second mechanism allows for the reservation of transmission opportunities with the HC. A non-AP QSTA based on its requirements requests the HC for transmission opportunities – both for its own transmissions as well as transmissions from the HC to itself.[3] The request itself may be initiated within the MAC of the non-AP QSTA or be requested by the Station Management Entity of the non-AP QSTA. The HC, based on an admission control policy within its SME either accepts or rejects the request. If the request is accepted, it schedules transmission opportunities for the non-AP QSTA. For transmissions for the STA, HC polls a 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 HC queues the frames and delivers them periodically, again based on the parameters supplied by the non-AP QSTA. Details of the mechanism are provided in clauses 9.10.2 and 11.5. This mechanism is expected to be used for applications such as voice and video which may need a periodic service from the HC. If the application constraints dictate the use of this mechanism, it shall initiate this mechanism using the Station management Entity messages.

Non-QoS STAs may associate in a QBSS. All frames that are sent to non-QoS STAs by an AP are conformant to IEEE Std. 802.11, 1999 or its extensions, if applicable. An AP also sends frames that are defined in IEEE Std. 802.11, 1999 or its extensions, if applicable.

5.3 Logical service interfaces

Insert the following item at the end of the list:

j) QoS traffic scheduling (QoS facility only)

5.3.1 Station service (SS)

Insert the following item at the end of the list:

e) 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)

5.4 Overview of the services

Change the first paragraph as follows:

There are ~~nine~~ several 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 of LAN applications with QoS requirements.

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 hybrid coordination function (HCF) using either contention-based or controlled channel access. At each transmission opportunity (TXOP), a traffic scheduling entity at the QSTA or QAP selects a frame for transmission, from the set of frames at the heads of a plurality of traffic queues, based on requested priority and/or parameter values in the traffic specification for the requested MSDU. Additional information is available in clause 9.10.2.4.

[3] In the case of downstream traffic streams.

5.4.2 Services that support the distribution service

5.4.2.1 Mobility types

Change the item b) as follows:

b) *BSS-transition:* This type is defined as a station movement from one BSS in one ESS to another BSS within the same ESS. In a QBSS, end-to-end QoS connections are maintained, although user-detectable, temporary disruption may occur during handover.

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 features (per MSDU traffic class and TSPEC negotiation) allow 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 within its own boundary 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's 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 STAs. The service is usable by more than one application at a time.

Although the timer synchronization methods and accuracy requirements are application-dependent and beyond the scope of this standard, they rely on an indication from each STA's MAC that is provided essentially simultaneously to the STAs. The MAC accomplishes this by indicating the occurrence 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 STAs 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 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 class 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 frame control (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) Action

3) Control frames

i) PS-Poll

ii) Group Ack (GroupAck)

iii) Group Ack Request (GroupAckReq)

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 used EDCF contention. The parameters that control differentiation of traffic classes in EDCF are fixed. A QIBSS does not support setting up TSPECS, has no HC and cannot support polled TXOP operation.

5.7 Message information contents that support the services

5.7.1 Data

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

QoS Data Messages

— 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 STA to STA

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

Insert the following subclause, including tables and figures included therein, after 5.7, renumbering 5.8 as 5.9 and renumbering tables and figures as necessary:

5.8 Direct Link Protocol

In the IEEE 802.11 Std. 1999, in a BSS, STAs are not allowed to transmit frames to other STAs which are not APs. This standard allows the transmission of frames directly from one non-AP QSTA to another by setting up such data transfer using Direct Link Protocol (DLP). The need for this protocol is motivated by the fact that the intended recipient may be in Power Save Mode, in which case it can only be woken up via the AP. The second feature of DLP is to exchange rate set and other information between the sender and the receiver. Finally, DLP messages can be used to attach security information elements. The security elements and the associated security handshake are not part of this section.

This protocol prohibits the stations going into power-save for the active duration of the Direct Stream.

DLP does not apply in an IBSS, where frames are always sent directly from one STA to another.

The DLP handshake is illustrated in Figure 10.1.

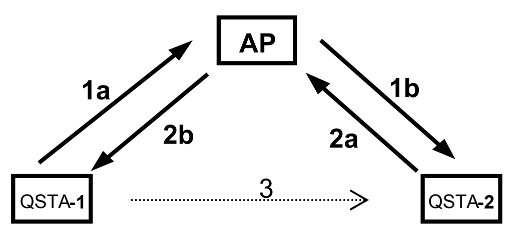


Figure 10.1 - The four steps that are involved in the Direct Link handshake.

A station QSTA-1 that has data to send invokes DLP and sends a DLP-request frame to the AP (1a in Figure 10.1). This request contains the rate set, and (extended) 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).

If QSTA-2 accepts the direct stream, it shall send a DLP-response frame to the AP (2a), 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), 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.

QSTA-2 shall not go into power-save for a duration of aDLPIdleTimeout, after it sent the DLP-action response frame (2a).

When the direct link is active, QSTA-1 may use DLP-probes (3) to gauge the quality of the link between QSTA-1 and QSTA-2.

The direct link becomes inactive when no frames have been exchanged as part of the direct link for a duration of aDLPIdleTimeout. After the timeout, frames with destination QSTA-2 shall be sent via the AP.

By omitting the first step (1a), this protocol can also be invoked by the AP to set up a Direct Link between two associated QSTAs.

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 MAC service data units (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 allows a traffic identifier to be communicated through the MAC SAP on a per-MSDU basis. 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, 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.~~

Insert the following paragraph at the end of 6.1.1:

If the MAC sublayer entity and its association in a QBSS support the QoS facility, the MAC will endeavor to deliver MSDUs with higher priority in preference to other MSDUs with lower priority that may be queued for delivery throughout the BSS. These MAC sublayer entities determine the priorities 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 will endeavor 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, which support the QoS facility, and some STAs, which do not, the non-QoS STA MSDU delivery corresponds to STA delivery of MSDUs with a priority of zero.

Insert the following two subclauses between 6.1.1 and 6.1.2:

6.1.1.1 Determination of priority

The QoS facility supports 8 " priority" values designated 7 to 0 and are identical to the IEEE 802.1D priority tags. All data with a particular priority is said to belong to a traffic category (TC) with that priority. The priority is provided with each MSDU at the medium access control service access point (MAC SAP), either directly, in the priority parameter, or indirectly, in a traffic specification (TSPEC) designated by the priority parameter.

The received unicast frames at the AP may be:

a) Non-QoS subtypes, in which case the AP shall assign a priority of 0 (best effort) to them.
b) QoS subtypes, in which case the AP shall extract the priority from the QoS control field directly if the value of the TID is between 0 and 7, or the "user priority" in the associated TSPEC (contained in the TSInfo field) if the value of TID is between 8 and 15.

In the event that the received frame has a destination address within the BSS, and there is an existing TSPEC set up for the frame, the AP shall determine the transmit queue according to the TSID of the TS with the destination address. In the event that the received frame has a destination address within the BSS and there is no TSPEC set up for the frame, the AP shall determine the transmit queue according to the priority mappings in Table 0.1.

In the event that the received frame had a destination address reachable through the bridge, the AP shall signal to the bridging layer the recovered 802.1D priority which is the priority at the MAC.

6.1.1.2 Interpretation of TID

The value of the priority parameter in the MAC service primitives may be any integer value in the range 0 through 15. This same range of values is used in the traffic identifier (TID) fields that appear in certain frames 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 priorities. Outgoing MSDUs with priority values 0 through 7 are handled by MAC entities at QSTAs in accordance with the local significance of the priority, and use the current, local default values for all other QoS parameters.

Priority parameter and TID field values 8 through 15 are interpreted as traffic stream identifiers (TSIDs), and select the TSPEC for the TS designated by the TID value minus 8 (i.e., TID 8 selects TSPEC 0, TID 9 selects TSPEC 1, ..., TID 15 selects TSPEC 7). Outgoing MSDUs with priority parameter values 8 through 15 are handled by MAC entities at QSTAs in accordance with the local significance of the priority value determined from the priority parameter in the selected TSPEC, and using any non-null values in the selected TSPEC in place of the default values for the corresponding QoS parameters. Selection of a TSPEC for which the MLME has not provided (non-null) QoS parameter values is equivalent to using a TSPEC with a priority parameter equal to the TID value minus 8 and the current, local default values for all other QoS parameters.

Even if the data associated with a TSPEC with an IEEE 802.1d priority tag the priority parameter that is passed at the MAC service primitive is still the TSID.

6.1.1.3 Access Categories

A STA accesses the channel based on the access category of the frame that is to be transmitted. The access category (AC) is derived from the priorities as shown in Table 0.1.

Table 0.1 –Priority to Access Category mappings

Priority (Same as 802.1D Priority)	802.1D Designation	Access Category (AC)	Designation (Informative)
1	BK	0	Best Effort
2	-	0	Best Effort
0	BE	0	Best Effort
3	EE	1	Video Probe
4	CL	2	Video
5	VI	2	Video

6	VO	3	Voice
7	NC	3	Voice

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. 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 priorities 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~~unicast MSDUs, and the reordering of MSDUs with different TID values, originating from a single source station address. There is no reordering of unicast MSDUs with the same TID value and addressed to the same destination.

In non-QoS STAs, ~~I~~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. The StrictlyOrdered service class is not available in QSTAs.

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

6.1.4 MAC Data service architecture

The MAC data plane architecture (i.e. those process that involve 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 in order: Power Save defer queuing, fragmentation, encryption and MPDU formatting. At some point the Data MPDUs 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 above.

During reception, a received Data MPDU goes through processes of MPDU header + CRC validation, duplicate removal, possible reordering if Group Ack mechanism is used, decryption, and defragmentation. The only queuing explicitly introduced by this standard is to perform the defragmentation process. After reassembly, the MSDU may be relayed within the BSS or indicated through the LLC interface.

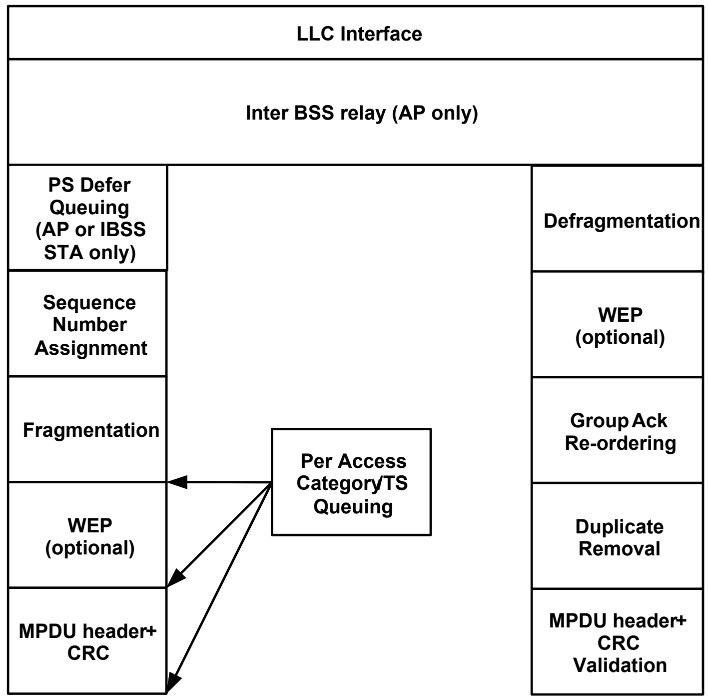


Figure 11.1 – MAC Data-Plane Architecture

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 3 paragraphs as follows:

The data parameter specifies the MSDU to be transmitted by the MAC sublayer entity. For IEEE 802.11, the length of the MSDU shall be less than or equal to 2304 octets. The MAC sublayer entity generates an MPDU that is longer than this MSDU. The increase in length varies due to the operation of several MAC facilities, as specified in subsequent clauses, including 7 and 8. In some cases the MAC transfers unicast MPDUs longer than a specified threshold using two or more shorter MPDUs, performing fragmentation (see 9.4) before transmission and defragmentation (see 9.5) after reception. [4]

The priority parameter specifies the ~~priority desired~~ TID value for the data unit transfer. ~~IEEE 802.11 allows two :~~ The following values are defined at all non-QoS STAs: Contention or ContentionFree. Sixteen additional values are supported only at QSTAs, associated in a QBSS: the integers between and including 0 and 15. 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 considered equivalent to those with TID 6. At QSTAs associated in a non-QoS BSS, MSDUs with any integer priority are considered equivalent to MSDUs with priority Contention.

[4] Several enhancements to the IEEE 802.11 MAC, including but not limited to the QoS facility, generate MPDUs with more MAC-specific information, hence a greater maximum MPDU length, than the earlier revisions of this standard.

The service class parameter specifies the service class desired for the data unit transfer. In non-QoS STAs, IEEE 802.11 allows two values: ReorderableMulticast or StrictlyOrdered. In QSTAs, only a single class of service is defined, so this parameter value has no normative effect and may be null.

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, 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, 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~~ TID value that was used for the data unit transfer. ~~IEEE 802.11 allows two~~ The following values are supported at all STAs: Contention or ContentionFree. Sixteen additional values are supported only at QSTAs that are associated in a QBSS: the integers between and including 0 and 15. 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 with priority of Contention if they are received during the CP, or ContentionFree, if they are received during the CFP. If a QSTA is associated in a non-QoS BSS, the QSTA is functioning as a non-QoS STA, so the priority value is always Contention or ContentionFree.

The service class parameter specifies the receive service class that was used for the data unit transfer. In non-QoS STAs, IEEE 802.11 provides allows two values: ReorderableMulticast or StrictlyOrdered. In QSTAs, only a single class of service is supported and the value of this parameter is null.

6.2.1.2.3 When generated

Change the existing paragraph as follows:

The MA-UNITDATA.indication primitive is passed from the MAC sublayer entity to the LLC sublayer entity or entities to indicate the arrival of an ~~frame~~ MSDU at the local MAC sublayer entity. ~~Frames~~ MSDUs are reported only when complete (e.g. after defragmentation, if received in fragments), and only if the received data frame(s)~~if they~~ are validly formatted at the MAC sublayer, received without error, received with valid (or null) WEP encryption or enhanced security properties satisfying the security policy at the local MAC sublayer entity, and their destination address designates the local MAC sublayer entity.

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 ~~a~~dot11ShortRetryMax or ~~a~~dot11LongRetryMax retry limit would otherwise be exceeded);

c) Excessive data length;

d) Non-null source routing;

e) 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) Unsupported service class (for service classes other than ReorderableMulticast or StrictlyOrdered) for non-QoS STAs; QSTAs do not return this value as there is only one service class;

g) Unavailable priority (for ContentionFree when no point coordinator is available, in which case the MSDU is transmitted with, and the provided priority ~~of~~ parameter value is set to, Contention; or for an integer between and including 1 and 15 at a QSTA which is not associated in a QBSS, in which case the MSDU is transmitted with, and the provided priority parameter value is set to, Contention);

h) 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 there is only one service class;

i) Undeliverable (TransmitMSDUTimer reached dot11MaxTransmitMSDULifetime before successful delivery);

j) Undeliverable (no BSS available);

k) Undeliverable (the STA MAC sublayer entity cannot encrypt with a null key, or does not have the required credentials or other security data to transmit the frame).

The provided priority parameter specifies the ~~priority~~TID that was used for the associated data unit transfer (Contention, ~~or~~ ContentionFree or an integer from 0 to 15).

The provided service class parameter specifies the class of service used for the associated data unit transfer. In non-QoS STAs, the value of this parameter is ~~(~~ReorderableMulticast or StrictlyOrdered~~)~~. In QSTAs, a~~A~~s only a single class of service is supported, the value of this parameter is null.

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 shall be able to properly construct frames for transmission and decode frames upon reception, as specified in this clause. All stations shall be able to validate every received frame using the frame check sequence (FCS), and to decode certain fields from the MAC headers of all frames. Clause 9 requires stations to decode MPDU header fields in all frames received with a valid FCS.

A STA shall be able to construct a subset of these frame formats for transmission, and to decode a (potentially different) subset of these frame formats upon validation following reception. The particular subset of these frames that a station shall construct and decode is determined by the functions supported by that particular station, as specified in 7.5.

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 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 at the end 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.

A STA shall ignore the contents of any field marked “Reserved” or “Undefined” unless otherwise specified in this clause.

A STA shall transmit any field marked “Reserved” set to the value zero. A STA shall not transmit any field containing a value that is marked “Reserved”. [5] [6]

All future extensions to this standard with the same protocol version will retain the order of existing fields in figure 12 as well as retain the encoding of those fields from prior versions of the standard.

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;

— "QoS (+)Null frame" refers to all 4 QoS data subtypes with “no data” ;

[5] This also prevents a STA from transmitting management elements with reserved ID values.

[6] A station conformant to an older revision of this standard can receive frames with what it considers to be reserved values in non-reserved fields and subfields. These fields, along with other fields in the same frame whose interpretation is directly dependent thereon, are ignored on reception, with the exception of the protocol version field.

— "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 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 type~~s~~ is defined in 7.2. The components of management frame bodies are defined in 7.3.

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 encryption encapsulation.

Octets: 2	2	6	6	6	2	6	2	~~0-2312~~n	4
Frame Control	**Duration / ID**	**Address 1**	**Address 2**	**Address 3**	**Sequence Control**	**Address 4**	**QoS Control**	**Frame Body**	**FCS**
MAC Header									

Figure 12 – MAC frame format

7.1.3 Frame fields

7.1.3.1 Frame Control field

Change the text in 7.1.3.1 and Figure 13 as follows:

The Frame Control field consists of the following subfields: Protocol Version, Type, Subtype, To DS, From DS, More Fragments, Retry, Power Management, More Data, Wired Equivalent Privacy (WEP) and Order. The format of the Frame Control field is illustrated in Figure 13.

B0 B1	B2 B3	B4 B7	B8	B9	B10	B11	B12	B13	B14	B15
Protocol Version	**Type**	**Subtype**	**To DS**	**From DS**	**More Frag**	**Retry**	**Pwr Mgt**	**More Data**	**WEP**	**Order**
Bits: 2	2	4	1	1	1	1	1	1	1	1

Figure 13 – Frame Control field

7.1.3.1.2 Type and Subtype fields

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** [7] 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
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
00	Management	1101	Action
00	Management	11~~0~~10-1111	Reserved
01	Control	0000-~~10~~0111	Reserved
01	Control	1000	Group Acknowledgement Request (GroupAckReq)
01	Control	1001	Group Acknowledgement (GroupAck)
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

[7] 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 [7] b7 b6 b5 b4	Subtype description
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~~	QoS Data~~Reserved~~
10	Data	1001	QoS Data + CF-Ack
10	Data	1010	QoS Data + CF-Poll
10	Data	1011	QoS Data + CF-Ack + CF-Poll
10	Data	1100	QoS Null (no data)
10	Data	1101	QoS CF-Ack (no data)
10	Data	1110	QoS CF-Poll (no data)
10	Data	1111	QoS CF-Ack + CF-Poll (no data)
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 by non-QoS STAs associated with an AP, all data type frames sent by QSTAs to the 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. ~~It~~ 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, as well as all management and control type frames.
To DS = 1 From DS = 0	~~Data frame destined for the DS.~~ A data frame from a STA to the DS via a AP.
To DS = 0 From DS = 1	~~Data frame exiting the DS.~~ A data frame from the DS to a STA via a AP.
To DS = 1 From DS = 1	Wireless distribution system (WDS) data frame being distributed from one AP to another AP via the WM.

7.1.3.1.7 Power Management field

Insert at the end of the subclause the following text:

The use of power-save mode precludes the use of direct data link between two non-AP QSTAs and the use of Group Ack.

7.1.3.1.8 More Data field

Change the text in 7.1.3.1.8 as follows:

The More Data field is 1 bit in length and is used to indicate to a STA in power-save mode that more MSDUs or MMPDUs are buffered for that STA at the AP. The More Data field is valid in unicast data or management type frames transmitted by an AP to a~~n~~ STA in power-save mode. A value of 1 indicates that at least one additional buffered MSDU, or MMPDU is present for the same STA.

The More Data field may be set to 1 in unicast data type frames transmitted by a contention-free (CF)-Pollable STA to the Point Coordinator (PC) in response to a CF-Poll to indicate that the STA has at least one additional buffered MSDU available for transmission in response to a subsequent CF-Poll.

The More Data field is set to 0 in all other ~~directed~~ unicast frames, including all QoS data type frames from a non-AP STA.

The More Data field is set to 1 in broadcast/multicast frames transmitted by the AP, when additional broadcast/multicast MSDUs, or MMPDUs, remain to be transmitted by the AP during this beacon interval. The More Data field is set to 0 in broadcast/multicast frames transmitted by the AP when no more broadcast/multicast MSDUs, or MMPDUs, remain to be transmitted by the AP during this beacon interval and in all broadcast/multicast frames transmitted by non-AP stations.

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 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 all other frames, the Duration/ID field contains a duration value as defined for each frame type in 7.2. For frames transmitted during the contention-free period (CFP), the duration field is set to 32768.~~ 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 (msb set to 1 and the 15 lsb set to 0) for transmission and ignored on reception.

c) In all other frames transmitted in a contention period (CP) following a contention access of the channel, the Duration/ID field is set to one of the following three values:

1) If Ack policy of normal acknowledgement is used, the time required for the transmission of one ACK frame (including appropriate IFS values).

2) If Ack policy of normal acknowledgement is used, the time required for the transmission of one ACK followed by another MPDU and its ACK (including appropriate IFS values).

3) The duration of the entire burst as determined by the STA.

d) In the frames sent in a response to a poll from the HC, the Duration/ID field is set to the duration of the entire burst.

Whenever the contents of a received Duration/ID field, treated as an unsigned integer and without regard for address values, type and subtype 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 9.10.2.2.1, 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	Duration value (in units of microseconds) that shall be used within all frames, other than PS-Poll frames, transmitted during the CP, and may be used under HCF for frames transmitted during the CFP.
0	0	1	Fixed value that shall be used under PCF and may be used under HCF, 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

7.1.3.3 Address fields

7.1.3.3.3 BSSID field

The BSSID field is a 48-bit field of the same format as an IEEE 802 MAC address. This field uniquely identifies each BSS. The value of this field, in an infrastructure BSS, is the MAC address currently in use by the STA in the AP of the BSS.

7.1.3.4 Sequence Control field

7.1.3.4.1 Sequence Number field

Change the text of 7.1.3.4.1 as shown:

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~~Non-QoS STAs (as well as QSTAs operating as non-QoS STAs because they are in a non-QoS BSS or non-QoS IBSS), assign~~ed~~ sequence numbers for management type frames, and data type frames from a single modulo-4096 counter, starting at 0 and incrementing by 1 for each MSDU or MMPDU that is assigned a sequence number using this counter. QSTAs associated in a QBSS maintain one additional 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 assigned a sequence number for that TID. Sequence numbers for management type frames sent by QSTAs and QAPs are assigned as specified in 7.2.3. Each Data or Management MPDU containing a fragment of an MSDU or MMPDU contains the ~~assigned~~ sequence number assigned to that MSDU or MMPDU. The sequence number remains constant in all retransmissions of a Data or management MMPDU.

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 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. 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

Bits 0-3	Bit 4	Bits 5-6	Bit 7	Bits 8-15	Usage
TID	Reserved	Ack Policy	Schedule Pending	TXOP limit in units of 32 microseconds	QoS data type frames that include CF-Poll sent by the HC
TID	Reserved	Ack Policy	Schedule Pending	Reserved	QoS data type frames without CF-Poll sent by the HC
TID	Reserved	Ack Policy	Reserved	Queue size in units of 256 octets	QoS data (non-null) frames sent by non-AP QSTAs
TID	0	Ack Policy	Reserved	TXOP duration requested in units of 32 microseconds	QoS null frames sent by non-AP QSTAs
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; or, in the case of QoS null the TC or TS of traffic for which a TXOP is being requested. 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 Figure 14.1. Additional information on the interpretation of the contents of this field appears in 6.1.1.2. [8]

Bit in QoS Control field:	0	1	2	3
Priority for prioritized QoS (TC):	**priority**			**0**
TSID for parameterized QoS:	**TSPEC selector**			**1**

Figure 14.1 – TID field

7.1.3.5.2 Ack Policy Field

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

Table 3.2 – Ack combination in 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 9.10.3
0	1	Reserved
1	0	No Acknowledgement The addressed recipient takes no action upon receipt of the frame. The transmitter shall assume that the frame has been received successfully without regard of the actual result.
1	1	Group Acknowledgement The addressed recipient shall take no action upon the receipt of the frame except for recording the state. The recipient can expect a Group Ack Request frame in the future to which it shall respond with the procedure described in 9.11.

7.1.3.5.3 Schedule pending field

The schedule pending field is set to 1 by the HC if there is a pending Schedule QoS Action frame that needs to be sent to the non-AP QSTA to which the current frame is addressed.

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 is used for TXOPs without an individually specified temporal extent. Any non-AP QSTA receiving a QoS (+)CF-Poll with TXOP limit =0 shall obey the rules pertaining to the temporal extent of EDCF TXOPs under HCF, specified in the HCF TXOP usage rules in 9.10.2.3. In QoS control fields of frames transmitted by an HC with subtypes that do not include CF-Poll the TXOP limit field is reserved.

[8] The presence of the TID field allows an 802.11 QBSS to function as a LAN that is "able to signal the priority" as this phrase is used in IEEE 802.1D.

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 traffic category at the non-AP QSTA sending this frame. The queue size field is present in all non-null QoS data type frames sent by STAs associated in a QBSS. The queue size field is also present in QoS null frames and sent by these stations with bit 4 of the QoS control field set to 1. The queue size value is the ceiling of the total size, in units of 256 octets, of all MSDUs buffered at the QSTA (excluding the frame body 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 QoS null frames sent by non-AP QSTAs associated 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 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. This may be used to cancel a pending unsatisfied TXOP request when its MSDU is no longer queued for transmission.

Change the text of the subclause presently numbered 7.1.3.5 as shown:

7.1.3.6 Frame Body field

The Frame Body is a variable length field and contains information specific to individual frame types and subtypes. The minimum frame body is 0 octets. The maximum length frame body is ~~defined by the maximum length (MSDU + ICV + IV), where ICV and IV are the WEP fields defined in 8.2.5~~ 2304 octets of MSDU or MMPDU content plus 0 or more octets of MPDU expansion to accommodate the fields added by the privacy function, as specified in clause 8.

The following heading is present to show the new number of the FCS field subclause. The contents of the FCS subclause are not modified.

7.1.3.7 FCS field

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, t~~T~~he 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 in contention period, following a contention access of the channel, the duration value is set as specified in 7.1.3.2. For all RTS frames sent under during a polled TXOP under HCF rules, the duration value is set as specified in 9.10.2.3.2.

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, t~~T~~he 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. For CTS frames sent at the start of a polled TXOP under HCF rules, the duration value is set as specified in 9.10.2.3.2.

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 during the contention period, i~~I~~f 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, as well as for all ACK frames sent under HCF rules by QSTAs associated in a QBSS, the duration value is the value obtained from the duration field of the immediately previous data or management 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. For ACK frames sent during the contention-free period (CFP) under PCF the Duration/ID field is set to 32768.

7.2.1.4 Power Save Poll (PS-Poll) frame format

Change the final two paragraphs of 7.2.1.4 as shown:

The BSSID is ~~the address of the STA contained in the AP~~defined in 7.1.3.3.3. The TA is the address of the STA transmitting the frame. The AID is the value assigned to the STA transmitting the frame by the AP in the (re)association response frame which established that STA's current association.

The Duration/ID field contains the AID value in the 11 least-significant bits, and ~~always~~ has ~~its~~ the two most-significant bits each set to 1. All STAs, upon receipt of a PS-Poll frame, update their NAV settings as appropriate under the coordination function rules and data rate selection rules using a duration value equal to the time, in microseconds, required to transmit one ACK frame plus one SIFS interval.

7.2.1.5 CF-End frame format

Change the second paragraph of 7.2.1.5 as shown:

The BSSID is ~~the address of the STA contained in the AP~~defined in 7.1.3.3.3. The RA is the broadcast group address.

7.2.1.6 CF-End + CF-Ack frame format

Change the second paragraph of 7.2.1.6 as shown:

The BSSID is ~~the address of the STA contained in the AP~~defined in 7.1.3.3.3. The RA is the broadcast group address.

The Duration field is set to 0.

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 Group Acknowledgement Request (GroupAckReq) frame format

The frame format of the Group Acknowledgement Request (GroupAckReq) frame is defined in Figure 21.1.

Octets: 2	2	6	6	2	2	4
Frame Control	Duration	RA	TA	GAR Control	Group Ack Starting Sequence Control	FCS
MAC Header						

Figure 21.1 – GroupAckReq frame

The value of Duration/ID shall be such that it covers at least one SIFS and one GroupAck frame. The Group Ack Starting Sequence Control defines an MPDU sequence control number equal to (Starting Sequence number * 16) + Fragment Number. The GAR Control field is shown in Figure 21.2 and contains the Sent Count and TID subfields. The sent count subfield contains the number of unique MPDUs sent by the transmitter beginning with the MPDU indicated by the Group Ack starting sequence control subfield. The TID subfield of GAR Control contains the TID for which a GroupAck frame is requested.

B0 B10	B11	B12 B15
Sent Count	Reserved	TID
Bits: 11	1	4

Figure 21.2 – GAR Control field

7.2.1.8 Group Acknowledgement (GroupAck) frame format

The frame format of the Group Acknowledgement (GroupAck) frame is defined in Figure 21.3.

Octets: 2	2	6	6	2	2	128	4
Frame Control	Duration	RA	TA	GA Control	Group Ack Starting Sequence Control	Group Ack Bitmap	FCS
MAC Header							

Figure 21.3 – Group Ack Frame

The Duration field follows the rules defined in section 9.11.3. The RA and TA fields follow the rules defined in section 7.2.1.3 for the ACK frame. The GA control field defined in Figure 21.4 consists of the Re-ordering Buffer Size, and TID sub-fields. The Re-ordering Buffer Size indicates the number of buffers of size 2304 octets available for grouping for this particular TC or TS (see 7.4.3.2). The Re-ordering buffer size is the total buffer size allocated for this transmission, and shall not be less than the number of the buffer size last negotiated through management frame exchanges.

B0 B7	B8 B11	B12 B15
Re-ordering Buffer Size	Reserved	TID
Bits: 8	4	4

Figure 21.4 – GA Control field

The Group Ack Starting Sequence Control field contains the fragment number and sequence number of the first MPDU for traffic class TID corresponding to bit 0 of the Ack Bitmap. The Group Ack Starting

Sequence Control field defines an MPDU sequence number equal to (Sequence Number * 16) + Fragment Number. The Group Ack Bitmap is 128 octets in length and is for indicating the receiving status of up to 64 MSDUs. Bit position *n* of the Group Ack bitmap, if set to a 1 acknowledges MPDU with MPDU sequence number equal to (Group Ack Starting Sequence Control + *n*) and 0 otherwise. For MSDUs with less than 16 fragments, the bits corresponding to unused fragment numbers are set to 0.

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 ~~in~~dependent on the QoS bit of the subtype field.~~and is as~~ Data frames with a value of 0 in the QoS bit of the subtype field are used for basic transfers, and have the format defined in Figure 22.

Octets: 2	2	6	6	6	2	~~6~~*6 or 0*	~~0-2312~~n	4
Frame Control	Dur/ID	Address 1	Address 2	Address 3	Seq Ctrl	*Address 4*	Frame Body	FCS
MAC Header								

Figure 22 – Basic Data Frame

Data frames with a value of 1 in the QoS bit of the subtype field are used for transmission of MSDUs to other QSTAs, and have the format defined in Figure 22.1. These subtypes 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.

Octets: 2	2	6	6	6	2	*6 or 0*	2	n	4
Frame Control	Dur/ID	Address 1	Address 2	Address 3	Seq Ctrl	*Address 4*	QoS Control	Frame Body	FCS
MAC Header									

Figure 22.1 – QoS Data Frame

The contents of the Address fields of data and QoS 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	STA to STA traffic in an IBSS and QSTA-to-QSTA traffic in a QBSS
0	1	DA	BSSID	SA	N/A	AP-to-STA traffic in a BSS
1	0	BSSID	SA	DA	N/A	STA-to-AP traffic in a BSS
1	1	RA	TA	DA	SA	WDS traffic between APs

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.

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

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

~~The BSSID of the Data 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.~~

The BSSID is defined in 7.1.3.3.3.

The QoS Control field is defined in 7.1.3.5.

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

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

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

Within all data type frames sent by STAs during the CFP under PCF, the Duration/ID field is set to the value 32768. Within all data type frames sent within a polled TXOP under HCF, the Duration/ID field is set as specified in 9.10.2.1. Within all data type frames sent in contention period by the QSTAs, following a contention access of the channel, the Duration/ID field is set as specified in 7.3.1.2. Within all data type frames sent during the contention period by non-QoS STAs, the Duration/ID field is set according to the following rules:

- If the Address 1 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 Address 1 field 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 Address 1 field contains an individual address, the duration value is set to the time, in microseconds, required to transmit the next fragment of this data frame, plus two ACK frames, plus three SIFS intervals.

The duration value calculation for the data 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 data frames to update their NAV settings as appropriate under the coordination function rules.

7.2.3 Management frames

Change the text in 7.2.3 as shown:

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

Octets: 2	2	6	6	6	2	0-2312	4
Frame Control	**Dur/ID**	**DA**	**SA**	**BSSID**	**Seq Ctrl**	**Frame Body**	**FCS**
MAC Header							

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 ~~10~~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 PCF, the Duration field is set to the value 32768. Within all management type frames sent within a polled TXOP under HCF the Duration field is set as specified in 9.10.2.2.2. Within all data type frames sent in the contention period by the QSTAs, following a contention access of the channel, the Duration/ID field is set as specified in 7.3.1.2. Within all management type frames sent during the contention period 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~~ only appear in the specified, relative order. Stations that encounter~~ing~~ an element ~~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:
A portion of the text in this subclause was added to 802.11 by the approval of 802.11d.

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.

Table 5 – Beacon frame body

Usage	Order	Information	Notes
Always present	1	Timestamp	
	2	Beacon interval	
	3	Capability information	
	4	SSID	
	5	Supported rates	
Present if required by PHY type, BSS type, or an active point coordinator (see notes)	6	FH Parameter Set	The FH Parameter Set information element is present within Beacon frames generated by STAs using frequency hopping PHYs.
	7	DS Parameter Set	The DS Parameter Set information element is present within Beacon frames generated by STAs using direct sequence PHYs.
	8	CF Parameter Set	The CF Parameter Set information element is present within Beacon frames generated by APs with an active PC.
	9	IBSS Parameter Set	The IBSS Parameter Set information element is only present within Beacon frames generated by STAs in an IBSS.
	10	TIM	The TIM information element is only present within Beacon frames generated by APs.
Multiple regulatory domains	11	Country Information	The Country Information element shall be present when dot11MultipleDomainCapbilityEnabled is true.
	12	FH Parameters	FH Parameters as specified in clause 7.3.2.13 may be included if dot11MultiDomainCapabilityEnabled is true.
	13	FH Pattern Table	FH Pattern Table information as specified in clause 7.3.2.14 may be included if dot11MultiDomainCapabilityEnabled is true
QBSS	14	QBSS Load	The QBSS Load information element is only present within Beacon frames generated by QAPs.
	15	QoS Parameter Set	The QoS Parameter Set information element is only present within Beacon frames generated by QAPs.
	16	Extended Capabilities	The Extended Capabilities information element is only present in Beacon frames generated by APs with Capability Information bit 15=1.

7.2.3.4 Association Request frame format

Change the contents of Table 7 in 7.2.3.4 as shown:

Table 7 – Association Request frame body

Usage	Order	Information
Always present	1	Capability information
	2	Listen interval
	3	SSID
	4	Supported rates
QBSS	5	Extended Capabilities (only if Capability[15]=1)
	6	Automatic Power Save Delivery

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

Usage	Order	Information
Always present	1	Capability information
	2	Status code
	3	Association identifier (AID)
	4	Supported rates
QBSS	5	Extended Capabilities (only if Capability[15]=1)

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

Usage	Order	Information
Always present	1	Capability information
	2	Listen Interval
	3	Current AP address
	4	SSID
	5	Supported rates
QBSS	6	Extended Capabilities (only if Capability[15]=1)
	7	Automatic Power Save Delivery

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

Usage	Order	Information
Always present	1	Capability information
	2	Status code
	3	Association identifier (AID)
	4	Supported rates
QBSS	5	Extended Capabilities (only if Capability[15]=1)

7.2.3.9 Probe Response frame format

Change the text and contents of Table 12 in 7.2.3.9 as shown:
A portion of the text in this subclause was added to 802.11 by the approval of 802.11d.

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. If the Probe Request was requesting QoS information, the Probe Response contains the additional items shown in the bottom section of Table 12.

Table 12 – Probe Response frame body

Usage	Order	Information	Notes
Always present	1	Timestamp	
	2	Beacon interval	
	3	Capability information	
	4	SSID	
	5	Supported rates	
Present if required by PHY type, BSS type, or an active point coordinator	6	FH Parameter Set	The FH Parameter Set information element is present within Probe Response frames generated by STAs using frequency hopping PHYs.
	7	DS Parameter Set	The DS Parameter Set information element is present within Probe Response frames generated by STAs using direct sequence PHYs.
	8	CF Parameter Set	The CF Parameter Set information element is present within Beacon frames generated by APs with an active PC or by QAPs.
	9	IBSS Parameter Set	The IBSS Parameter Set information element is only present within Probe Response frames generated by STAs in an IBSS.
Multiple regulatory domains	10	Country Information	Included if dot11MultiDomainCapabilityEnabled is true
	11	FH Parameters	FH Parameters as specified in clause 7.3.2.13 may be included if dot11MultiDomainCapabilityEnabled is true
	12	FH Pattern Table	FH Pattern Table information as specified in clause 7.3.2.14 may be included if dot11MultiDomainCapabilityEnabled is true
	13 - n	Requested information elements	Elements requested by the Request information element of the Probe Request frame.
QBSS	n+1	QBSS Load	The QBSS Load information element is only present within Probe Response frames generated by QAPs.
	n+2	QoS Parameter Set	The QoS Parameter Set information element is only present within Beacon frames generated by QAPs.
	n+3	Extended Capabilities	The Extended Capabilities information element is only present in Probe Response frames generated by STAs with Capability Information bit 15=1.

Insert following 7.2.3.11 the following subclause and the figures and tables included therein, renumbering subclauses, figures and tables as necessary:

7.2.3.12 Action frame format

The frame body of a management frame of subtype Action con contains the information shown in Table 15.1.

Table 15.1 – Action frame body

Order	Information
1	Action

7.3 Management frame body components

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 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, and Channel Agility. The remaining bits in the Capability Information field are reserved.~~ The format of the Capability Information field is defined~~as illustrated~~ in Figure 27.

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	rsrv	QoS	APSD	rsrv	rsrv	rsrv	Group Ack	Extended capability element

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.

Non-QoS STAs set the CF-Pollable and CF-Poll Request subfields in Association and Reassociation Request management frames according to Table 16, where QoS = 0. QSTAs desiring to associate in a QBSS set the CF-Pollable and CF-Poll Request subfields in Association and Reassociation Request management frames according to Table 16, where QoS = 1. QSTAs desiring to associate in a BSS set these bits as if they were non-QoS STAs.

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

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

Non-QoS 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, where QoS = 0. A non-QoS AP 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, with the QoS subfield always set to 0.

QAPs set the CF-Pollable and CF-Poll Request subfields in Beacon and Probe Response management frames to indicate the type of contention free service available to STAs, according to Table 17, where QoS = 1. A QAP sets the CF-Pollable, CF-Poll Request, and QoS 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, with the QoS subfield always set to 1.

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

QoS	CF-Pollable	CF-Poll Request	Meaning
0	0	0	No Point Coordinator at AP
0	0	1	Point Coordinator at AP for delivery only (no polling)
0	1	0	Point Coordinator at AP for delivery and polling
0	1	1	Reserved
1	0	0	QAP does not use CFP for delivery of unicast data type frames
1	0	1	QAP uses CFP for delivery, but does not send CF-Polls to non-QoS STAs
1	1	0	QAP uses CFP for delivery, and may send CF-Polls to non-QoS STAs [9]
1	1	1	Reserved

Non-AP QSTAs shall not use the CF-ACK bit in non-QoS data frame subtypes. QSTAs shall not recognize the CF-Poll bit in non QoS-data frame subtypes.

[9] While a QAP may indicate availability of CF-Polls to STAs, and thereby provide non-QoS contention free transfers during the CFP, this 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, nor non-QoS (+)CF-Polls, 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.

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.

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 ~~MMPDU~~frames to indicate that the use of the short preamble option, as described in subclause 18.2.2.2, is allowed within this BSS. To indicate that the use of the short preamble option is not allowed the Short Preamble subfield ~~shall be~~is set to 0 ~~in Beacon, Probe Response, Association Response, and Reassociation Response management MMPDUs transmitted within the BSS~~.

STAs ~~shall~~ set the Short Preamble subfield to 1 in transmitted Association Request and Reassociation Request ~~MMPDU~~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 use of the PBCC modulation option, as described in subclause 18.4.6.6, is allowed within this BSS. To indicate that the use of the PBCC modulation option is not allowed the PBCC subfield ~~shall be~~is 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 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 HR/DSSS PHY. STAs shall set the Channel Agility bit to 1 when Channel Agility is in use, and shall set it to 0 otherwise.~~ STAs set the Channel Agility subfield to 1 to indicate the usage of channel agility by the HR/DSSS PHY. Otherwise STAs set the Channel Agility subfield to 0.

~~Bits 8–15 of the Capability Information field are reserved.~~

STAs set the QoS subfield to 1 with the capability information field if they support QoS. Otherwise, it is set to 0.

APs set the APSD subfield to 1 with the capability information field if they support APSD. Otherwise, APs set it to 0. STAs always set this subfield to 0.

STAs set the Group Ack subfield to 1 within the capability information field if they support Group ACK. Otherwise, it is set to 0.

The Extended Capability Element subfield is set to 1 to indicate that an Extended Capability information element is present in this frame.

7.3.1.7 Reason Code field

Change the contents of Table 18 in clause 7.3.1.7 as shown:

Table 18 – Reason codes

Reason code	Meaning
0	Reserved
1	Unspecified reason
2	Previous authentication no longer valid
3	Deauthenticated because sending station is leaving (or has left) the IBSS or ESS
4	Disassociated due to inactivity
5	Disassociated because AP is unable to handle all currently associated STA
6	Class 2 frame received from non-authenticated station
7	Class 3 frame received from non–associated station
8	Disassociated because sending station is leaving (or has left) the BSS
9	Station requesting (re)association is not authenticated with responding station
10	Disassociated for unspecified, QoS-related reason
11	Disassociated because QAP lacks sufficient bandwidth for this QSTA
12	Reserved
13	Disassociated because of excessive frame losses and/or poor channel conditions
14	Disassociated because QSTA is transmitting outside the limits of its polled TXOPs
15	Reserved
16	QBSS reconfiguration is in progress
1~~0~~7 - 65535	Reserved

7.3.1.9 Status Code field

Change the contents of Table 19 in clause 7.3.1.9 as shown:

Table 19 – Status codes

Status code	Meaning
0	Successful
1	Unspecified failure
2–9	Reserved
10	Cannot support all requested capabilities in the Capability Information field
11	Reassociation denied due to inability to confirm that association exists
12	Association denied due to reason outside the scope of this standard
13	Responding station does not support the specified authentication algorithm
14	Received an Authentication frame with authentication transaction sequence number out of expected sequence
15	Authentication rejected because of challenge failure
16	Authentication rejected due to timeout waiting for next frame in sequence
17	Association denied because AP is unable to handle additional associated STA
18	Association denied due to requesting station not supporting all of the data rates in the BSSBasicRateSet parameter
19	Association denied due to requesting station not supporting the short preamble option
20	Association denied due to requesting station not supporting the PBCC modulation option
21	Association denied due to requesting station not supporting the channel agility option
22	Unspecified, QoS-related failure
23	Association denied due to QAP having insufficient bandwidth to handle another QSTA
24	Association denied due to poor channel conditions
25	Association (with QBSS) denied due to requesting station not supporting the QoS facility
2~~2~~6 - 65535	Reserved

Insert the following subclause, renumbering the subclause, figures and tables where appropriate:

7.3.1.11 Action field

The Action field provides a mechanism for specifying extended management actions. The format of the Action field is shown in Figure 33.1.

	Category	Action Details
Octets:	1	variable

Figure 33.1 – Action Field

The Category field shall be set to one of the non-reserved values shown in Tab1e 1. 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 to a category defined in table 18.1, then the STA shall return the entire Action frame to the source without change except that the most significant bit of the Category field shall be set equal to 1.

The Action Details field contains the details of the action. The details of the actions allowed in each category are described in the appropriate clause referenced in Tab1e 18.1.

Table 18.1 – Category codes

Code	Meaning	See Clause
0	Reserved	
1	QoS management	7.4.1
2	DLP	7.4.2
3-127	Reserved	
128-255	Error	

7.3.2 Information elements

Change the text and contents of Table 20 in 7.3.2 as shown:
A portion of the text in this subclause was added to 802.11 by the approval of 802.11d.

Elements are defined to have a common general format consisting of a 1 octet Element ID field, a 1 octet length field and a variable-length element-specific information field. Each element is assigned a unique Element ID as defined in this standard. The length field specifies the number of octets in the information field. See Figure 34.

Element ID (1 octet)	Length (1 octet)	Information (*length* octets)

Figure 34 – Element format

The set of valid elements is defined in Table 20.

Table 20 – Element IDs

Information Element	Element ID
SSID	0
Supported rates	1
FH Parameter Set	2
DS Parameter Set	3
CF Parameter Set	4
TIM	5
IBSS Parameter Set	6
Country	7
Hopping Pattern Parameters	8
Hopping Pattern Table	9
Request	10
QBSS Load	11
QoS Parameter Set	12
Traffic Specification	13
Traffic Classification	14
Schedule	15
Challenge text	16
Reserved for challenge text extension	17-31
Reserved	32-41
Extended Capability	42
Automatic Power Save Delivery	43
Random Data	44
Reserved	~~32~~45 - 255

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 results in elements ordered by ascending element ID.

Renumber 7.3.2.12 to 7.3.2.15 as 7.3.2.9 to 7.3.2.12.

Insert following 7.3.2.12 the 7.3.x 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 42.5.

Octets: 1	1	2	1	1
Element ID (11)	**Length** (4)	**Station Count**	**Channel Utilization**	**Frame Loss Rate**

Figure 42.5 – QBSS Load element format

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

The channel utilization field is defined as the percentage of time the QAP sensed the medium busy, as a measure of the carrier sense mechanism. This percentage is represented as a moving average of ((channel busy time/(dot11ChannelUtilizationBeaconIntervals * Beacon Interval)) * 100), 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 dot11ChannelUtilizationBeaconIntervals represents the number of target beacon transmission times (TBTTs) during which the average should be calculated. The default value of dot11ChannelUtilizationBeaconIntervals is defined in Annex D.

The frame loss rate field is interpreted as an unsigned integer that indicates the portion of transmitted MPDUs that require retransmission or are discarded as undeliverable. The value is given by: (100 * (total retries + discarded MPDUs and MMPDUs)) / (total MPDU and MMPDU transmission attempts). Where totals should be accumulated over the same period used to calculate the channel utilization value.

7.3.2.14 QoS Parameter Set element

The QoS Parameter Set element provides information needed by QSTAs for proper operation of the QoS facility during the contention period. This information includes the EDCF TXOP limit, the QoS parameter set count, the contention window values, and AIFS values for EDCF channel access. The format of the QoS Parameter Set element is defined in Figure 42.6.

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

Octets: 1	1	2	1 * 4	2 * 4	1 * 4
Element ID (12)	**Length** (44)	**QoS Parameter Set Count**	**CWmin[AC] Values** CWmin[0] … CWmin[3]	**CWmax[AC] values** CWmax[0] … CWmax[3]	**AIFS[AC] values** AIFS[0] … AIFS[3]

2 * 4	2 * 3	2 * 3	2 * 3
TXOPLimit[AC] Values TXOP[0] … TXOP[3]	**TXOPBudget[AC] values** TXOPBudget[1] … TXOPBudget[3]	**Load[AC] values** Load[1] … Load[3]	**SurplusFactor[AC]** SurplusFactor[1] … SurplusFactor[3]

Figure 42.6 – QoS Parameter Set element format

The QoS Parameter set Count is initialized to zero and incremented by one each time the parameter set changes. This field can be used by QSTAs to determine whether the QoS parameter set has changed and requires updating the appropriate MIB values.

The CWmin[AC] values field specifies 4 minimum contention window values, for traffic categories 0 through 3, respectively. Each contention window value is 1 octet in length and contains an unsigned integer. A QSTA shall update the dot11EDCFTableCWmin MIB values according to the CWmin[AC] values in the most recent QoS parameter set element received by the QSTA from the QAP of a QBSS. A QSTA shall use the updated dot11EDCFTableCWmin MIB values for all transmissions within a beacon interval of the reception of the updated QoS parameter set element.

The CWmax[AC] values field specifies 4 maximum contention window values, for traffic categories 0 through 3, respectively. Each contention window value is 2 octets in length and contains an unsigned integer. A QSTA shall update the dot11EDCFTableCWmax MIB values according to the CWmax[AC] values in the most recent QoS parameter set element received by the QSTA from the QAP of a QBSS. A QSTA shall use the updated dot11EDCFTableCWmax MIB values for all transmissions within a beacon interval of the reception of the updated QoS parameter set element.

The AIFS[AC] values field specifies 4 AIFS values, for traffic categories 0 through 3, respectively. Each AIFS value is 1 octet in length and contains an unsigned integer. A QSTA shall update the dot11EDCFTableAIFS MIB values according to the AIFS[AC] values in the most recent QoS parameter set element received by the QSTA from the QAP of a QBSS. A QSTA shall use the updated dot11EDCFTableAIFS MIB values for all transmissions within a beacon interval of the reception of the updated QoS parameter set element.

The TXOPLimit[AC] specifies the time limit on TXOPs when the channel is accessed using contention (i.e., not through polls by the HC), for access categories 0 through 3, respectively. Each TXOPLimit value is 2 octets in length and contains an unsigned integer. All non-polled non-AP QSTA TXOPs during the CP last no longer than the number of 32-microsecond periods specified by the TXOPLimit[AC] value. A TXOPLimit[AC] value of 0 indicates that each TXOP during the CP can be used to transmit a single MPDU at any rate for the frame belonging to that AC. A QSTA shall update the dot11EDCFTableTXOPLimit MIB values according to the TXOPLimit[AC] values in the most recent QoS parameter set element received by the QSTA from the QAP of a QBSS. A QSTA shall use the updated dot11EDCFTableTXOPLimit MIB values for all transmissions within a beacon interval of the reception of the updated QoS parameter set element.

The TXOPBudget[AC] specifies the additional amount of time available during the next beacon interval, for access categories 1 through 3, respectively. Each TXOPBudget value is 2 octets in length and contains a 2's complement signed integer. The available TXOPBudget is the number of 32 microsecond periods specified by the TXOPBudget[AC]. The maximum positive value (32767) is deemed to be infinity.

The Load[AC] specifies the amount of time used during the previous beacon interval, for access categories 1 through 3, respectively. Each Load value is 2 octets in length and contains an unsigned integer. The Load is the number of 32 microsecond periods specified by the Load[AC].

The SurplusFactor[AC] field specifies 3 SurplusFactor values, for traffic categories 1 through 3, respectively. Each SurplusFactor value is a 2 octet field and contains an unsigned binary number with an implicit binary point after the leftmost 3 bits. It represents the ratio of over-the-air bandwidth reserved for that AC to bandwidth of the transported MSDUs required for successful transmission. As such, it must be greater than unity.

The default values for the parameters in QOS Parameter set are defined in Table 20.1.

Table 20.1 – Default QoS Parameter Set

AC	CWmin	CWmax	AIFS	TXOP Limit (802.11b)	TXOP Limit (802.11a/g)	TXOP Budget
0	aCWmin	aCWmax	2	0	0	N/A
1	aCWmin	aCWmax	1	3.0ms	1.5ms	32767 (∞)
2	(aCWmin+1)/2 - 1	aCWmin	1	6.0ms	3.0ms	32767 (∞)
3	(aCWmin+1)/4 - 1	(aCWmin+1)/2 - 1	1	3.0ms	1.5ms	32767 (∞)

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 unidirectional traffic stream, in the context of a particular non-AP QSTA, for use by the HC and non-AP QSTA(s) in support of parameterized QoS traffic transfer using the procedures defined in 11.5. The element information field comprises the items as defined below and the structure is defined in Figure 42.7.

Octets: 1	1	2	2	2	4	4	4	4
Element ID (13)	Length (44)	TS Info	Nominal MSDU Size	Maximum MSDU Size	Minimum Service Interval	Maximum Service Interval	Inactivity Interval	Minimum Data Rate

4	4	4	4	4	2
Mean Data Rate	Maximum Burst Size	Minimum PHY Rate	Peak Data Rate	Delay Bound	Surplus Bandwidth Allowance

Figure 42.7 – 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. The fields are set to zero for any unspecified parameter values.

The structure of the TS Info field is defined in Figure 42.8.

	B0	B1	B2 B3	B4	B5 B7	B8 B9	B10 B12	B13 B15
	Traffic Type	Reserved	TSInfo Ack Policy	Reserved	Priority	Direction	TSID	Reserved
Bits:	1	1	2	1	3	2	3	3

Figure 42.8 – TS Info field

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

The TSInfo Ack Policy sub-field is 2 bits in length and indicates whether MAC acknowledgement is 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 20.2. If the TS Info Ack Policy is set to Group Acknowledgement, the HC shall assume, for TXOP scheduling, that the Immediate Group Ack policy is being used (see 9.10.5).

Table 20.2 – 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 9.10.3.
1	0	No acknowledgement The recipient(s) shall not acknowledge the transmission, and the sender treats the transmission as successful without regard for the actual result.
0	1	Alternate acknowledgement Reserved for future use, interpreted as normal IEEE 802.11 acknowledgement if received.
1	1	Group Acknowledgement A separate Group Ack set up mechanism described in 9.10.5 shall be used.

The Priority subfield is 3 bits that indicates the actual priority value to be used for the transport of MSDUs belonging to this traffic stream in cases where relative prioritization is required.

The direction field defines the direction of Data carried by the traffic stream as defined in table 20.3.

Table 20.3 – Direction field encoding

Bit 8	Bit 9	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	Reserved

The TSID subfield is 4 bits in length and contains the TSID values in the format defined in 7.1.3.5.1. The combination of TSID and Direction identify the traffic stream, in the context of the non-AP QSTA, to which the traffic specification applies. The same TSID may be used for multiple traffic streams at different non-AP QSTA. A non-AP QSTA may use the TSID value for a downlink TSPEC and either an uplink or a direct link TSPEC at the same time. A non-AP QSTA shall not use the same TSID for both uplink and direct link TSPECs.

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 42.9. If the Fixed subfield is set to 1, then the size of the MSDU is fixed and is indicated by the Size Subfield. If the Fixed subfield is set to 0, then the size of the MSDU might not be fixed and the Size indicates the nominal MSDU size. If both Fixed Subfield and Size are set to 0, then the nominal MSDU size is unspecified.

	B0 B14	B15
	Size	Fixed
Bits:	15	1

Figure 42.9 – Nominal MSDU Size Field

The Maximum MSDU Size field is 2 octets long and contains an unsigned integer that specifies the maximum size, in octets, of MSDUs belonging to the TS under this traffic specification.

The Minimum Service Interval field is 4 octets long and contains an unsigned integer that specifies the minimum interval, in units of microseconds, required by the TS in this TSPEC between the start of two successive TXOPs. If the Direction field is set to uplink or sidelink, the Minimum Service Interval is the minimum interval between the start of two successive QoS(+)CF-Polls.

The Maximum Service Interval field is 4 octets long and contains an unsigned integer that specifies the maximum interval, in units of microseconds, required by the TS in this TSPEC between the start of two successive TXOPs. If the Direction field is set to uplink or sidelink, the Maximum Service Interval is the maximum interval between the start of two successive QoS(+)CF-Polls.

The Inactivity Interval field is 4 octets long and contains an unsigned integer that specifies the maximum amount of time in units of microseconds that may elapse without arrival or transfer of an MSDU belonging to the TS before this TS is deleted by the MAC entity at the HC. A value of 0 disables the Inactivity Interval, indicating that the TS is not to be deleted based on inactivity.

The Minimum Data Rate field is 4 octets long and contains an unsigned integer that specifies the lowest data rate, in units of bits per second, for transport of MSDUs belonging to this TS within the bounds under this traffic specification. The Minimum Data Rate does not include the MAC and PHY overheads incurred in transferring the MSDUs. This field is to be used to compute the bandwidth that the TS requires for transmission to meet minimum QoS requirements.

The Mean Data Rate field is 4 octets long and contains an unsigned integer that specifies the average data rate, in units of bits per second, for transport of MSDUs belonging to this TS within the bounds under this traffic specification. The Mean Data Rate corresponds to the rate of the second token bucket in a twin token bucket [B5][10] based traffic policer. The Mean Data Rate does not include the MAC and PHY overheads incurred in transferring the MSDUs.

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

The Peak Data Rate field is 4 octets long and contains an unsigned integer that specifies the maximum allowable data rate in units of bits/second, for transfer of the MSDUs belonging to this TS within the bounds under this traffic specification. The Peak Data Rate corresponds to the rate of the first token bucket in a twin token bucket based traffic policer. If "p" is the peak rate in bit/s, then the maximum amount of data, belonging to this TS, arriving in any time interval [t1,t2], where t1<t2 and t2-t1>1 TU, must not exceed p*(t2-t1) bits.

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

The Delay Bound field is 1 octet long and contains an unsigned integer that specifies the maximum amount of time, in units of microseconds, allowed to transport an MSDU belonging to the TS in this TSPEC, measured between the time marking the arrival of the MSDU at the local MAC sublayer from the local MAC SAP and the time starting the successful transmission or retransmission of the MSDU to the destination.

The Surplus Bandwidth Allowance Factor field is 2 octets long and contains specfies the excess allocation of time (and bandwidth) over and above the stated rates required to transport an MSDU belonging to the TS in this TSPEC. This field is a 2 octet field, which is represented as an unsigned binary number with an implicit binary point after the leftmost 3 bits. This field is included to account for retransmissions, and MAC and PHY overheads. It represents the ratio of over-the-air bandwidth to bandwidth of the transported MSDUs required for successful transmission to meet throughput and delay bounds under this TSPEC, when specified. As such, it must be greater than unity.

The TS Priority, Minimum Data Rate, Mean Data Rate, Peak Data Rate, Maximum Burst Size, Minimum PHY Rate, and Delay Bound fields in a TSPEC express the QoS expectations requested by a non-AP QSTA, when these fields are specified with non-zero values. Unspecified parameters in these fields as indicated by a zero value indicate that the non-AP QSTA does not have requirements for these parameters.

7.3.2.16 Traffic Classification (TCLAS) Element

The Traffic Classification (TCLAS) element represents an abstract container that contains a set of parameters necessary to identify incoming MSDUs with a particular traffic stream to which they belong. The structure of this element is shown in figure 42.10. The content of the element, Frame Classifier field, is undefined as the

[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].

[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.

classification mechanism is beyond the scope of this document. Informative Annex I provides a recommended definition of the container for interoperability purposes.

Octets: 1	1	L
Element ID (14)	**Length** (L)	**Frame Classifier**

Figure 42.10 –TCLAS element format

The Frame Classifier field is 0-255 octets in length with the formats defined as in Annex I.

7.3.2.17 Extended Capability element

The Extended Capability element is present in any management frame body that includes a Capability Information field with its “Extended capability element” subfield set to 1. This element provides additional bits to indicate optional or configurable capabilities. The element information field is a positive integer multiple of 2 octets in length, with a default length of 2 octets, as shown in Figure 42.11.

Element ID (35)	**Length (2*n)**	**Extended Capabilities (2*n octets)**

Figure 42.11 – Extended Capability element format

The Extended Capabilities field is (at least) 2 octets in length, and contains capability information bits as defined in Figure 42.12. [12] Once assigned, the positions of individual capability bits within this field remain fixed. This allows the length of this field to be extended over time without ambiguity. The mutually available capabilities for a pair of QSTAs which use Extended Capability elements of different lengths are, by definition, capabilities indicated by bits starting with bit 0 of the first octet of the element information field and ending with bit 15 of the last octet pair in the shorter of the two elements.

B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12	B13	B14	B15
rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)	rsrv (0)
Octets								2							

Figure 42.12 – Extended Capabilities field (first 2 octets)

7.3.2.18 Automatic Power-Save Delivery Element

The automatic power-save delivery element contains information that a non-AP QSTA can use to indicate to the QAP whether the non-AP QSTA is currently in APSD mode, and how long traffic should be buffered for before being delivered to the non-AP QSTA. The APSD element may be included in (re)association requests in order to activate the facility at association time. The APSD element may also be sent to the QAP using the action frame, to enable or disable automatic power-save delivery. The element information field is defined in Figure 42.13.

Octets: 1	1	1	1
Element ID (TBD)	**Length** (2)	**Wakeup Period**	**Beacon Offset**

Figure 42.13 – Automatic Power-Save Delivery element format

The wakeup period is the number of beacon intervals during which the non-AP QSTA is requesting the QAP to buffer MSDU and management frames before releasing the frames for delivery using a prioritized, or parameterized, delivery mechanism.

[12] Subfields marked “rsrv” are reserved.

The station wakes at a time when (TSF / Beacon Interval) modulo Wakeup period = Beacon offset.

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 element information field is shown in Figure 42.14.

Octets: 1	1	4	4	2	2	2
Element ID (TBD)	**Length** (14)	**Min Service Interval**	**Max Service Interval**	**Minimum TXOP Duration**	**Maximum TXOP Duration**	**Specification Interval**

Figure 42.14 – Schedule Element

The Min Service Interval specifies the minimum of:

a) the minimum time in units of microseconds between the start of successful successive QoS(+)CF-Poll that is sent to a non-AP QSTA for uplink or direct link transmissions, and;

b) the minimum time between successive successful TXOP downstream transmissions to a non-AP QSTA.

The Max Service Interval specifies the maximum of:

a) the maximum time in units of microseconds between the start of scheduled successive QoS(+)CF-Poll that is sent to a non-AP QSTA for uplink or direct link transmissions, and;

b) the maximum time between successive scheduled TXOP downstream transmissions to a non-AP QSTA.

Minimum TXOP Duration specifies the minimum TXOP duration in units of microseconds that is allocated to this non-AP QSTA.

Maximum TXOP Duration specifies the maximum TXOP duration in units of microseconds that is allocated to this non-AP QSTA.

The Specification Interval is a time interval in units of microseconds to verify schedule conformance. A schedule is considered conformant if it provides the specified schedule within any Specification Interval.

7.3.2.20 Random Data

The Random Data Information element is an element with no information. It has an arbitrary bit pattern and an arbitrary length, but upper bounded so that the containing MPDU size does not exceed aFragmentationThreshold. The structure of the element is defined in Figure 42.15.

The Random Data Field shall be discarded by the receiving MAC entity.

Octets: 1	1	L
Element ID (TBD)	**Length** (L)	**Arbitrary Bit pattern**

Figure 42.15 – Random Data element format

Insert following the last 7.3.x subclause the following 7.4.x subclauses with the figures and tables included therein, renumbering as necessary:

7.4 Management Actions

This clause describes the Action frame formats, including the Action Details field, allowed in each of the action categories defined in Table 18.1 in clause 7.3.1.11.

7.4.1 QoS Management Actions

Several Action frame formats are defined for QoS 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 20.4. The Management Action codes within the QoS category are defined in Table 20.4.

Table 20.4 – QoS Action codes

Code	Meaning
0	ADDTS request
1	ADDTS response
2	DELTS
3	Reserved
4	Schedule
5	Reserved
6	ADDGA request
7	ADDGA response
8	DELGA request
9	Reserved
10	APSD
11– 255	Reserved

7.4.1.1 ADDTS request QoS Action frame format

The ADDTS QoS Action 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 Action Body of the ADDTS request QoS Action frame is defined in Figure 42.16.

Octets: 1	1	44	0 or 5-255
Dialog Token	**Reserved**	**TSPEC Element**	**TCLAS Element** (optional)

Figure 42.16 – ADDTS request action body

The Dialog Token, Traffic Specification, and Traffic Classification in this frame are contained in an MLME-ADDTS.request primitive that causes the frame to be sent, except for the Surplus Bandwidth Allowance, Minimum Service Interval, Maximum Service Interval, and Minimum PHY rate, which are generated within the MAC.

The TSPEC element and optional TCLAS Element contain 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 QoS Request frame.

7.4.1.2 ADDTS response QoS Action frame format

The Action Body of the ADDTS response QoS Action frame is defined in Figure 42.17.

Octets: 1	1	44	16	0 or 5-2270
Dialog Token	**Status Code**	**TSPEC Element**	**Schedule Element**	**TCLAS Element** (optional)

Figure 42.17 – ADDTS response action body

The Dialog Token, Traffic Specification, and Traffic Classification in this frame are contained in an MLME-ADDTS.response primitive that causes the frame to be sent. The status codes are defined in table 20.5. The HC announces the schedule in the ADDTS response frame.

Table 20.5 – Status Codes

Status Code	Result Code	Definition
0	SUCCESS	The TS has been created with the parameters contained in the Action Body of the request frame.
1	INVALID_PARAMETERS	No TS has been created because one or more parameters have invalid values.
2	ALTERNATIVE	The TS has been created with the parameters contained in the TSPEC Element in the Action Body of the response frame. These are not the same as the parameters in the request frame.
3	REFUSED	The TS has not been created because the request cannot be honored at this time due to other QoS commitments.
4-255	Reserved	

7.4.1.3 DELTS QoS Action frame format

The DELTS QoS Action request frame is used to delete a traffic stream using the procedures defined in 11.5.

The Action Body of a DELTS QoS Action frame is defined in Figure 42.18. Only the TSID and Direction fields of the TSPEC element are significant, all other fields are undefined.

Octets: 44
TSPEC Element

Figure 42.18 – DELTS action body

A Delete TS QoS Action frame is used to delete a traffic stream characterized by the TSPEC element included in the frame. A Delete TS QoS Action frame may be sent from the HC to the source station and/or destination station(s) of that traffic stream, or vice versa, to indicate an imperative request, to which no response is required from the recipient station(s).

7.4.1.4 Schedule QoS Action Frame format

The Schedule QoS Action Body may be transmitted by the HC to a non-AP QSTA. The Schedule QoS Action frame contains the schedule information element. This information may be used by the non-AP QSTA for power management, internal scheduling or for any other purpose. The action body of the Schedule QoS Action Frame is defined in Figure 42.19.

Octets: 16
Schedule Element

Figure 42.19 – Schedule frame action body

The dialog token in the request frame is copied into the response frame. There are two action-specific status values in the response frame: "action completed successfully" and "unrecognized action".

7.4.1.5 ADDGA request QoS Action frame format

The ADDGA QoS Action frame is used to initiate group acknowledgement for a specific TC or TS between the SA and RA in the header as described in 9.11.

The action body of an ADDGA request QoS Action frame is defined in Figure 42.20. The Dialog Token field shall be set equal to a non-zero value chosen by the STA. TID contains the value of the TC or TS for which the Group Ack is being requested. The transmit buffer size is the available buffer for the group in the sender side. This field is intended to provide guidance for the receiver to decide its Re-ordering buffer size, and is advisory only. When this subfield is set to 0, this information is not available from the transmitter.

		B0 B3	B4 B7	B8 B15
Dialog Token	**Reserved**	**Reserved**	**TID**	**Transmit Buffer Size**
Octets: 1	1	2		

Figure 42.20 – ADDGA request action body

7.4.1.6 ADDGA response QoS Action frame format.

The action body of an ADDGA response QoS Action frame format is defined in Figure 42.21. This frame is sent in response to an ADDGA request QoS Action frame. The dialog token is copied from the corresponding received ADDTS request QoS Action frame. The status codes are defined in Table 20.6. The Group Ack Policy subfield is set to 1 for immediate Group Ack and 0 for delayed Group Ack.

		B0 B2	B3	B4 B7	B8 B15
Dialog Token	**Status Codes**	**Reserved**	**Group Ack Policy**	**TID**	**Reordering Buffer Size**
Octets: 1	1	2			

Figure 42.21 – ADDGA response action body

TID contains the value of the TC or TS for which the Group Ack is being requested. The Re-ordering buffer size indicates the number of buffers of size 2304 octets available for grouping for this particular TID. This number shall be at least 1. [13]

Table 20.6 – ADDGA response QoS action frame status field

[13] For Re-ordering Buffer size, the recipient advertises a single scalar number that is the number of maximum-size fragment buffers available for bursting. 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.

Status Code	Result Code	Definition
0	SUCCESS	The ADDGA request has been successful.
1	REFUSED	The request is refused because the recipient can not support Group Ack
2-255	Reserved	

The action body of an ADDGA response QoS Action frame format is defined in Figure 42.20. This frame is sent in response to an ADDGA request QoS Action frame. The Group Ack Policy subfield is set to 1 for immediate Group Ack and 0 for delayed Group Ack.

TID contains the value of the TC or TS for which the Group Ack is being requested. The Re-ordering buffer size indicates the number of buffers of size 2304 octets available for grouping for this particular TID. This number shall be at least 1. [14]

If the Result Code is set to "REFUSED", the Group Ack Request has been rejected by the intended recipient, and no Group Ack has been set up. In this case, the Group Ack Policy, TID and Re-ordering buffer size fields are undefined. Otherwise, the Re-ordering buffer size indicates the number of fragment buffers available for grouping using this TID.

7.4.1.7 DELGA request QoS Action frame format

The action body of a DELGA request QoS Action frame format is defined in Figure 42.21. This frame is sent to terminate the Group Ack participation by either the originator of the traffic or the recipient. There is no response QoS action frame and the immediate acknowledgement that is sent by the receiver of this frame is considered as a positive response.

B0 B10	B11	B12 B15
Reserved	**Direction**	**TID**

Octets: 2

Figure 42.22 – DELGA request action body

The Direction field indicates if the originator or the recipient of the data sends this frame. It is set to 0 to indicate the originator and 1 the recipient. TID field indicates the TSID or the priority for which the Group Ack has been originally set up.

7.4.1.8 Automatic Power-Save Delivery Action Frame

The automatic power-save delivery action frame contains and is shown in Figure 42.23.

Octets: 1	1	4
Activation Delay	**Reserved**	**Automatic Power-save Delivery Element**

Figure 42.23 – Automatic Power-Save Delivery frame action body

[14] For Re-ordering Buffer size, the recipient advertises a single scalar number that is the number of maximum-size fragment buffers available for bursting. 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.

The activation delay indicates the number of beacon intervals, the activation for automatic power-save delivery is to be delayed.

7.4.2 DLP Action Frames

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 forma.t are defined in Table 20.7.

Table 20.7 – DLP Action Codes

Code	Meaning
0	DLP request
1	DLP response
2	DLP Probe
3-255	Reserved

7.4.2.1 DLP request

The Action body of the DLP-request QoS Action frame body is defined in Table 20.8.

Table 20.8 – DLP request action body

Order	Information	Notes
1	Destination MAC Address	
2	Source MAC Address	
3	Capability Information	
4	Supported rates	
5	Extended Capabilities	The Extended Capabilities information element is only present in DLP Request frames generated by QSTAs with Capability Information bit 15=1.

The Destination MAC address shall be the target destination address.

The Source MAC address shall be the MAC address of the originator.

The Capability information shall be the capability information of the originator of the request.

The supported rates information element shall contain the supported rates information of the originator.

The Extended Capabilities shall be the extended capabilities information element corresponding to those extended capabilities supported by the originator of the request.

7.4.2.2 DLP response

The action body of a DLP-response frame is defined in Table 20.9.

Table 20.9 – DLP response action body

Order	Information	Notes
1	Status Code (1 octet)	
2	Reserved (1 octet)	
1	Destination MAC Address	
2	Source MAC Address	
3	Capability Information	
4	Supported rates	
5	Extended Capabilities	The Extended Capabilities information element is only present in DLP Request frames generated by QSTAs with Capability Information bit 15=1.

The status codes of a DLP-response QoS action frame are defined in Table 20.10.

Table 20.10 – DLP response QoS action frame status field

Status Code	Result Code	Definition
0	SUCCESS	The non-AP QSTA is willing to participate.
2	Not Allowed	Direct Link is not enabled in the BSS policy
3	Not Present	The Destination STA is not present within this QBSS.
4	Not a QSTA	The Destination STA is not a QSTA.
5	Refused	The STA is not willing to participate.
6-255	Reserved	

The Destination MAC address in the DLP-response action body when the frame is sent by the sender shall be the target destination address.

The Source MAC address shall be the MAC address of the originator.

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

The supported rates information element shall contain the supported rates information of the target destination. This information shall only be included if the status code corresponds to Successful (status code 0).

The Extended Capabilities shall be the extended capabilities information element corresponding to those extended capabilities supported by the originator of the response. This information shall only be included in the response if the status code corresponds to Successful (status code 0).

7.4.2.3 DLP probe

The action body of the DLP-probe is defined in Table 20.11.

1 **Table 20.11 – DLP probe action body**

Order	Information
1	Destination MAC Address
2	Source MAC Address
3	Random Data

2

3 ***Insert after 7.4 the following new subclause, including the table therein, renumber items as appropriate:***

7.5 Frame usage

Table 20.12 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 20.12 – Frame subtype usage by BSS type, MAC entity type, and coordination function

Frame subtype	IBSS	non-QoS				QoS	
	CP	CP		CFP		CP & CFP	
	STA	STA	AP	STA	PC	QSTA	HC
(Re)Association Request	- - -	T	R	- - -	- - -	T	R
(Re)Association Response	- - -	R	T	- - -	- - -	R	T
Probe Request	T, Rbe	T	R	- - -	- - -	T, R	R
Probe Response	Tbe, R	R	T	- - -	- - -	T, R	T
Beacon	Tb, R	R	T	R	T	R	T, R
ATIM	T, R	- - -	- - -	- - -	- - -	- - -	- - -
Disassociation	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
Deauthentication	T, R	T, R	T, R	T, R	T, R	T, R	T, R
Action Request/Response	- - -	- - -	- - -	- - -	- - -	T, R	T, R
GroupAck/GroupAckReq	- - -	- - -	- - -	- - -	- - -	T, R	T, R
PS-Poll	- - -	T	R	- - -	- - -	T	R
RTS	T, R	T, R	T, R	- - -	- - -	T, R	T, R
CTS	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
CF-End	(R)	(R)	(R)	R	T	R	T
CF-End+CF-Ack	(R)	(R)	(R)	R	T	(R)	- - -
Null	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
(Data+)CF-Poll+(CF-Ack)	- - -	- - -	- - -	R	T	- - -	- - -
(Data+)CF-Ack	- - -	- - -	- - -	T, R	T, R	- - -	- - -
QoS Null	- - -	- - -	- - -	- - -	- - -	T, R	T, R
QoS Data	- - -	- - -	- - -	- - -	- - -	T, R	T, R
(QoS Data+)CF-Poll	- - -	- - -	- - -	- - -	- - -	R	T
(QoS Data+)CF-Poll+CF-Ack	- - -	- - -	- - -	- - -	- - -	Rda	Tda
(QoS Data+)CF-Ack	- - -	- - -	- - -	- - -	- - -	T, Rda	Tda, R

Symbols:

- T — frame subtype for row is transmitted by MAC entity for column
- R — frame subtype for row is received by MAC entity for column
- (R) — frame subtype for row is received, but only from other BSSs, by MAC entity for column
- Tb, Tbe — frame subtype for row is transmitted by station that most recently won beacon arbitration if "Tbe" is also transmitted by a QSTA in an IBSS pursuant to receiving directed request
- Rbe — frame subtype for row is received by station that most recently won beacon arbitration, also received as directed request by a QSTA in an IBSS
- Tcc — frame subtype for row is transmitted only during controlled contention intervals.

Tda	frame subtype for row is transmitted only if recipient of +Cf-Ack function is addressee.
Rda	frame subtype for row is received if QSTA is addressee
- - -	frame subtype for row is neither received nor transmitted by MAC entity for column

9. MAC sublayer functional description

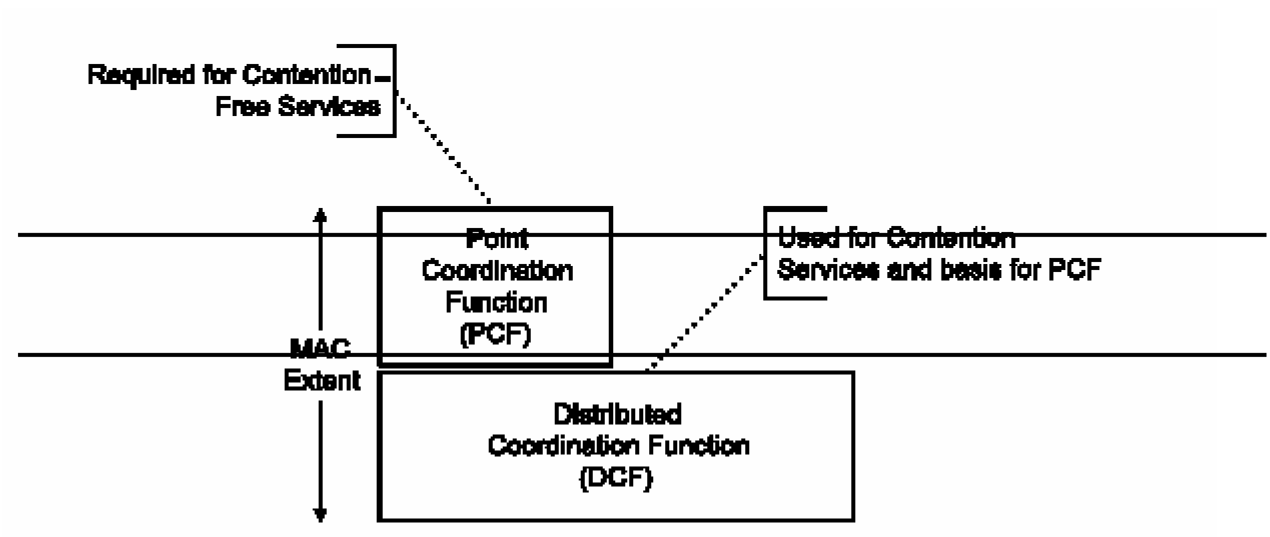
Change the text in the introductory paragraph of 9 as shown below:

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 defined~~expanded on~~ in 9.2 (DCF), ~~and~~ 9.3 (PCF), 9.10.1 (HCF contention-based channel access) and 9.10.2 (HCF controlled channel access)~~, 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 9.7 (DCF and PCF) and 9.10.3 (HCF). ~~Finally, a~~A number of additional restrictions to limit the cases in which MSDUs are reordered or discarded are described in 9.8. Operation across regulatory domains is defined in 9.9. The QoS facility is discussed in 9.10. Group 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.



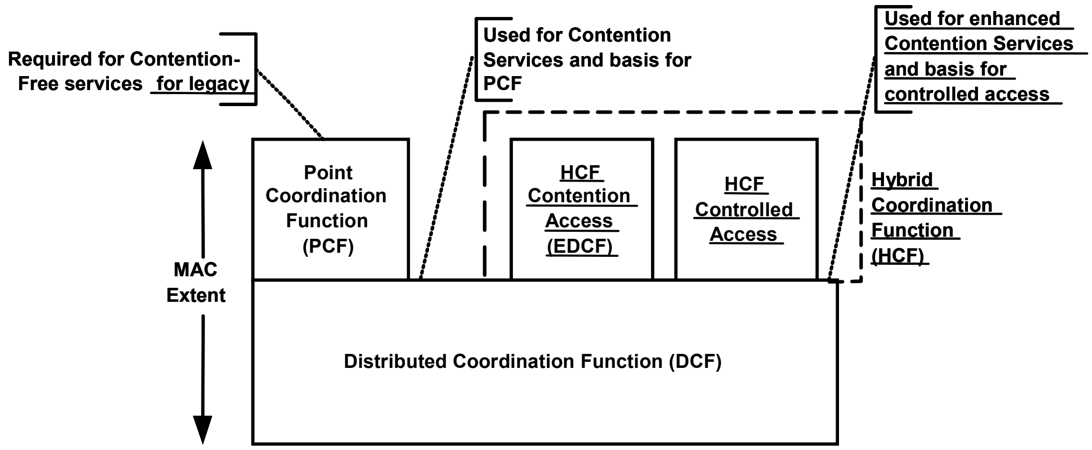


Figure 47 – MAC architecture

The MAC control plane architecture is shown in Figure 47.1 and should be interpreted as described here. The PCF is formed from the following parts: CFP Scheduler, CF-Poll and CF-Delivery. The HCF is formed from all the parts in Figure 47.1 except DCF.

DCF can operate in a fully distributed fashion, without any explicit management. This is the case for an IBSS or an AP without a PC. When a PC is present, the DCF respects this by setting the NAV to protect scheduled CFPs.

The PC uses the CFP to deliver individual CF-Polls to CF-pollable STAs and to deliver its own Data.

EDCF operates as DCF, honoring scheduled CFPs, but also subject to global parameters that control its operation.

An HC uses PIFS to start a CAP, or can schedule them within its CFP. Within the CAP it can deliver data and polled TXOPs to QSTAs.

The HC can also deliver individual CF-Polls to STAs (although this is deprecated) and it can use the CFP for delivery of its own (downlink) data. The delivery of the downlink multicast data is according to the rules in 11.2.1.4 and 11.2.1.5.

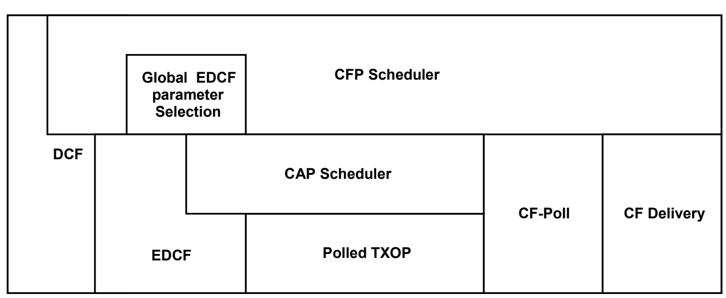


Figure 47.1 – Mac Control-Plane Architecture

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 transfers during both the CP and CFP. The HCF uses a contention-based channel access method, called the enhanced DCF (EDCF) that operates concurrently with a controlled channel access mechanism based on a polling mechanism.

QSTAs may obtain transmission opportunities (TXOPs) using one or both of the channel access mechanisms specified in 9.10. If a TXOP is obtained using the contention-based channel access, it is called an EDCF TXOP. It is called a polled TXOP if it is obtained using the controlled channel access.

If a CFB that is initiated during any TXOP needs local NAV protection, the initiating QSTA may precede the CFB with a CTS frame with the RA equal to the own MAC address, with the appropriate duration to protect the pending CFB.

9.1.3.1 HCF contention-based channel access (EDCF)

The EDCF provides differentiated, distributed access to the WM for 8 priorities for non-AP QSTAs. EDCF channel access defines the access category (AC) mechanism that provides support for the priorities at the non-AP QSTAs. The mapping from priorities to access categories is defined in Table 0.1. Note that there is only one access mechanism at the HC and consequently only one access category, even though there might be multiple queues.

Each AC is an enhanced variant of the DCF. It contends for TXOPs using a set of EDCF channel access parameters from the QoS Parameter Set element, where:

a) The parameters used by an AC to control its operation are defined by dot11EDCFTableAIFS, dot11EDCFTableCWmin, dot11EDCFTableCWmax, dot11EDCFTableTXOPLimit, TXOPBudget[AC] and Load[AC].

b) The minimum specified idle duration time is not the constant value (DIFS) as defined for DCF, but is a distinct value (dot11EDCFTableAIFS, see sections 9.10.1) 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 dot11EDCFTableCWmin and dot11EDCFTableCWmax values, assigned by a management entity or by a QAP.

d) Collisions between contending ACs within a QSTA are resolved within the QSTA such that the data frames from the higher-valued AC receives the TXOP and the data frames from the lower-valued 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 heads of lower-valued ACs, as would be done after a transmission attempt that was unsuccessful due to an actual external collision on the WM.

e) During an EDCF TXOP won by an AC, a QSTA may initiate multiple frame exchange sequences to transmit MMPDUs and/or MSDUs within the same AC or a higher-valued AC. The duration of this EDCF TXOP is bounded by EDCF TXOP Limit [AC] from the most recently received QoS Parameter Set element. A value of 0 for this duration means that the EDCF TXOP is limited to a single MPDU at any rate in the operational set of the QBSS.

f) QSTAs shall ensure that the TXOP Budgets assigned by the QAP for any AC are not violated (See 9.10.1.7).

A QSTA shall update its MIB values of the EDCF access parameters within one beacon interval after receiving an updated QoS parameter set. A QAP may change the EDCF access parameters by changing the QoS Parameter Set element in the beacon and probe response. However, the QAP should change them only rarely.

Management type frames and GroupAckReq and GroupAck Control frames shall be sent by the access category 3. PS-Poll Control frames shall be sent by the access category 0.

The operating rules of HCF contention-based channel access are defined in 9.10.1.

9.1.3.2 HCF controlled channel access

The HCF controlled channel access mechanism uses a QoS-aware point 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 enhanced access point (QAP) of the QBSS and uses the PC's higher priority of access to the WM to initiate frame exchange sequences and to allocate TXOPs to non-AP QSTAs so as to provide limited-duration controlled access phase (CAP) to transfer QoS data.

HC traffic delivery and TXOP allocation may be scheduled during both the CFP and CP, to meet the QoS requirements of particular TCs or TSs. TXOPs, Contention free transfers of QoS traffic from the HC can be based on the HC's QBSS-wide knowledge of the amounts of pending traffic belonging to different TSs and/or TCs and subject to QBSS-specific QoS policies.

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, which is no longer dependent for protection solely on 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 9.10.2.

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 (E)DCF, ~~and~~ PCF and HCF

The DCF and ~~the~~ a point 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, there is a CFP and a CP in each superframe, and non-QoS STAs treat the HC as if it were a PC, using the DCF access method only during the CP. The HCF access methods (polled and contention-based) operate concurrently, throughout the superframe. Concurrent operation allows the polled and contention-based access methods to alternate, within intervals as short as the time to transmit a pair of frame exchange sequences, under rules defined in 9.10.

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 also use fragmentation to increase Media Efficiency by allowing TXOPs to be used which otherwise would be too small. 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 ~~a~~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 ~~a~~dot11FragmentationThreshold, the MSDU or MMPDU shall be fragmented. The MSDU or MMPDU is divided into MPDUs. Each fragment is a frame no larger than ~~a~~dot11FragmentationThreshold. It is possible that any fragment may be a frame smaller than ~~a~~dot11FragmentationThreshold. 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 medium access procedure. The fragments of a single MSDU or MMPDU are sent during a CFP as individual frames obeying the rules of the PC medium access procedure.

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 associated parameters. This may cause one or more Data MPDUs containing the MSDU to be transmitted following fragmentation and encryption encapsulation, as appropriate.

The MA-UNITDATA.indication is generated in response to one or more received Data MPDUs containing part or all of 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 output service request. This TID associates the output data with the AC queue for the indicated traffic.

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.

All Control frames except the GroupAckReq and GroupAck frames shall be transmitted at one of the rates in the BSS basic rate set so they will be understood by all STAs in the BSS.

All frames with multicast and broadcast RA shall be transmitted at one of the rates included in the BSS basic rate set, regardless of their type or subtype.

Data MPDUs, GroupAckReq and GroupAck frames and/or management MPDUs with a unicast RA shall be sent using~~on~~ any ~~supported~~ data rate subject to the following constraints. ~~selected by the rate switching mechanism (whose output is an internal MAC variable called MACCurrentRate, which is used for calculating~~

~~the Duration/ID field of each frame).~~ A STA shall not transmit at a rate that is known not to be supported by the receiver~~destination~~ STA, as reported in the 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, the responding STA shall transmit its Control Response frame (either CTS or ACK), other than the GroupAck control frame, at the highest rate in the BSS basic rate set that is less than or equal to the rate of the immediately previous frame in the frame exchange sequence (as defined in 9.7). In addition, the Control Response frame shall be sent using the same PHY options as the received frame. The GroupAck control frame shall be sent at the same rate as the GroupAckReq frame if it is sent in response to a GroupAckReq frame.

~~For the HR/DSSS PHY, t~~The time required to transmit a frame for use in the Duration/ID field is determined using the PLME-TXTIME.request primitive and the PLME-TXTIME.confirm primitive, ~~both~~ defined in 17.4.3 or 18.3.4 as appropriate. In QSTAs, the Duration/ID field may cover multiple frames and may involve using the PLME-TXTIME.request several times.

~~For the 5 GHz PHY, the time required to transmit a frame for use in the Duration/ID field is determined using the PLME-TXTIME.request primitive and the PLME-TXTIME.confirm primitive. The calculation method of TXTIME duration is defined in 17.4.3.~~

Insert the following subclause, including tables and figures included therein, after 9.9 (added by 802.11d), renumbering tables and figures as necessary:

9.10 HCF

9.10.1 HCF contention-based channel access

The HCF contention-based channel access is also called Enhanced DCF (EDCF) as it is based on the DCF rules.

9.10.1.1 Reference Implementation

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

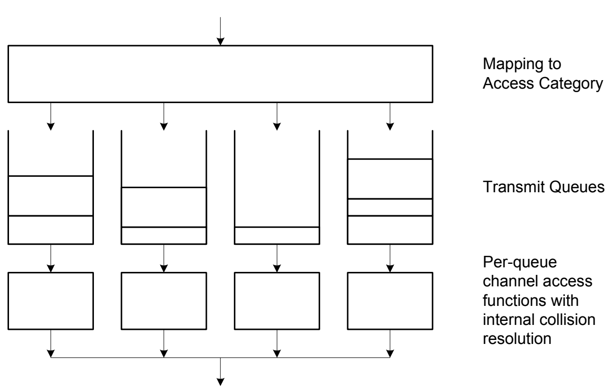


Figure 62.1 – Reference Implementation Model

A model of the reference implementation is shown in Figure 62.1 and illustrates a mapping from frame type or priority to access category, the four transmit queues and four independent channel access functions, one for each queue.

9.10.1.2 Transmit Opportunities & TXOP Limits

All STAs and APs shall maintain a medium occupancy timer for each channel access function[15], which is used to qualify the validity of certain TXOPs.

Each medium occupancy timer shall be initialized to zero. When not zero, it shall count down to zero, and otherwise shall remain at zero.

There are two types of TXOP defined, and rules are defined for each as to how the medium occupancy timer is set and interpreted.

An EDCF TXOP occurs when the EDCF channel access rules permit access to the medium. When a channel access function gains access to the medium by using an EDCF TXOP, it shall set the medium occupancy timer to the value contained in the TXOPLimit parameter corresponding to the access category of the transmitted frame.

A continuation TXOP 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. Such TXOPs have no effect on the medium occupancy timer.

9.10.1.3 Obtaining an EDCF TXOP

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

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

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

An EDCF TXOP is granted to a channel access function when:

a) the medium is indicated by CCA and virtual carrier sense to be idle, and has been idle for a time greater than or equal to AIFSD[AC]+aSlotTime, or a time greater than or equal to EIFS-DIFS+AIFSD[AC] in the event that the previously received frame was in error, and

b) the backoff timer for that channel access function is zero, and

c) these conditions are not simultaneously met by an access category of higher priority.

These conditions may be met immediately at the time a frame is requested to be transmitted in the case that the medium is already idle, or it may be necessary to wait for the expiry of the relevant backoff timer.

The backoff timer is decremented at the end of each backoff slot provided that the medium has been idle for the duration of the slot. Each backoff slot begins immediately following a previous backoff slot or following a period AIFSD[AC] from the end of the last indicated busy medium, or a period of EIFS-DIFS+AIFSD[AC] in the event that the previously received frame was in error.

[15] The specification of an independent medium occupancy timer per channel access function is for ease of specification only; only one timer is actually required since only one of the timers can actually be active at any time

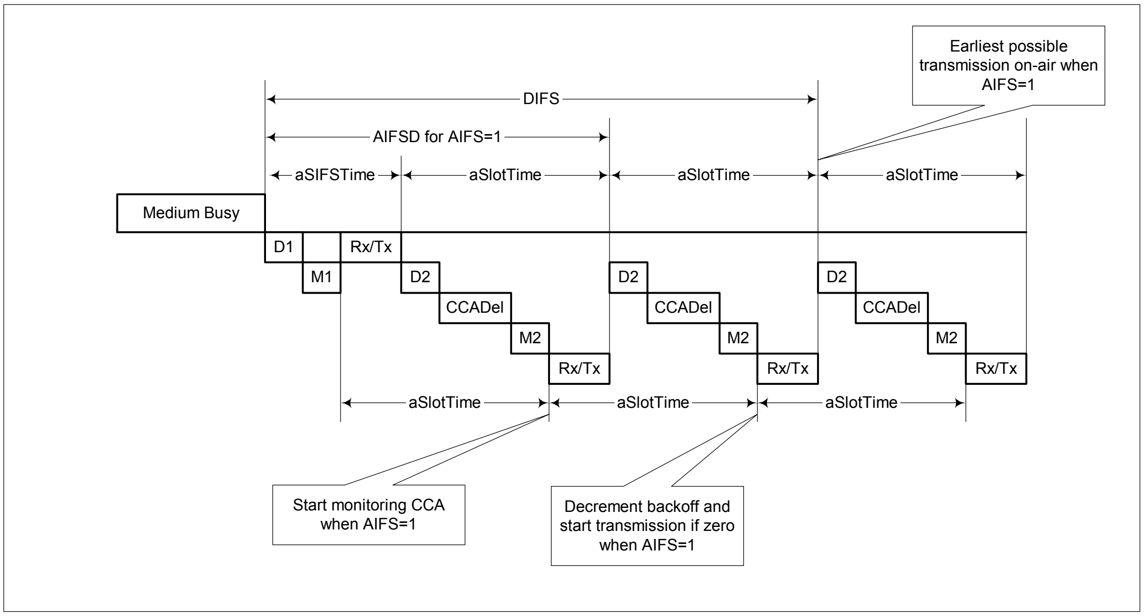


Figure 62.2 – EDCF Timing Relationships

An example showing the relationship between AIFSD, AIFS, 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 AIFS=1, the channel access function starts observing CCA for backoff purposes at a time

aSIFSTime + aSlotTime – aRxTxTurnaroundTime

following the end of the medium busy condition. If the medium remains idle for the duration of aSlotTime, the backoff counter shall be decremented for the first time at a time

aSIFSTime + 2 × aSlotTime - aRxTxTurnaroundTime

following the end of the medium busy condition. 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 EDCF TXOP on-air at a time

aSIFSTime + 2 × aSlotTime

following the end of the medium busy condition.

9.10.1.4 Obtaining a Continuation TXOP

A continuation TXOP is granted to a channel access function at a SIFS period following the successful completion of a transmit frame exchange.

A frame exchange may be either a multicast frame transmitted by an AP 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 STA or an AP.

Note that, as for an EDCF TXOP, a continuation TXOP is granted to a channel access function, not to a non-AP STA or AP, such that a continuation TXOP only permits transmission of a frame of the same priority as that which was granted the EDCF TXOP.

9.10.1.5 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 for a specific AC, 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 MMPDU, 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.

b) A frame is transmitted for that AC and a continuation TXOP cannot be used to send the following frame.

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 MMPDU.

d) The transmission collides internally with another channel access function of higher AC, that is, two channel access functions in the same STA or AP are granted a TXOP at the same time.

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 before invoking the backoff procedure:

a) if the short or long retry count for the STA has reached aShortRetryLimit or aLongRetryLimit respectively, CW[AC] shall be reset to CWmin[AC].

b) Otherwise, if CW[AC] is less than CWmax[AC], CW[AC] shall be set to the value (CW[AC]+1)*2-1

Following the update of the value of CW[AC], the backoff timer is set to a value chosen randomly with a uniform distribution taking values in the range (1,CW[AC]+1).

9.10.1.6 Retransmit Procedures

If a station or AP, in an infrastructure BSS or an IBSS, transmits frames to a destination using QoS data types, it may follow a failed transmission of a frame with an attempt to transmit a frame belonging to a different access category.

If a station or AP, in an infrastructure BSS or an IBSS, does not use QoS data types when sending frames to a particular destination address, it must retry each frame, once an initial attempt to transmit is made, until that frame is either successful or is discarded.

9.10.1.7 Distributed Admission Control Procedures

The amount of on-air time for transmissions of a specific access category (AC) is capped with a hysteresis based distributed admission control mechanism. When the transmission budget for an AC is depleted, new nodes will not be able to gain transmission time, while existing nodes will not be able to increase the transmission time that they are already using. This mechanism protects existing flows.

9.10.1.7.1 Procedure at the AP

The AP shall measure the amount of time occupied by transmissions for each AC during the beacon period, including associated SIFS and ACK times if applicable. The AP shall maintain a set of counters TxTime[AC], which shall be set to zero immediately following transmission of a beacon. For each Data frame received by the AP with the RA equal to the AP MAC address, or transmitted by the AP, and which has a nonzero AC, the AP shall add to the TxTime counter corresponding to the AC of that frame, a time equal to:

a) The time on-air of the frame, including the preamble and PHY header, if the acknowledgement policy is set to “no acknowledgement”.

b) The time on-air of the frame, including the preamble and PHY header, plus the duration of the acknowledgement frame and aSIFSTime if the acknowledgement policy is set to “normal acknowledgement”.

The AP shall transmit in each beacon the TxBudget and SurplusFactor for each AC, contained in the QoS Parameter Set element. The TxBudget is the additional amount of time available for an AC during the next beacon period. The AP shall set the TxBudget to be:

TxBudget[AC] = Max(aACTransmitLimit[AC] – TxTime[AC] * SurplusFactor[AC], 0)

The variable aACTransmitLimit[AC] is a MIB variable at the AP for the maximum amount time that may be spent on transmissions of a specific AC, per beacon interval. This value should be scaled to aDot11BeaconPeriod. If no admission control is applied (for a specific AC), the TxBudget shall be set to 32767, which is deemed to be infinity.

9.10.1.7.2 Procedure at the Station

Stations, including the AP, shall maintain four variables for each of AC, as shown in 20.13.

Table 20.13 - Admission Control variables at the station

State Variable	Description
TxCounter	Counts the transmission time during this beacon interval, not including unsuccessful transmission if the transmission status is available;
TxUsed	All transmission time used.
TxLimit	Limits the counter
TxRemainder	Stores a possibly capped limit remainder
TxMemory	Memorizes the limit

The variable TxCounter counts the amount of time occupied on-air by transmissions from this station for each specific AC, including associated SIFS and ACK times if applicable. For each data frame transmitted by the station which has a nonzero AC, regardless of whether the frame has been successfully transmitted, the station shall add to the TxUsed[AC] corresponding to the AC of the frame, a time equal to:

a) The time on-air of the frame, including the preamble and PHY header, if the acknowledgement policy is set to “no acknowledgement”

b) The time on-air of the frame, including the preamble and PHY header, plus the duration of the acknowledgement frame and aSIFSTime if the acknowledgement policy is set to “normal acknowledgement”

The station also maintains another counter, TxCounter[AC], that counts the successful transmissions. The station shall not transmit a data frame if doing so would result in the value in Txused[AC] exceeding the value in TxLimit[AC]. If the station is prevented from sending a frame for this reason, it may carry over the partial frame time remainder to the next beacon period, by storing the remainder in TxRemainder[AC]:

TxRemainder[AC] = TxLimit[AC] – TxUsed[AC]

Otherwise, TxRemainder[AC] shall be zero.

At each target beacon transmission time, irrespective of whether a beacon was actually received, the TxMemory, TxLimit and TxCounter state variables are updated according to the following procedure:

If TxBudget[AC]=0,

TxMemory[AC] shall be set to zero for new STAs which starts transmission with this AC in the last Beacon interval, other STAs remain TxMemory[AC] unchange;

If the TxBudget[AC] >0,

TxMemory[AC] = f × TxMemory[AC] + (1-f)(TxCounter[AC]*SurplusFactor[AC] + TxBudget[AC])

TxCounter[AC] = 0

TxLimit[AC] = TxMemory[AC] + TxRemainder[AC]

Where the damping factor f is the MIB parameter dot11EDCFTableTXLimitDampingFactor, which has a default value of 0.9. Damping does not affect the entrance of a new flow into the system when enough budget is available, because the decreased TxBudget is offset by an increased TxCounter instantaneously, so TxMemory does not change. The damping does affect TxMemory when a new flow starts up in another node. In that case, the decreased TxBudget is not offset by an increased TxCounter and the TxMemory converges to the lower target value consequently.

The TxBudget that is used in this calculation shall be the budget that was most recently obtained from the AP. The TxCounter value shall be the value of the beacon period before the period that just ended (i.e. if the beacon period that just ended has index k, then TxCounter(k-1) shall be used in the calculation, instead of TxCounter(k)). Taking the earlier value accounts for the delay that occurs between the moment at which the AP determined the TxBudget and the point at which this budget is used in the above calculations.

The value TxCounter + TxBudget is the target to which TxMemory converges. The TxLimit is equal to TxMemory plus a possible capped remainder.

TxMemory ‘memorizes’ the amount of resource the node has been able to spend in a specific AC. Once the budget is depleted (i.e. TxBudget hovers around 0), TxMemory converges to TxCounter, which is the lower limit. This ensures that the node can continue consuming the same amount of resource in following beacon periods. The damping allows for some amount of fluctuation to occur. But TxMemory will not be able to grow any further in the saturated state. This prevents new flows from entering the specific AC when it is saturated. A suitable initial value for this variable could be between 0 and TxBudget[AC]/SurplusFactor[AC].

STAs shall not increase their TxLimit[AC] if they did not transmit traffic with the AC in the previous beacon interval.

Distributed Admission Control should be used for ACs with traffic that is rate capped, like video or voice.

9.10.2 HCF controlled channel access

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

The HC is a type of point coordinator (PC), but differs from the point coordinator 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 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 and QSTAs inherently obey the medium access rules of the HCF, because these rules are based on the DCF, and because each frame transmitted under HCF by the HC or by a non-AP QSTA contains a duration value to cause STAs and QSTAs in the BSS to set their NAV to protect the frames expected to follow that frame.

All QSTAs shall be able to respond to QoS (+)CF-Polls received from an HC.

The HC shall perform delivery of queued broadcast and multicast frames following DTIM beacons in a CFP. The HC may use a longer CFP for QoS delivery and/or QoS polling by continuing with HCF frame exchange sequences, after broadcast/multicast delivery, for a total duration not to exceed dot11CFPMaxDuration. 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. [16]

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 9.10.2.1.2, using dot11HCCWmin, dot11HCCWmax and dot11HCAIFS and the backoff rules in 9.10 and 9.10.1.5.

9.10.2.1 HCF controlled channel access procedure

The HCF transfer protocol is based on a polling scheme controlled by an HC operating at the QAP of the QBSS. The HC gains control of the WM as needed to send QoS traffic to QSTAs and to issue QoS (+)CF-Polls to QSTAs by waiting a shorter time between transmissions than the stations using the EDCF or DCF access procedures. The duration values used in QoS frame exchange sequences reserve the medium for an aSlotTime period longer than the end of the sequence (see Figure 62.3) to permit continuation of a NAV-protected CF transfer by concatenation of CFBs. [17] This extra WM reservation allows the HC to initiate a subsequent TXOP with reduced risk of collision because STAs and all QSTAs other than the TXOP holder and the HC will not be able to begin contending until a DIFS interval later than end of the last transfer within the TXOP.

Because the HC is a type of point coordinator, the HC shall include a CF Parameter Set element in the Beacon frames it generates. This causes a QBSS to appear to be a point-coordinated BSS to STAs. This causes all STAs (other than the HC of the same QBSS) to set their NAVs to the dot11CFPMaxDuration value at TBTT, as specified in 9.3.3.2. This prevents most contention with the CFP by preventing non-polled transmissions by STAs whether or not they are CF-Pollable.

9.10.2.1.1 CAP generation

When the HC needs access to the WM to start a CFB or CFP 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 exchange sequence, with the duration value set as provided in 9.10.2.2.1.

During a CFB or CFP, after each data, QoS data or management type frame with a group address in the Address1 field, the HC shall wait for 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 which convey unicast QoS data or management type frames) during a TXOP the HC or holder of the current TXOP shall wait for one SIFS period and then start transmitting the first frame of the next frame exchange sequence.

9.10.2.1.2 Recovery from the absence of an expected reception

QSTAs, including the HC, are required to respond within any frame exchange sequence after a SIFS period. If the beginning of reception of an expected response, as detected by the occurrence of PHY-CCA.indication(busy) at the QSTA which is expecting the response, does not occur during the first slot time following SIFS then,

a) If the transmitting station is the HC, it may initiate recovery by transmitting a PIFS after the end of the last transmission.

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

[17] The reason for this extra time is due to the possibility of a one-OFDM-symbol-time (4µs) uncertainty in the achievable TSF synchronization tolerance with the 802.11a PHY. This 4µs uncertainty, added to the specified (4µs+PHY propagation delay) TSF synchronization tolerance specified in 11.1.2 is roughly as long as aSlotTime for 802.11a (9µs). This might result in a contending non-AP STA's TSF timer differing by this amount from the HC's timer.

b) If the transmitting station is a non-AP QSTA, it shall initiate recovery by transmitting a PIFS after the end of the last transmission.

If a bad frame, as detected by an FCS error after occurrence of PHY-RXSTART.indicate followed by PHY-RXEND.indicate(no error) is received at a QSTA that expects a response to its transmission, the QSTA may initiate recovery by transmitting a frame after SIFS from the end of that reception. [18]

Non-AP QSTAs that receive a QoS (+)CF-Poll shall respond within a SIFS period. If the polled non-AP QSTA has no queued traffic to send, or if the MPDUs available to send are all too long to transmit within the specified TXOP limit, the non-AP QSTA shall send a QoS (+) Null frame. In the case of no queued traffic, this QoS (+) Null frame shall have a QoS control field that reports a queue size of 0 for any TID. In the case of insufficient TXOP size, this QoS (+) Null frame shall have a QoS control field that contains the TID and TXOP duration needed to send the highest priority MPDU that is ready for transmission.

If PHY-CCA.indication(busy) occurs during the slot following SIFS, followed by PHY-RXSTART.indication or PHY-RXEND.indication prior to PHY-CCA.indication(idle) then an HC may assume that a QoS (+)CF-Poll frame was successfully received by the QSTA.

9.10.2.1.3 CFP generation by the HC

Every HC functions as a point coordinator that uses the CFP for delivery, generating a CFP as shown in Figure 59, 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 non-QoS CF-Polls by a QAP is deprecated (for further discussion see 7.3.1.4).

9.10.2.2 TXOP structure and timing

Under HCF the basic unit of allocation of the right to transmit onto the WM is the TXOP. Each TXOP is defined by an implicit starting time (relative to the end of a preceding frame) and a defined maximum length. The TXOP may be obtained by a non-AP QSTA receiving a QoS (+)CF-Poll during the CP or CFP, or by a QSTA winning an instance of EDCF contention during the CP. In the case of the former (a “polled” TXOP), the entire TXOP is protected by the NAV set by the duration of the frame that contained the QoS (+)CF-Poll function, as shown in Figure 62.3.

Any QoS data type frame of a subclass that includes CF-Poll contains a TXOP limit in its QoS control field. For TXOPs resulting from EDCF contention the TXOP limit value from the QoS Parameter Set element in the most recent Beacon frame shall be used. Within a polled TXOP a QSTA may initiate the transmission of one or more frame exchange sequences, with all such sequences nominally separated by a SIFS interval. The QSTA shall not initiate transmission of a frame unless the transmission, and any acknowledgement or other immediate response expected from the peer MAC entity, are able to complete prior to the end of the remaining TXOP duration.

A TXOP or transmission within a TXOP shall not extend across TBTT, dot11CFPMaxDuration (if during CFP), dot11MaxDwellTime (if using an FH PHY) or dot11CAPLimit. The HC shall ensure that the full duration of any granted TXOP meets these non-AP QSTAs may use the time prior to the TXOP limit of a polled TXOP without checking for these constraints. However, a QSTA shall enforce the TBTT and

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

dot11MaxDwellTime constraints during contention-based (EDCF) TXOPs. Subject to these limitations, all decisions regarding what MSDUs and/or MMPDUs are transmitted during any given TXOP are made by the QSTA which holds the TXOP.[19][20]

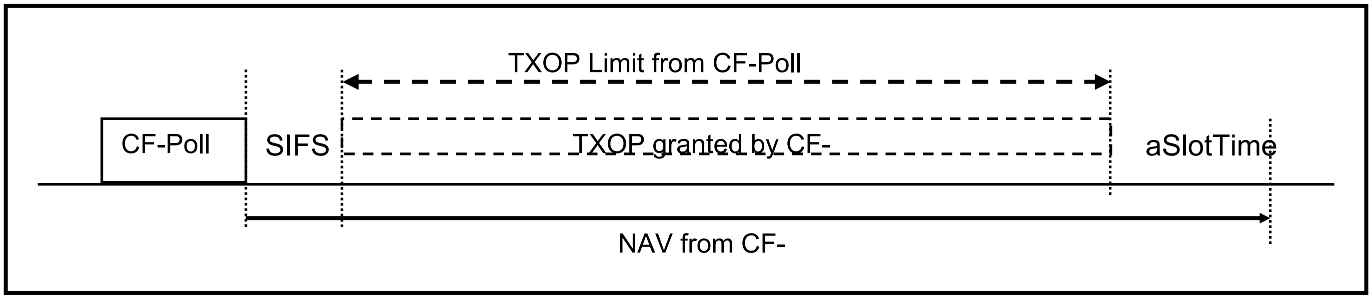


Figure 62.3 – TXOP

9.10.2.2.1 NAV operation during a CAP

The Duration/ID field in any QoS data frame of a subtype that includes CF-Poll shall contain a value that exceeds the TXOP limit specified in the QoS control field by one aSlotTime period. The Duration/ID in any QoS data type frame sent within a polled TXOP shall be set to the number of microseconds remaining until the specified end of the TXOP.

When a QSTA updates its NAV setting using the duration value from a QoS (+)CF-Poll containing the BSSID of this 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, management type, data type or QoS data type frame is received with a MAC address in the SA field (TA field in the case of RTS) which matches this saved TXOP holder address, the QSTA shall send the appropriate response 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 this QBSS shall reset its NAV. A non-AP QSTA that receives a QoS CF-Poll with a MAC address in the Address1 which matches the HC's MAC address and a Duration/ID value equal to zero shall clear its NAV.

Within a polled TXOP, if there is sufficient time left in the TXOP after the transmission of the final frame and its expected ACK response, then the recipient of this final frame shall be HC. If there are no frames to be sent to the HC, then the non-AP QSTA shall send to the HC a QoS Null with queue size subfield in QoS Control set to 0.

If the beginning of the reception of an expected ACK response does not occur, detected as the non-occurrence of PHY-CCA.indication(busy) at the non-AP QSTA which is expecting the response during the first slot time

[19] In certain regulatory domains, channel sensing should 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 should 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 dot11MediumOccupancyLimit TU by waiting for at least for the duration of one PIFS during which it will sense 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 x aSlotTime, where n is the number of times the QSTA needs to sense the channel and is given by floor(TXOP limit/aMediumOccupancyLimit) for the channel sensing.

[20] The TID value in the QoS Control field of a QoS (+)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.

following SIFS, the non-AP QSTA shall retransmit the frame or transmit a QoS Null frame, with the Ack policy set to Normal acknowledgement and the queue size field set to 0, after PIFS from the end of last transmission, until such time that it receives an acknowledgement or when there is not enough time remaining in the TXOP for sending such a frame. This is to avoid the situation where the HC may not receive the frame and may result in an inefficient use of the channel. If the HC does not have any more QSTAs to be polled, it shall send a QoS CF-Poll with the RA matching its own MAC address and with the Duration/ID set to zero. If a PHY-CCA.indication(busy) occurs at the non-AP QSTA which is expecting the ACK response during the first slot following SIFS after the end of the transmission of the final frame, it shall be interpreted as indicating that the channel control has been successfully transferred and no further action is necessary, even though the ACK from HC may by incorrectly received.

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

9.10.2.2.2 Duration values within polled TXOPs

The Duration/ID value in each of the second and subsequent frames sent by a QSTA during a TXOP shall be the Duration/ID value of the preceding frame in the sequence, diminished by the time required to send the response frame plus one SIFS period.

The initial frame of a non-final frame exchange sequence sent by a QSTA within a TXOP shall contain a duration value that is the remaining duration of the TXOP and aSlotTime. The initial frame of the sole or final frame exchange sequence of the TXOP shall contain a duration value that covers the actual remaining time needed for this frame exchange sequence plus one aSlotTime period.

9.10.2.3 HCF controlled channel access transfer rules

A TXOP obtained by receiving a QoS (+)CF-Poll uses the specified TXOP limit, resulting in a CFB that consists of one or more frame exchange sequences with the sole time-related restriction being that the final sequence shall end not later than the TXOP limit. MSDUs may be fragmented in order to fit within TXOPs.

QSTAs shall use QoS data type frames for all MPDU transfers to/from an HC, and should use QoS data type frames for direct non-AP QSTA-to-non-AP QSTA transfers. The TID in the QoS control fields of these frames shall indicate the TC or TS to which the MPDU belongs, and the queue size field shall indicate the amount of queued traffic present in the output queue that the QSTA uses for traffic belonging to this TC or TS. The queue size value reflects the amount on the appropriate queue not including the present MPDU. A non-AP QSTA should acknowledge the receipt of a QoS data type frame received from the HC, subject to normal Ack policy, using a QoS CF-Ack in cases where that non-AP QSTA has new or changed bandwidth requirements, and wants to send the TID and TXOP duration request along the required acknowledgement (also see 9.10.2.3.1).

QSTAs shall be able to transmit and receive QoS CF-Ack frames. QSTAs are not required to be able to transmit QoS data type frames with subtypes that include +CF-Ack. QSTAs shall be able to handle received QoS data type frames with subtypes that include +CF-Ack when the QSTA to which the acknowledgement is directed is the same as the QSTA addressed by the Address1 field of that QoS data type frame. QSTAs are not required to handle received QoS data type frames in which the +CF-Ack function pertains to a different QSTA than is addressed by the Address1 field of that QoS data type frame. The net effect of these restrictions on the use of QoS +CF-Ack is that the principal QoS +CF-Ack subtype that is useful is the QoS Data+CF-Ack, which can be sent by a non-AP QSTA as the first frame in a polled TXOP when that TXOP was conveyed in a QoS Data+CF-Poll(+CF-Ack) and the outgoing frame is directed to the HC's QSTA address. QoS (Data+)CF-Poll+CF-Ack frames are only useful if the HC wants to grant another TXOP to the same non-AP QSTA a SIFS after receiving the final transmission of that non-AP QSTA's previous TXOP.

The HC assumes that all QSTA transfers using non-QoS frames are best effort traffic.

HCF contention-based channel access shall not be used to transmit MSDUs belonging to traffic streams for which the traffic specification as furnished to/by the HC has a specified minimum data rate and a specified delay bound, except as may be necessary to obtain the first polled TXOP from the HC for a newly added or modified traffic stream.

9.10.2.3.1 TXOP requests

Non-AP QSTAs may send TXOP requests during polled TXOPs or EDCF TXOPs using the QoS Control field in the QoS Data, or QoS Null frame, or QoS CF-Ack frame directed to the HC, with the request duration or queue size and TID value indicated to the HC. During the CP, if the Ack policy bit in the QoS control field of this QoS Null frame is set to Normal acknowledgement, the HC shall respond to this QoS Null with a CF-Poll+CF-Ack frame, but the HC is not required to grant a TXOP of the requested length in this frame.

9.10.2.3.2 Use of RTS/CTS

QSTAs may send an RTS as the first frame of any frame exchange sequence for which improved NAV protection is desired, during either the CP or CFP, and without regard for dot11RTSThreshold. For an RTS frame sent at the beginning of a non-final frame exchange within a polled TXOP, as well as for an RTS frame sent by the HC at the beginning of a frame exchange which grants a TXOP, the Duration/ID field shall be set to the number of microseconds remaining until the end of the specified TXOP plus aSlotTime. [21]

For an RTS frame sent at the beginning of the sole or final frame exchange within a polled TXOP, including the case of a polled TXOP with a TXOP limit value of 0, as well as for an RTS frame sent by the HC at the beginning of a frame exchange which does not grant a TXOP, the Duration/ID field shall be set to the time, in microseconds, required to transmit the data or management type frame of the pending frame exchange, plus one CTS time, plus one ACK time, plus three SIFS intervals, plus aSlotTime. If the calculated duration includes a fractional microsecond, that value is rounded up to the next higher integer.

The HC and TXOP holder after transmitting RTS may recover from the failure of the successful CTS reception by transmitting a frame after PIFS following the end of the RTS transmission if the PHY-CCA.indication(busy) does not occur, or after SIFS following the end of the last frame reception if the frame after the RTS transmission is received incorrectly, as determined by an FCS error, after occurrence of PHY-RXSTART.indication and PHY-RXEND.indication(no error).

If NAV protection is desired for a transmission to the HC in response to a QoS data frame with a subtype that includes CF-Poll, the non-AP QSTA is allowed to send a CTS frame with RA containing its own MAC address in order to set the NAV in its own vicinity without the extra time to send an RTS (which is unnecessary because the NAVs in vicinity of the HC were set by the QoS (+)CF-Poll frame). For a CTS frame sent to the non-AP QSTA's own MAC address at the start of a polled TXOP, the Duration/ID field shall be set to the number of microseconds until the end of the TXOP. [22]

---

[21] 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 will set 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 STA in the same BSS will have its NAV set to protect the CFP, which will render those 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.

[22] 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 will set 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

9.10.2.4 Schedule Management by the HC

When the HC provides controlled channel access to non-AP QSTAs, it is responsible to grant or deny polling service based on the admitted TSPEC. If the TSPEC is admitted, the HC is responsible for scheduling channel access to this TSPEC based on the negotiated TSPEC parameters. The polling service based on admitted TSPECs provides a "guaranteed channel access" from the scheduler in order to have its QoS requirements met. This is an achievable goal when the wireless medium operates free of external interference. The nature of wireless communications may preclude absolute guarantees to satisfy QoS requirements. However, in a controlled environment (e.g. no interference), the behavior of the scheduler can be observed and verified to be compliant to meet the service schedule.

The normative behavior of the scheduler is as follows.

- The scheduler shall be implemented such that, under controlled operating conditions, all stations with admitted TS are offered TXOPs that satisfy the service schedule.
- Specifically, if a TSPEC is admitted by the HC, then the scheduler shall send polls anywhere between the Minimum Service interval and the Maximum Service interval within the specification interval. Additionally, the Minimum TXOP duration shall be at least the time to transmit one maximum size MSDU successfully at the minimum PHY rate specified in the TSPEC. 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 aggregates admitted TSPECs for a single non-AP QSTA and establishes a Service Schedule for the non-AP QSTA. The Service Schedule is communicated to the non-AP QSTA in a Schedule element contained in an ADDTS QoS Action response message. The HC can update the Service Schedule at any time by sending a Schedule element in a Schedule QoS Action frame. The updated schedule is in effect when the HC receives the acknowledgement frame for the Schedule QoS Action frame.

A non-AP QSTA cannot directly reject a Service Schedule. A non-AP QSTA can affect the Service Schedule by modifying or deleting its existing TSPECs as specified in clause 11.4.

Clause 9.10.2.4.1 contains guidelines for deriving an aggregate Service Schedule for a single non-AP QSTA from the non-AP QSTA's admitted TSPECs. The Schedule shall meet the QoS requirements specified in the TSPEC.

During any time interval [t1, t2], the cumulative TXOP duration shall be greater than the total time required to transmit all MSDUs (of Nominal MSDU Size) belonging to all the admitted streams, each 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 QoS action frame. Otherwise, the HC may use an observed PHY rate in calculating TXOPs.

If a non-AP QSTA is operating in an active mode, then the Minimum Service Interval specifies the minimum interval between successful QoS (+)CF-Polls sent by the HC. If a non-AP QSTA is operating in power-save mode, then the Minimum Service Interval specifies the minimum interval between the start of successive scheduled service periods, where a "service period" is defined as follows. A service period consists of a set of one or more downlink frames and/or one or more polled TXOPs. A service period starts with the first successful data or QoS (+)CF-Poll transmission by the HC. A service period ends a) after a downlink frame, which does not include a piggybacked poll, is transmitted, by the HC, with the More Data subfield in the Frame Control of the header set to 0 or b) at the end of a polled TXOP. A non-AP QSTA shall be in active mode before the start of a service period.

STA in the same BSS will have its NAV set to protect the CFP, which will render those 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.

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 an admitted stream. These parameters are Mean Data Rate, Nominal MSDU Size and at least one of Maximum Service Interval and Delay Bound in the AddTS QoS Action Request frame. In the AddTS QoS Action Response frame, these parameters are Mean Data Rate, Nominal MSDU Size and Maximum Service Interval and shall be non-zero when a stream is admitted.

If the minimum set of parameters do not have the required non-zero values as specified above in the AddTS QoS Action Request frame, 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 they will be indicated to the WSTA through the AddTS QoS action response frame. 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 priority 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. An example use of the TSPEC for admission control is described in Annex H.

9.10.2.4.1 Guidelines for Deriving Service Schedule Parameters (Informative)

The HC establishes the service interval for each admitted TSPEC 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 TSPEC 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 Minimum 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 TSPECs. The aggregate maximum burst size is equal to the sum of the maximum burst size of all of the non-AP QSTA's admitted TSPECs. 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 Minimum TXOP Duration contained in a non-AP QSTA's Service Schedule is equal to the maximum MSDU transmission time for any of the non-AP QSTA's admitted TSPECs, where the maximum MSDU transmission time is the time required to send the Maximum Size MSDU at the TS' Minimum PHY Rate.

The Maximum TXOP Duration contained in a non-AP QSTA's Service Schedule is bounded by the aggregated Maximum Burst Size.

9.10.2.4.2 Admission Control

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 TSPEC. All of these criteria affect the admissibility of a given TSPEC; any TSPEC may be rejected. If the HC has admitted no TSPECs that require polling, it may not find it necessary to perform the scheduler or related HC functions.

9.10.2.4.3 Reference Design for a Simple Scheduler and Admission Control Unit (Informative)

This clause 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 9.10.2.4. The scheduler and admission control unit use the minimum set of mandatory TSPEC parameters as specified in clause 7.3.2.15.

9.10.2.4.3.1 A Simple Scheduler

This clause includes the reference design for a Simple Scheduler. The simple 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 WSTA in the TSPEC, the Simple Scheduler uses the Maximum Service Interval for the calculation of the schedule.

The schedule generated by the simple scheduler meets the minimum performance requirements as specified in clause 9.10.2.4. 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 it is a submultiple of the beacon interval. This value will be the Scheduled Service Interval for all WSTAs with admitted streams. See figure 62.4.

For the calculation of the TXOP duration for an admitted stream, the simple 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), Size of Maximum 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 frame, 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 clause. The TXOP duration is calculated as follows. First the scheduler calculates the number of MSDUs that arrived at the Mean Data Rate during the

$$N_i = \left\lceil \frac{SI \times \rho_i}{L_i} \right\rceil$$

SI. This number is N:

Then the scheduler calculates the TXOP duration as the maximum of (1) time to transmit N_i frames at R_i and (2) time to transmit one maximum size MSDU at R_i (plus overheads):

$$TXOP_i = \max\left(\frac{N_i \times L_i}{R_i} + O, \frac{M}{R_i} + O \right)$$

An example is shown in figure 62.4. Stream from QSTA “i” is admitted. The beacon interval is 100 ms and the Maximum Service Interval for the stream is 60 ms. The scheduler calculates a Scheduled Service Interval (SI) equal to 50 ms using the steps explained above.

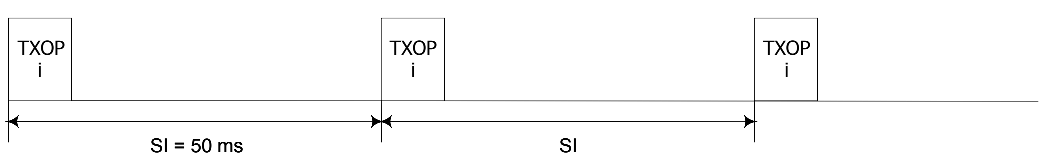


Figure 62.4 – Schedule for stream from QSTA “I”

The same process is repeated continuously while the Maximum Service Interval for the admitted stream is smaller than current SI. An example is shown in figure 62.5:

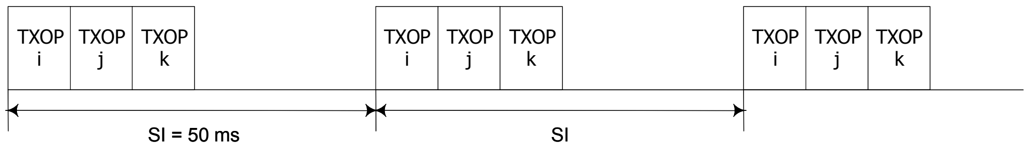


Figure 62.5 – Schedule for streams from QSTAs “I” to “k”

If a new stream is admitted with a Maximum Service Interval smaller than the current SI, the scheduler needs to change the current SI to a smaller number than the Maximum Service Interval of the newly admitted stream. Therefore the TXOP duration for the current admitted streams needs also to be recalculated with the new SI.

If a stream is dropped, the scheduler may use the time available to resume contention. The scheduler may also choose to move the TXOPs for the QSTAs following the QSTA dropped to use the unused time. An example is shown in figure 62.6, when stream for QSTA j is removed. However, this last option may require the announcement of a new schedule to all QSTAs.

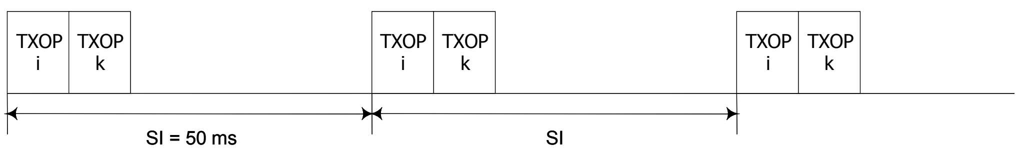


Figure 62.6 – Reallocation of TXOPs when a stream is dropped

Different modifications can be implemented to improve the performance of the minimum scheduler. For example, a scheduler may generate different Scheduled Service Intervals (SI) for different QSTAs, and/or a scheduler may consider a sandbag factor when calculating the TXOP durations to accommodate for retransmissions.

9.10.2.4.3.2 An Admission Control Unit

This clause describes a reference design for an Admission Control Unit (ACU) which administers admission of TSPECs. The ACU uses the same set of parameters that the scheduler uses in clause 9.10.2.4.3.1.

When a new stream requests for admission, the admission control process is done in three steps. First, the ACU calculates the number of MSDUs that will arrive at the Mean Data Rate during the Scheduled Service Interval. The Scheduled Service Interval (SI) is the one that the scheduler calculates for the stream as specified in clause 9.10.2.4.3.1. For the calculation of number of MSDUs the ACU uses the equation for N_i shown in 9.10.2.4.3.1. Second, the ACU calculates the TXOP duration that needs to be allocated for the stream. The ACU uses the equation for $TXOP_i$ shown in 9.10.2.4.3.1. And third the ACU decides if the stream is admitted when the following equation holds:

$$\frac{TXOP_{k+1}}{SI} + \sum_{i=1}^{k} \frac{TXOP_i}{SI} \leq \frac{T - T_{CP}}{T}$$

where k+1 stands for the newly arriving stream and the summation index counts for the streams already admitted and undergoing service from the HC. T indicates the beacon interval and T_{CP} is the time used for EDCF traffic.

The ACU should also consider the value for dot11CAPlimit, i.e., the scheduler shall not allocate TXOPs that exceed dot11CAPlimit. The ACU may also consider additional time to allow for retransmissions.

The ACU ensures that all admitted streams have guaranteed access to the channel. Any modification can be implemented for the design of the ACU. For example, priority based ACU is possible by examining the priority Field in TSPEC to decide whether to admit, retain or drop a stream. If the priority is not specified, a default value of ‘0’ will be used. If a higher priority stream needs to be serviced, an ACU may drop lower priority streams

9.10.3 HCF frame exchange sequences

The allowable frame exchange sequences for use between and among QSTAs and the QAP in a QBSS are shown in Table 25.1. This table uses the same notation as tables 21 and 22 (see 9.7).

Table 25.1 – QBSS frame sequences

HCF Frame Sequence (in CP or CFP unless noted)	Frames in Sequence	Usage
Beacon(CF)	1	beacon during CF period
Data(bc/mc)	1	broadcast or multicast MSDU
Mgmt(bc)	1 or 2	broadcast MMPDU
Mgmt(dir) – ACK	2 or 3	directed MMPDU
QoS CF-Poll{+CF-Ack}(no data) [– CTS self] – QoS Data(dir) – ACK [– QoS Data(dir) – ACK] [-QoS Null]	3 +	polled TXOP, QSTA has frames to send
QoS CF-Poll{+CF-Ack}(no data) – QoS Null	2 or 4	polled TXOP, QSTA has nothing to send (that fits in TXOP)
[RTS – CTS –] QoS Data(dir) – (QoS CF-)Ack(no data)	2 or 4	frame delivery by QSTA or HC, continuation of TXOP
[RTS – CTS –] QoS Data(dir)+CF-Poll{+CF-Ack} – {(QoS CF-)ACK –} QoS Data(dir) –	2, 3 or 5	start of TXOP sent with MPDU

9.11 Group acknowledgment

9.11.1 Introduction

The Group Acknowledgement (Group Ack) mechanism allows a group of QoS Data MPDUs to be transmitted, each separated by a SIFS period. The mechanism is for improving the channel efficiency by aggregating several acknowledgements into one frame. There are two types of Group Ack mechanisms: immediate and delayed. Immediate Group Ack is suitable for high-bandwidth, low latency traffic while the delayed Group Ack is suitable for applications that can tolerate moderate latency. In this clause, the QSTA with data to send is referred to as originator and the receiver of that data as the recipient. [23]

The Group Ack mechanism is initialized by an exchange of ADDGA QoS Action request/response frames. After initialization, groups of QoS data type frame bursts can be transmitted from the originator to the recipient. A burst can be started within a polled TXOP or by winning EDCF contention. The MPDUs within this exchange usually fit within a single TXOP and are all separated by a SIFS. The number of frames in the group is limited, and the amount of state that must be kept by the recipient is bounded. The MPDUs within the group of frames are acknowledged by a GroupAck control frame, which is requested by a GroupAckReq control frame.

[23] The delayed Group 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.

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

Figure 62.4 illustrates the message sequence chart for the set up, data and group acknowledgement transfer and the tear down of Group Ack mechanism which are discussed in detail in the following subclauses.

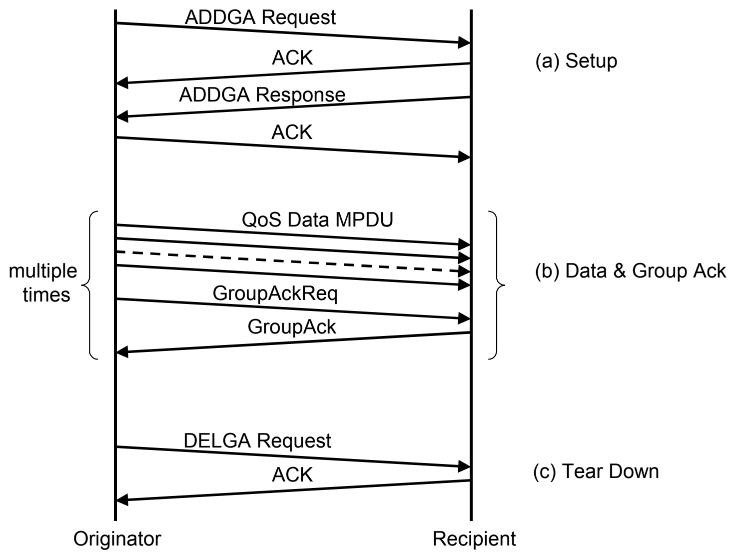


Figure 62.4 – Message Sequence Chart for Group Ack Mechanism: (a) Set up, (b) Data and acknowledgement transfer and (c) Tear down

9.11.2 Set up and modification of the Group Ack parameters

An originating QSTA that has data traffic to send and intends to use Group Ack facility mechanism should first check if the intended recipient QSTA is capable of participating in Group Ack mechanism by discovering and examining its "Group Ack" capability bit. If the recipient is capable of participating, the originator sends an ADDGA request QoS Action request frame indicating the TID. The receiving QSTA recipient shall respond by an ADDGA response QoS Action response frame. The receiving QSTA recipient has the option of accepting or rejecting the request. When the recipient QSTA accepts, it indicates the type of Group Ack and the number of buffers that it might allocate for the support of this Group. If the QSTA recipient rejects the request, then the originator shall not use the Group Ack mechanism and might use either the IEEE 802.11 acknowledgement or not rely on acknowledgements.

If the Group 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 Group Ack mechanism.

Once the Group Ack exchange has been set up, Data and acknowledgements are transferred following the procedure described in clause 9.11.3.

9.11.3 Data and acknowledgement transfer

After setting up for the Group exchange following the procedure in 9.11.2, the originator may transmit a Group of QoS data type frames separated by SIFS period, number not exceeding the Re-ordering buffer size subfield in the associated ADDGA response QoS action management frame. Each of the frames shall have the Ack policy subfield in the QoS Control set to “Group Acknowledgement”. When the originator is ready to receive an acknowledgement, it shall seek the acknowledgement by sending a GroupAckReq frame. The recipient shall maintain a Group Ack record for the group.

Subject to any constraints herein about permitted use of TXOP according to the channel access mechanism used: the originator can separate the Group and the GroupAckReq into separate TXOPs; the originator can split a Group across multiple TXOPs; the originator can sequence frames with different TIDs in the same TXOP; the originator can interleave MPDUs from with different TIDs within the same TXOP; the originator can sequence or interleave MPDUs for different RA within a TXOP.

The duration values of Data MPDUs and any Group Ack exchange transmitted within a polled TXOP shall follow the rules defined in 9.10.2.2.2.

The duration rules during an EDCF TXOP are as follows: the duration field of any data frames shall protect any following transmitted MPDU and its response MPDU if there is one. In this context “protect” means that the duration value causes the NAV to expire at the end of the protected MPDU.

The originator shall use the Group Ack Starting Sequence Control to signal the first MPDU for which an acknowledgement is expected. MPDUs in the recipient’s buffer with a sequence number that precedes the Starting Sequence number 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 release any buffers held by MPDUs of MSDUs thus indicated or that cannot subsequently be completed.

The recipient shall maintain a group acknowledgement record consisting of originator address, TID, and a bitmap of size of Re-ordering Buffer Size indexed by received MPDU sequence number, where the MPDU sequence number 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 Group Ack policy is used, the recipient shall respond to a GroupAckReq, with either a GroupAck frame or with an ACK or QoS CF-Ack frame. If the recipient sends the GroupAck frame, the originator updates its own record and retries any frames that are not acknowledged in the GroupAck frame, either in another burst or individually. If the recipient sends an ACK or a QoS CF-Ack frame, as long as there is sufficient time within the TXOP, the originator shall transmit a GroupAckReq and shall continue to seek the acknowledgement either in the same TXOP or in a subsequent TXOP.

If the delayed Group Ack policy is used, the recipient shall respond to GroupAckReq with either an ACK frame or a QoS (+) CF-Ack frame. The recipient shall then send its GroupAck response in a subsequently obtained TXOP. Once the contents of the GroupAck 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 the receipt of the GroupAck frame.

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

A typical Group Ack frame exchange sequence using the immediate Group Ack, for a single TID is shown in figure 62.5.

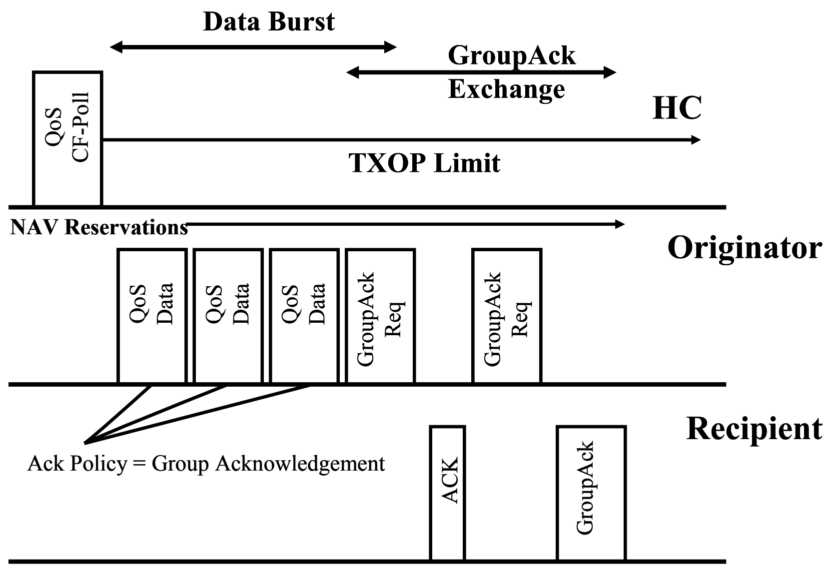


Figure 62.5 – A typical Group Ack sequence when immediate policy is used

A typical Group Ack sequence using the delayed Group Ack is shown in Fig 62.6.

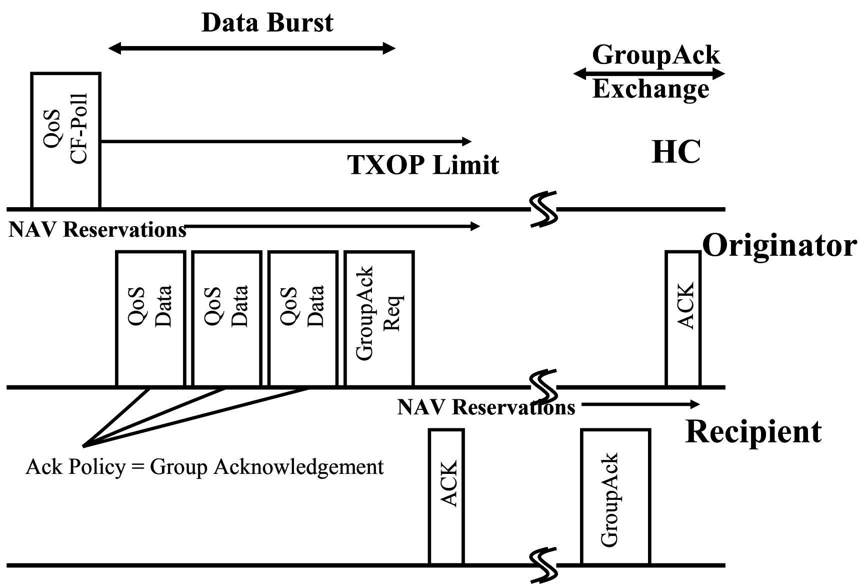


Figure 62.6 – A typical Group Ack sequence when delayed policy is used

If there is no response (i.e., neither a GroupAck nor an ACK) to the GroupAckReq frame, the originator can retransmit the GroupAckReq within the current TXOP (if time permits) or within a subsequent TXOP. This retransmission is subject to the dot11ShortRetryLimit times the number of MSDUs referenced by this GroupAckReq. If the GroupAckReq is discarded due to reaching this limit, all MPDUs in the group are considered to have failed transmission and are discarded.

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

In order to improve efficiency, originators using the Group 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 Group Ack. Originators, however, may not switch from immediate to delayed or delayed to immediate without renegotiation of the Group Ack parameters.

9.11.4 Receive buffer operation

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

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

All comparisons of sequence numbers are performed circularly modulo 2**12.

The Sent Count subfield in the Group ACK request message contains the number of MPDUs that should be present in the receive buffer starting with Group ACK starting sequence control. If the number of MPDUs present is equal to the Sent Count, then all of the complete MSDUs in the receive buffer shall be indicated to the MAC client using the MAC-UNIDATA.indication primitive.

If the receive buffer is not full and the number of received buffers does not match Sent Count, then any complete MSDU in the receive buffer that includes the Group ACK starting Sequence count shall be indicated to the MAC client using the MAC-UNIDATA.indication primitive. [24]

9.11.5 Tear down of the Group Ack mechanism

When the originator has no data to send it shall signal the end of its use of the Group Ack mechanism by sending the DELGA QoS Action Management frame to its recipient. There is no response from the recipient. [25]

9.11.6 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 a defined timeout. The duration of this timeout shall be the same as the Inactivity Interval if the data belongs to a TS and shall be dot11PeerLivenessTimeout if the data belongs to a TC. An originator failure is detected if there is no Data or GroupAckReq MPDU received from it using the TID for the duration of the timeout. A recipient failure is detected if there is no GroupAck 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 Delete Group Ack had been received.

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

[24] No other MSDUs can be indicated because the recipient cannot be sure that there are no missing subsequent MSDUs that may eventually be successfully retransmitted.

[25] Normal acknowledgement rules apply.

10. Layer management

10.3 MLME SAP Interface

Change 10.3.2 as shown below:

10.3.2 Scan

This mechanism supports the process of determining the characteristics of the available BSSs.

10.3.2.1 MLME-SCAN.request

10.3.2.1.1 Function

This primitive requests a survey of potential BSSs that the STA may later elect to try to join.

10.3.2.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-SCAN.request     (
                        BSSType,
                        BSSID,
                        SSID,
                        ScanType,
                        ProbeDelay,
                        ChannelList,
                        MinChannelTime,
                        MaxChannelTime
                        )
```

Name	Type	Valid Range	Description
BSSType	Enumeration	INFRASTRUCTURE, INDEPENDENT, ANY_BSS, ONLY_QOS	Determines whether Infrastructure BSS, Independent BSS, or both are included in the scan. These scans report QBSSs of the appropriate type. At QSTAs the ONLY_QOS value may be specified to limit the report to QBSSs.
BSSID	MACAddress	Any valid individual or broadcast MAC address	Identifies a specific or broadcast BSSID.
SSID	octet string	0 - 32 octets	Specifies the desired SSID or the broadcast SSID.
ScanType	Enumeration	ACTIVE, PASSIVE	Indicates either active or passive scanning.
ProbeDelay	integer	N/A	Delay (in μs) to be used prior to transmitting a Probe frame during active scanning
ChannelList	Ordered Set of Integer	Each channel will be selected from the valid channel range for the appropriate PHY and Carrier Set.	Specifies a list of channels which are examined when scanning for a BSS.

MinChannelTime	integer	greater than or equal to ProbeDelay	The minimum time (in TU) to spend on each channel when scanning
MaxChannelTime	integer	greater than or equal to MinChannelTime	The maximum time (in TU) to spend on each channel when scanning

10.3.2.1.3 When generated

This primitive is generated by the SME for a STA to determine if there are other BSSs that it may join.

10.3.2.1.4 Effect of receipt

This request initiates the scan process when the current frame exchange sequence is completed.

10.3.2.2 MLME-SCAN.confirm

10.3.2.2.1 Function

This primitive returns the descriptions of the set of BSSs detected by the scan process.

10.3.2.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-SCAN.confirm    (
                      BSSDescriptionSet,
                      ResultCode
                      )
```

Name	Type	Valid Range	Description
BSSDescriptionSet	Set of BSSDescription	N/A	The BSSDescriptionSet is returned to indicate the results of the scan request. It is a set containing zero or more instances of a BSSDescription.
ResultCode	enumeration	SUCCESS, INVALID_ PARAMETERS	Indicates the result of the MLME-SCAN.confirm.

Each BSSDescription consists of the following elements:

Name	Type	Valid Range	Description
BSSID	MACAddress	N/A	The BSSID of the found BSS
SSID	octet string	1 - 32 octets	The SSID of the found BSS
BSSType	Enumeration	INFRASTRUCTURE, INDEPENDENT	The type of the found BSS
Beacon Period	integer	N/A	The Beacon period of the found BSS (in TU)
DTIM Period	integer	As defined in Frame Format	The DTIM Period of the BSS (in Beacon Periods)
Timestamp	integer	N/A	The timestamp of the

			received frame (probe response/beacon) from the found BSS
Local Time	integer	N/A	The value of the station's TSF timer at the start of reception of the PHY symbol which contains the first bit of the first octet of the timestamp field of the received frame (probe response or beacon) from the found BSS.
PHY parameter set	As defined in Frame Format	As defined in Frame Format	The parameter set relevant to the PHY
CF parameter set	As defined in Frame Format	As defined in Frame Format	The parameter set for the CF periods, if found BSS supports CF mode.
IBSS parameter set	As defined in Frame Format	As defined in Frame Format	The parameter set for the IBSS, if found BSS is an IBSS.
CapabilityInformation	As defined in Frame Format	As defined in Frame Format	The advertised capabilities of the BSS.
BSSBasicRateSet	set of integers	1 through 127 inclusive (for each integer in the set)	The set of data rates (in units of 500kbit/s) that shall be supported by all STAs that desire to join this BSS. The STAs shall be able to receive at each of the data rates listed in the set.
QBSSLoad (QoS only)	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.
ExtendedCapabilities (QoS only)	As defined in Frame Format	As defined in Frame Format	The values from the Extended Capabilities information element if such an element was present in the probe response or beacon, else null.

10.3.2.2.3 When generated

This primitive is generated by the MLME as a result of an MLME-SCAN.request to ascertain the operating environment of the STA.

10.3.2.2.4 Effect of receipt

The SME is notified of the results of the scan procedure.

Change 10.3.6.1 as shown below:

10.3.6.1 MLME-ASSOCIATE.request

10.3.6.1.1 Function

This primitive requests association with a specified peer MAC entity that is acting as an AP.

10.3.6.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ASSOCIATE.request        (
                               PeerSTAAddress,
                               AssociateFailureTimeout,
                               CapabilityInformation,
                               ListenInterval,
                               ExtendedCapabilities
                               )
```

Name	Type	Valid Range	Description
PeerSTAAddress	MACAddress	Any valid individual MAC address	Specifies the address of the peer MAC entity with which to perform the association process.
AssociateFailureTimeout	integer	Greater than or equal to 1	Specifies a time limit (in TU) after which the associate procedure will be terminated.
CapabilityInformation	As defined in Frame Format	As defined in Frame Format	The operational capability definitions to be used by the MAC entity.
ListenInterval	Integer	Greater than or equal to 0	Specifies the number of Beacon intervals which may pass before the station awakens and listens for the next beacon.
ExtendedCapabilities (QoS only)	As defined in Frame Format	As defined in Frame Format	The extended operational capability definitions to be used by the QSTA.

10.3.6.1.3 When generated

This primitive is generated by the SME when a STA wishes to establish association with an AP.

10.3.6.1.4 Effect of receipt

This primitive initiates an association procedure. The MLME subsequently issues a MLME-ASSOCIATE.confirm that reflects the results.

Change 10.3.7.1 as shown below:

10.3.7.1 MLME-REASSOCIATE.request

10.3.7.1.1 Function

This primitive requests a change in association to a specified new peer MAC entity that is acting as an AP.

10.3.7.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-REASSOCIATE.request   (
                            NewAPAddress,
                            ReassociateFailureTimeout,
                            CapabilityInformation,
                            ListenInterval,
                            ExtendedCapabilities
                           )
```

Name	Type	Valid Range	Description
NewAPAddress	MACAddress	any valid individual MAC address	Specifies the address of the peer MAC entity with which to perform the reassociation process.
ReassociateFailureTimeout	integer	greater than or equal to 1	Specifies a time limit (in TU) after which the reassociate procedure will be terminated.
CapabilityInformation	As defined in Frame Format	As defined in Frame Format	The operational capability definitions to be used by the MAC entity.
ListenInterval	Integer	Greater than or equal to 0	Specifies the number of Beacon intervals which may pass before the station awakens and listens for the next beacon.
ExtendedCapabilities (QoS only)	As defined in Frame Format	As defined in Frame Format	The extended operational capability definitions to be used by the QSTA.

10.3.7.1.3 When generated

This primitive is generated by the SME when a STA wishes to change association to a specified new peer MAC entity that is acting as an AP.

10.3.7.1.4 Effect of receipt

This primitive initiates a reassociation procedure. The MLME subsequently issues a MLME-REASSOCIATE.confirm that reflects the results.

Replace 10.3.10.1 with the updated subclause below:

10.3.10.1 MLME-START.request

10.3.10.1.1 Function

This primitive requests that the MAC entity start a new BSS.

10.3.10.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-START.request          (
                             SSID,
                             BSSType,
                             BeaconPeriod,
                             DTIMPeriod,
                             CF parameter set,
                             PHY parameter set,
                             IBSS parameter set,
                             ProbeDelay.
                             CapabilityInformation,
                             BBSBasicRateSet,
                             OperationalRateSet,
                             ExtendedCapabilities,
                             QoS parameter set
                             )
```

Name	Type	Valid Range	Description
SSID	octet string	1 - 32 octets	The SSID of the BSS.
BSSType	Enumeration	INFRA-STRUCTURE, INDEPEN-DENT	The type of the BSS.
Beacon Period	integer	Greater than or equal to 1	The Beacon period of the BSS (in TU).
DTIM Period	integer	As defined in Frame Format	The DTIM Period of the BSS (in Beacon Periods)
CF parameter set	As defined in Frame Format	As defined in Frame Format	The parameter set for CF periods, if the BSS supports CF mode. aCFPPeriod is modified as a side effect of the issuance of a MLME-START.request primitive.
PHY parameter set	As defined in Frame Format	As defined in Frame Format	The parameter set relevant to the PHY.
IBSS parameter set	As defined in Frame Format	As defined in Frame Format	The parameter set for the IBSS, if BSS is an IBSS.
ProbeDelay	integer	N/A	Delay (in µs) to be used prior to transmitting a Probe frame during active scanning
CapabilityInformation	As defined in Frame Format	As defined in Frame Format	The capabilities to be advertised for the BSS.

BSSBasicRateSet	set of integers	1 through 127 inclusive (for each integer in the set)	The set of data rates (in units of 500 kbit/s) that shall be supported by all STAs that desire to join this BSS. The STA that is creating the BSS shall be able to receive at each of the data rates listed in the set.
OperationalRateSet	set of integers	1 through 127 inclusive (for each integer in the set)	The set of data rates (in units of 500 kbit/s) that the STA desires to use for communication within the BSS. The STA shall be able to receive at each of the data rates listed in the set. The OperationalRateSet is a superset of the BSSBasicRateSet advertised by the BSS.
ExtendedCapabilities	As defined in Frame Format	As defined in Frame Format	The extended operational capability definitions to be used by the QSTA.
QoS parameter set (QoS only)	As defined in Frame Format	As defined in Frame Format	The initial QoS parameter set values to be used in the QBSS.

10.3.10.1.3 When generated

This primitive is generated by the SME to start either an infrastructure BSS (with the MAC entity acting as an AP), or start an Independent BSS (with the MAC entity acting as the first STA in the IBSS).

The MLME-START.request primitive shall be generated after a MLME-RESET.request primitive has been used to reset the MAC entity and before an MLME-JOIN.request primitive has been used to successfully join an existing infrastructure BSS or Independent BSS.

The MLME-START.request primitive shall not be used after successful use of the MLME-START.request primitive or successful use of the MLME-JOIN.request without generating an intervening MLME-RESET.request primitive.

10.3.10.1.4 Effect of receipt

This primitive initiates the BSS initialization procedure once the current frame exchange sequence is complete. The MLME subsequently issues a MLME-START.confirm that reflects the results of the creation procedure.

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 11.5.

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 frame body. Requests and responses may cause these frames to be sent. Confirms and indications are emitted when an appropriate QoS action frame is received.

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

Table 22.1 – Supported QoS Management Primitives

Primitive	Request	Confirm	Indication	Response
ADDTS	non-AP STA	non-AP STA	HC	HC
DELTS	non-AP STA & HC	non-AP STA & HC	non-AP STA & 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,
                     TrafficSpecification,
                     TrafficClassification
                     )
```

Name	Type	Valid Range	Description
DialogToken	Integer	0-255	Specifies a number unique to the QoS management action primitives and frames used in adding (or modifying) the traffic stream of concern
TrafficSpecification	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 frame format 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 traffic stream of concern
TrafficClassification (optional)	Opaque object (maximum size of 256	Beyond the scope of this standard	Specifies the rules by which an MSDU may be classified

	octets)		

This primitive supports the creation of or modification of a traffic stream. The stream is defined by the parameters in the TSPEC and optional TCLAS parameters and identified by the TSID [26] and Direction fields within the TSPEC, selected by the non-AP QSTA SME.

This primitive results in an ADDTS QoS Action request frame being sent from the non-AP QSTA to the HC. The MLME-ADDTS.confirm contains the HC's response.

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 the procedures defined in 11.5.

10.3.11.2 MLME-ADDTS.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-ADDTS.confirm  (
                    ResultCode,
                    DialogToken,
                    TrafficSpecification,
                    Schedule Element,
                    TrafficClassification
                                          )
```

Name	Type	Valid Range	Description
ResultCode	Enumeration	SUCCESS, INVALID-PARAMETERS, ALTERNATIVE, REFUSED, TIMEOUT	Indicates the results of the corresponding MLME-ADDTS.request
DialogToken	Integer	As defined in the corresponding MLME-ADDTS.request	Specifies a number unique to the QoS management action primitives and frames used in adding (or modifying) the TS
TrafficSpecification	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

[26] When the non-AP STA SME is intending to create a new TS, it is recommended that the SME allocate new TSIDs from among those unused in a circular fashion, thus avoiding any recently used TSIDs.

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
TrafficClassification (optional)	Opaque object (maximum size of 256 octets)	As defined in the corresponding MLME-ADDTS.request	Specifies the rules by which an MSDU may be classified

For ResultCode values of SUCCESS and ALTERNATIVE, 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.

In the case of ALTERNATIVE, they represent an alternative proposal by the HC. A TS is created with this definition. If the alternative is not acceptable to the non-AP QSTA, it is the responsibility of the non-AP QSTA to delete the unwanted TS.

For other ResultCodes, no TS has been created. If this is the result of a modification of an existing TS, the status of that TS is undefined and the TS shall not be used by the non-AP QSTA. The non-AP QSTA should attempt to delete the TS and recreate it if necessary.

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 QoS Action frame 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 the procedures defined in 11.5. In the case of and ALTERNATIVE ResultCode, if the alternative is not acceptable to the SME, it is the responsibility of the SME to delete the unwanted TS.

In the case of other failure ResultCode values, if this is the result of a modification of an existing TS, the status of that TS is undefined and the TS shall not be used by the non-AP QSTA. The SME should attempt to delete the TS and recreate it if necessary.

10.3.11.3 MLME-ADDTS.indication

10.3.11.3.1 Function

This primitive reports the initiation of adding (or modifying) a traffic stream to the HC SME.

10.3.11.3.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDTS.indication        (
                              DialogToken,
                              non-AP QSTA Address
```

TrafficSpecification,
TrafficClassification
)

Name	Type	Valid Range	Description
DialogToken	Integer	As defined in the received Add TS QoS Action frame	Specifies a number unique to the QoS management action primitives and frames used in adding (or modifying) the TS
non-AP QSTA Address	MAC Address		Contains the MAC address of the non-AP QSTA that initiated the MLME-ADDTS.request
TrafficSpecification	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 Add TS QoS Action 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
TrafficClassification (optional)	Opaque object (maximum size of 256 octets)	As defined in the received Add TS QoS Action frame	Specifies the rules by which an MSDU may be classified

The TrafficSpecification and optional TrafficClassification define the QoS parameters of the requested TS. The TS is uniquely identified in the HC by a combination of the MPDU SA field, and the TSID and Direction fields within the TSPEC.

The TrafficClassification is optional at the discretion of the non-AP QSTA that originated the request. An HC shall be capable of receiving an ADDTS QoS Action frame that contains a TrafficClassification element and generating an indication that contains this parameter.

10.3.11.3.3 When generated

This primitive is generated by the MLME as a result of receipt of an initiation to add (or modify) a traffic stream by a specified non-AP QSTA in the form of an Add TS QoS Action frame.

10.3.11.3.4 Effect of receipt

The SME is notified of the initiation of 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 the procedures defined in 11.5.

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 initiation of 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,
                        Non-AP QSTAAddress
                        TrafficSpecification,
                        Schedule Element
                        TrafficClassification
                        )
```

Name	Type	Valid Range	Description
ResultCode	Enumeration	SUCCESS, INVALID-PARAMETERS, ALTERNATE, REFUSED	Indicates the results of the corresponding MLME-ADDTS.indication
DialogToken	Integer	As defined in the corresponding MLME-ADDTS.indication	DialogToken of the matching MLME-ADDTS.indication
Non-AP QSTA Address	MAC Address		non-AP QSTA Address of the matching MLME-ADDTS.indication
TrafficSpecification	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
TrafficClassification (optional)	Opaque object (maximum size of 256 octets)	As defined in the corresponding MLME-ADDTS.indication	Specifies the rules by which an MSDU may be classified

The DialogToken and non-AP QSTA Address parameters shall contain the value from the matching MLME-ADDTS.indication.

If the ResultCode is SUCCESS, the TrafficSpecification and (optional) TrafficClassifiation parameters shall contain the values from the matching MLME-ADDTS-indication.

If the ResultCode is ALTERNATIVE, the TrafficSpecification and TrafficClassification parameters represent an alternative proposed TS. The TSID and Direction within the TSPEC shall be as in the matching indication. The difference may lie in the QoS (e.g., minimum data rate, mean data rate, delay bound, and jitter 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 QoS Action 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 the deletion of a traffic stream with a specified peer MAC.

This primitive may be generated at either the non-AP QSTA or HC.

10.3.11.5.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DELTS.request        (
                          DialogToken,
                          non-AP QSTAAddress
                          TrafficSpecification
                          )
```

Name	Type	Valid Range	Description
DialogToken	Integer	0-255	Specifies a number unique to the QoS management action primitives and frames used in deleting the TS
non-AP QSTA Address (HC only)	MAC Address		(At the HC only) Specifies the MAC address of the non-AP QSTA that initiated this TS.
TrafficSpecification	As defined in frame format	As defined in frame format	The TSID and Direction fields within the traffic specification specify the TS to be deleted. All other fields are undefined.

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 QoS Action 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 QoS Action 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 results of a traffic stream deletion attempt with a specified peer MAC entity or entities.

10.3.11.6.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DELTS.confirm          (
                            ResultCode,
                            DialogToken,
                            non-AP QSTAAddress,
                            TrafficSpecification
                            )
```

Name	Type	Valid Range	Description
ResultCode	Enumeration	SUCCESS, INVALID-PARAMETERS, FAILURE	Indicates the results of the corresponding MLME-DELTS.request
DialogToken	Integer	As defined in the corresponding MLME-DELTS.request	Contains the DialogToken of the matching request
non-AP QSTA Address (HC only)	MAC address		(HC only) contains the non-AP QSTA MAC address of the matching request.
TrafficSpecification	As defined in frame format	As defined in the corresponding MLME-DELTS.request	Contains the traffic specification 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 QoS Action 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:

```
MLME-DELTS.indication(
                        DialogToken,
                        non-AP QSTAAddress
                        TrafficSpecification
                        )
```

Name	Type	Valid Range	Description
DialogToken	Integer	As defined in the received Add TS QoS Action frame	Specifies a number unique to the QoS management action primitives and frames used deleting the TS
Reason	Enumeration	Requested and Timeout values	Indicates the reason why the TS is being deleted.
non-AP QSTA Address (HC only)	MAC address		(HC only) The MAC address of the non-AP QSTA for which the TSPEC is being deleted.
TrafficSpecification	As defined in frame format	As defined in the received Del TS QoS Action frame	Specifies the source address, destination address, TSInfo, traffic characteristics and QoS requirements of the traffic stream of concern.

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 Medium Status update

The following primitive describes how the medium status is determined.

10.3.12.1 MLME-WMSTATUS.request

10.3.12.1.1 Function

This primitive requests the information on the state of the wireless medium.

10.3.12.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-WMSTATUS.request      ( )
```

10.3.12.1.3 When generated

This primitive is generated by the SME at a QSTA when a higher-layer QoS management entity wishes to obtain information on the state of the wireless medium.

10.3.12.1.4 Effect of receipt

This primitive causes generation of a MLME-WMSTATUS.confirm that reports on the state of the wireless medium.

10.3.12.2 MLME-WMSTATUS.confirm

10.3.12.2.1 Function

This primitive reports the results of the preceding MLME-WMSTATUS.request.

10.3.12.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-WMSTATUS.confirm      (
                             ResultCode,
                             << additional parameter values >>
                             )
```

Name	Type	Valid Range	Description
ResultCode	enumeration	SUCCESS, NOT_AVAILABLE	Indicates the result of the MLME-WMSTATUS.request.

10.3.12.2.3 When generated

This primitive is generated by the MLME as a result of an MLME-WMSTATUS.request to report on the state of the wireless medium.

10.3.12.2.4 Effect of receipt

The SME is notified of the state of the wireless medium.

10.3.13 Management of Direct Links

This clause describes the management procedures associated with Direct Links. The following DLP management primitives are defined:

- MLME-DLP.request
- MLME-DLP.indication
- MLME-DLP.confirm

The extended DLP message flow is illustrated in the following figure 63.1:

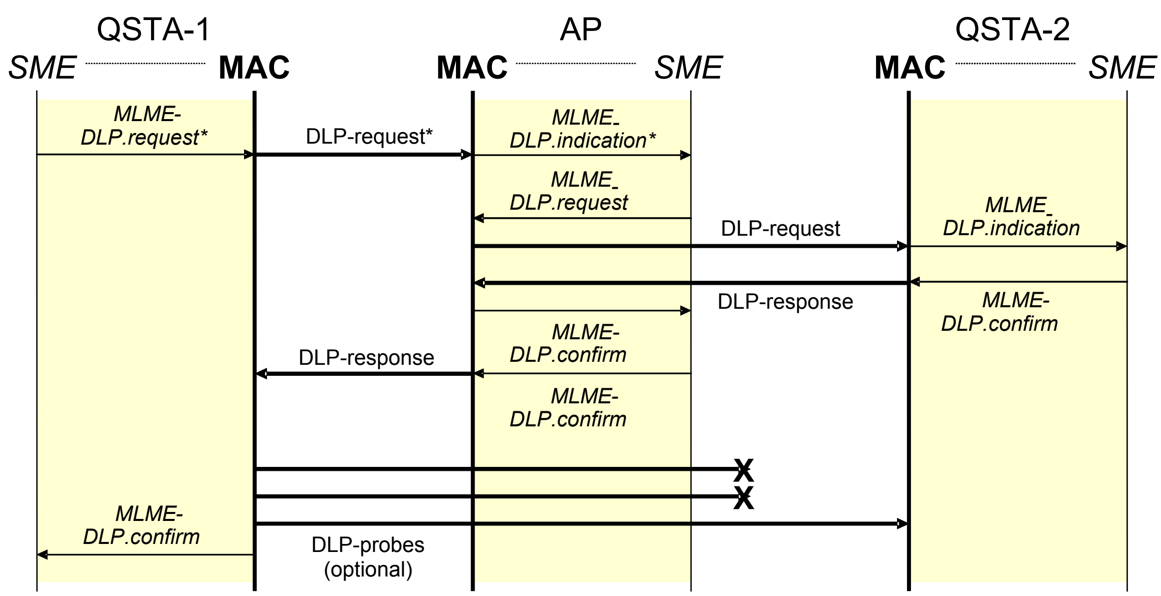


Figure 63.1 – Direct Link Protocol message flow. (Messages indicated with an asterisk are omitted when the AP is the initiator of the Direct Link.)

10.3.13.1 MLME-DLP.request

10.3.13.1.1 Function

This primitive requests that the MAC entity set up a direct link between two QSTAs.

10.3.13.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DLP.request        (
                        Sender MAC Address
                        Receiver MAC Address
                        )
```

Name	Type	Valid Range	Description
Sender MAC Address	MACAddress	Any valid individual MAC address	Specifies the MAC address of the non-AP QSTA that is the sender of the data flow. The initiating non-AP QSTA shall use the own MAC address. The AP shall use the address which it received in the MLME-DLP.indication, or the address of the intended sender.
Receiver MAC Address	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.

10.3.13.1.3 When generated

This primitive is generated by the SME at a QSTA that wishes to set up a Direct Link to another non-AP QSTA.

10.3.13.1.4 Effect of receipt

This request initiates the DLP handshake.

10.3.13.2 MLME-DLP.indication

10.3.13.2.1 Function

This primitive indicates the reception of a DLP-request frame.

10.3.13.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DLP.indication     (
                        Sender MAC Address
                        Receiver MAC Address
                        )
```

Name	Type	Valid Range	Description
Sender MAC Address	MACAddress	Any valid individual MAC address	Specifies the MAC address of the non-AP QSTA that is the sender of the data flow.
Receiver MAC Address	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.

10.3.13.2.3 When generated

This primitive is generated when the MAC received a DLP-request.

10.3.13.2.4 Effect of receipt

This primitive informs the receiver about the pending Direct Link handshake.

10.3.13.3 MLME-DLP.confirm

10.3.13.3.1 Function

This primitive informs the node about the decision that was taken regarding the pending Direct Link request.

10.3.13.3.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DLP.confirm        (
                        Sender MAC Address
                        Receiver MAC Address
                        Result Code
                        )
```

Name	Type	Valid Range	Description
Sender MAC Address	MACAddress	Any valid individual MAC address	Specifies the MAC address of the non-AP QSTA that is the sender of the data flow.
Receiver MAC Address	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.
Result Code	Enumeration	SUCCESS, INVALID_PARAMETERS, NOT_ALLOWED, NOT_PRESENT, NOT_QSTA, REFUSED	Specifies the MAC address of the non-AP QSTA that is the intended immediate recipient of the data flow.

10.3.13.3.3 When generated

This primitive is generated by the MLME after a decision was taken regarding the pending Direct Link request.

10.3.13.3.4 Effect of receipt

If the AP receives a positive confirmation, it will send a confirmation to the sender non-AP QSTA, QSTA-1. At the sender, the confirmation indicates the point at which the Direct Link becomes active.

10.3.15 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 proposal. In principle, the MLME indicates the transmission/reception of frames with a specific multicast address in the Address 1 field of the MAC data type frame.

10.3.15.1 MLME-HL-SYNC.request

10.3.15.1.1 Function

This primitive requests activating the synchronization support mechanism.

10.3.15.1.2 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.

10.3.15.1.3 When generated

This primitive is generated by the SME when a higher layer protocol initiates a synchronization process.

10.3.15.1.4 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 MMLE-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.

10.3.15.2 MLME-HL-SYNC.confirm

10.3.15.2.1 Function

This primitive confirms the activation of the higher layer synchronization support mechanism.

10.3.15.2.2 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.

10.3.15.2.3 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.

10.3.15.2.4 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.

10.3.15.3 MLME-HL-SYNC.indication

10.3.15.3.1 Function

This primitive indicates the last symbol on air of a higher layer synchronization frame, whether transmitted or received by the MAC.

10.3.15.3.2 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.

10.3.15.3.3 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.

10.3.15.3.4 Effect of Receipt

The SME is notified of the reception or transmission of a higher layer synchronization frame.

10.3.16 Group Ack

This mechanism supports the initiation (or modification) and termination of Group Ack.

The primitives used for this mechanism are called Group Ack primitives, which include MLME-ADDGA.xxx and MLME-DELGA.xxx primitives, where xxx denotes request, confirm, indication, or response. They each contain the frame body, starting with the Dialog Token field, of the corresponding QoS Management Action frame, i.e., ADDGA QoS Action request and response frames and DELGA QoS Action request frame, as their parameters.

10.3.16.1 MLME-ADDGA.request

10.3.16.1.1 Function

This primitive requests the initiation (or modification) of Group Ack with a peer MAC entity.

10.3.16.1.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDGA.request (
                    PeerQSTAAddress,
                    Dialog token,
                    TID,
                    Transmit Buffer Size
                    )
```

Name	Type	Valid Range	Description
PeerQSTAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity with which to perform the Group Ack initiation (or modification).
Dialog Token	Integer	0-255	Specifies a number unique to the QoS management action primitives and frames used in defining or deleting a Group Ack.
TID	Integer	0-15	Specifies the TID of the data.
Transmit Buffer Size	Integer	0-128	Specifies the number of MSDUs that can be held in its buffer.

10.3.16.1.3 When generated

This primitive is generated by the SME at a QSTA to request initiation (or modification) of Group Ack with the specified peer MAC entity.

10.3.16.1.4 Effect of receipt

The QSTA shall send the ADDGA request QoS Action frame to the specified peer MAC entity.

10.3.16.2 MLME-ADDGA.confirm

10.3.16.2.1 Function

The primitive reports the results of initiation (or modification) of the Group Ack attempt with the specified peer MAC entity

10.3.16.2.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDGA.confirm (
                    PeerQSTAAddress,
                    Dialog token,
                    TID,
                    ResultCode,
                    Group Ack Policy,
                    Re-ordering Buffer Size
                    )
```

Name	Type	Valid Range	Description
PeerQSTAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity with which the Group Ack initiation (or modification)

			was attempted. This value must match the PeerQSTAAddress parameter specified in MLME-ADDGA.request.
Dialog Token	Integer	0-255	Specifies a number unique to the QoS management action primitives and frames used in defining or deleting a Group Ack. This value must match the Dialog Token parameter specified in MLME-ADDGA.request.
TID	Integer	0-15	Specifies the TID of the data. This value must match the TID specified in MLME-ADDGA.request
ResultCode	Enumeration	SUCCESS, REFUSED, TIMEOUT	Indicates the result of the corresponding MLME-ADDGA.request.
Group Ack Policy	Enumeration	Immediate, Delayed	Specifies the Group Ack Policy
Re-ordering Buffer Size	Integer	0-128	Specifies the number of MSDUs that can be grouped for the specified TID.

10.3.16.2.3 When generated

This primitive is generated by the MLME as a result of an MLME-ADDGA.request to indicate the results of that request.

10.3.16.2.4 Effect of receipt

The SME is notified of the results of the Group Ack initiation (or modification).

10.3.16.3 MLME-ADDGA.indication

10.3.16.3.1 Function

This primitive reports the indication of initiation (or modification) of Group Ack with a peer MAC entity.

10.3.16.3.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDGA.indication        (
                              PeerQSTAAddress,
                              Dialog token,
                              TID,
                              Transmit Buffer Size
                              )
```

Name	Type	Valid Range	Description
PeerQSTAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity with which to perform the Group Ack initiation (or modification).
Dialog Token	Integer	0-255	Specifies a number unique to the QoS management action primitives and frames used in defining or deleting a Group Ack.
TID	Integer	0-15	Specifies the TID of the data.
Transmit Buffer Size	Integer	0-128	Specifies the number of MSDUs that can be

			held in its buffer.

10.3.16.3.3 When generated

This primitive is generated by the MLME as a result of receipt of a Group Ack initiation (or modification) by the specified peer MAC entity in the form of an ADDGA request QoS Action frame.

10.3.16.3.4 Effect of receipt

The SME is notified of the initiation (or modification) of the Group Ack by the specified peer MAC entity.

10.3.16.4 MLME-ADDGA.response

10.3.16.4.1 Function

The primitive responds to the initiation (or modification) by a specified peer MAC entity.

10.3.16.4.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-ADDGA.response(
                    PeerQSTAAddress,
                    Dialog token,
                    TID,
                    ResultCode,
                    Group Ack Policy,
                    Re-ordering Buffer Size
                    )
```

Name	Type	Valid Range	Description
PeerQSTAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity which attempted the Group Ack initiation (or modification). This value must match the PeerQSTAAddress parameter specified in MLME-ADDGA.indication.
Dialog Token	Integer	0-255	Specifies a number unique to the QoS management action primitives and frames used in defining or deleting a Group Ack. This value must match the Dialog Token parameter specified in MLME-ADDGA.indication.
TID	Integer	0-15	Specifies the TID of the data. This value must match the TID specified in MLME-ADDGA.indication.
ResultCode	Enumeration	SUCCESS, REFUSED, INVALID PARAMETERS, TIMEOUT	Indicates the result of the corresponding MLME-ADDGA.indication.
Group Ack Policy	Enumeration	Immediate, Delayed	Specifies the Group Ack Policy. Undefined when ResultCode is REFUSED.
Re-ordering Buffer Size	Integer	0-128	Specifies the number of MSDUs that can be Group for the specified TID. Undefined when ResultCode is REFUSED.

10.3.16.4.3 When generated

This primitive is generated by the MLME as a result of an MLME-ADDGA.indication to initiate Group Ack with the specified peer MAC entity.

10.3.16.4.4 Effect of receipt

The primitive causes the MAC entity to send an ADDGA response QoS Action frame to the peer MAC entity.

10.3.16.5 MLME-DELGA.request

10.3.16.5.1 Function

This primitive requests the deletion of Group Ack with a peer MAC entity.

10.3.16.5.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DELGA.request   (
                     PeerQSTAAddress,
                     Dialog token,
                     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 Group Ack deletion.
Dialog Token	Integer	0-255	Specifies a number unique to the QoS management action primitives and frames used in defining or deleting a Group Ack.
Direction	Enumeration	Originator, recipient	Specifies if the MAC entity is the originator or the recipient of the data stream that uses the Group Ack.
TID	Integer	0-15	Specifies the TID of the data.

10.3.16.5.3 When generated

This primitive is generated by the SME at a QSTA to request deletion of Group Ack with the specified peer MAC entity.

10.3.16.5.4 Effect of receipt

The QSTA shall send the DELGA request QoS Action frame to the specified peer MAC entity.

10.3.16.6 MLME-DELGA.confirm

10.3.16.6.1 Function

The primitive reports the results of the Group Ack deletion attempt with a specified peer MAC entity.

10.3.16.6.2 Semantics of the service primitive

The primitive parameters are as follows:

```
MLME-DELGA.confirm   (
```

PeerQSTAAddress,
Dialog token,
Direction,
TID,
ResultCode,
)

Name	Type	Valid Range	Description
PeerQSTAAddress	MAC Address	N/A	Specifies the address of the peer MAC entity with which the Group Ack initiation (or modification) was attempted. This value must match the PeerQSTAAddress parameter specified in MLME-DELGA.request.
Dialog Token	Integer	0-255	Specifies a number unique to the QoS management action primitives and frames used in defining or deleting a Group Ack. This value must match the Dialog Token parameter specified in MLME-DELGA.request.
Direction	Enumeration	Originator, recipient	Specifies if the MAC entity is the originator or the recipient of the data stream that uses the Group Ack.
TID	Integer	0-15	Specifies the TID of the data. This value must match the TID specified in MLME-DELGA.request
ResultCode	Enumeration	SUCCESS, INVALID-PARAMETERS, FAILURE	Indicates the result of the corresponding MLME-DELGA.request.

10.3.16.6.3 When generated

This primitive is generated by the MLME as a result of an MLME-DELGA.request to indicate the results of that request.

10.3.16.6.4 Effect of receipt

The SME is notified of the results of the Group Ack deletion.

10.3.16.7 MLME-DELGA.indication

10.3.16.7.1 Function

This primitive reports the indication of deletion of Group Ack with a peer MAC entity.

10.3.16.7.2 Semantics of the service primitive

The primitive parameters are as follows:

MLME-DELGA.indication (
PeerQSTAAddress,
Dialog token,
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 Group Ack deletion.
Dialog Token	Integer	0-255	Specifies a number unique to the QoS management action primitives and frames used in defining or deleting a Group Ack.
Direction	Enumeration	Originator, Recipient	Specifies if the MAC entity is the originator or the recipient of the data stream that uses the Group Ack.
TID	Integer	0-15	Specifies the TID of the data.

10.3.16.7.3 When generated

This primitive is generated by the MLME as a result of receipt of a Group Ack deletion by the specified peer MAC entity in the form of a Delete Group Ack request QoS Action frame.

10.3.16.7.4 Effect of receipt

The SME is notified of the deletion of the Group Ack by the specified peer MAC entity.

10.3.17 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.

10.3.17.1 MLME-SCHEDULE.request

10.3.17.1.1 Function

This primitive requests transmission of a Schedule QoS Action frame. It is valid at the HC.

10.3.17.4.2 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 QoS Action 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

10.3.17.4.3 When generated

This primitive is generated by the SME at the HC to send the Schedule information, in the form of a Schedule QoS Action frame, to a specified non-AP QSTA when the Schedule Information for the non-AP QSTA is changed.

10.3.17.4.4 Effect of receipt

This primitive shall cause the MAC entity at the HC to send a Schedule QoS Action frame to the non-AP QSTA specified in the primitive containing the specified Schedule parameters.

10.3.17.2 MLME-SCHEDULE.confirm

10.3.17.2.1 Function

This primitive reports the results of a MLME-SCHEDULE.request.

10.3.17.2.2 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

10.3.17.2.3 When generated

This primitive is generated by the MLME as a result of an MLME-SCHEDULE.request when the action completes.

10.3.17.2.4 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 QoS Action Frame.

10.3.17.3 MLME-SCHEDULE.indication

10.3.17.3.1 Function

This primitive reports the reception of a new Schedule by the non-AP QSTA in the form of a Schedule QoS Action frame. It is valid at the non-AP QSTA.

10.3.17.3.2 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

10.3.17.3.3 When generated

This primitive is generated by the MLME as a result of receipt of a new Schedule in the form of a Schedule QoS Action frame.

10.3.17.3.4 Effect of receipt

The SME is notified of the receipt of QoS Schedule in the form of a Schedule QoS Action 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 subclause after subclause 11.2.1.9

11.2.1.10 Power management for STAs with TSPEC

STAs with admitted TSPECs can go into power-save after they are serviced, for a duration of minimum service interval indicated by the HC in the schedule element. At the end of a duration of minimum service interval, the STA shall be awake to receive any data frames from the AP as well as to receive the QoS polls from the HC for its own transmissions.

A STA that receives a frame from the HC with the Schedule Pending field in the QoS control field set to 1, shall remain in the awake state after the TXOP is finished to receive the Schedule QoS Action frame from the HC. If the Schedule QoS Action frame is not received within dot11ScheduleTimeout after the end of the TXOP, the STA may go into power save mode.

Insert the following clause after subclause 11.2.2:

11.2.3 Automatic Power-Save Delivery in a QBSS

QAPs capable of supporting automatic power-save delivery mode shall signal this capability through the use of the automatic power-save delivery extended capability bit.

Non-AP QSTAs operating in a QBSS wishing to utilize the automatic power-save delivery mechanism shall inform the AP of this fact by using an automatic power-save delivery information element, signaled through a (re)association or action management frame. The QAP shall not arbitrarily transmit MSDUs to non-AP QSTAs operating in an automatic power-save delivery (APSD) mode, but shall buffer MSDUs and only transmit them at beacon intervals that correspond to the specified wakeup period.

A non-AP QSTA can signal its desire to utilize APSD as the power-save mode delivery method through the use of a Automatic Power-Save Delivery element contained in a (re)association request or action frame. non-

AP QSTAs use the power-save mode bit in the frame control field of a frame to indicate whether it is in active or power-save mode. 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 utilize the activation delay field of the APSD QoS action frame to delay the activation of APSD mode.

Automatic power-save delivery can be disabled by the non-AP QSTA by setting the wakeup period to zero in an action request frame or (re)association frame. Additionally, a non-AP QSTA may disable automatic power-save delivery by (re)associating with the QAP, and not including an Automatic Power-Save Delivery element.

11.2.3.1 Transmit operation at the QAP

QAPs shall maintain an automatic power-save delivery (APSD) status for each currently associated non-AP QSTA that indicates whether the non-AP QSTA is presently in APSD mode and the wakeup period for the non-AP QSTA. A QAP shall, based on the APSD mode of the non-AP QSTA, temporarily buffer the MSDU or management frames destined to the non-AP QSTA. MSDUs and management frames received for non-AP QSTAs not operating in APSD mode shall follow the appropriate frame delivery rules as related to their respective power-saving mode.

a) 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 the QAP must preserve the order of arrival of frames on a per priority basis, including during transitions between active and APSD modes.
b) MSDUs, or management frames, destined for APSD capable non-AP QSTAs not operating in APSD shall be transmitted according to the rules in this standard that correspond to the power-save state of the non-AP QSTA.
c) At every beacon interval, the QAP shall assemble the partial virtual bitmap containing the buffer status per destination for non-AP QSTAs in the APSD mode, and shall send this out in the TIM field of the beacon.
d) At every beacon interval the QAP shall determine, based on the wakeup period specified by the non-AP QSTA, during (re)association or through a management action frame, whether this beacon corresponds to an APSD beacon for each non-AP QSTA. If this is determined to be an APSD beacon, the QAP shall transmit all frames destined for the non-AP QSTA. The More Data field of each directed data or management frame, except the last frame, shall be set to indicate the presence of multiple frames destined for the non-AP QSTA. If necessary the QAP can generate an extra (QoS)-Null frame with the more data field cleared. When the QAP has transmitted a directed frame with “more data” zero during the APSD beacon interval, it shall not transmit any more frames using this mechanism until the next APSD beacon.
e) A QAP shall have an aging function to delete pending traffic when it is buffered for an excessive time period. Aging may be based on the listen interval specified by the non-AP QSTA in the (re)association request.
f) Whenever a QAP is informed that a APSD capable non-AP QSTA changes it’s delivery mode to not be APSD mode, then the QAP shall send buffered MSDUs and management frames (if any exist) to that non-AP QSTA according to the rules corresponding to the current power-save state of the non-AP QSTA.

11.2.3.2 Receive operation for non-AP QSTAs in the APSD mode

A non-AP QSTA operating in APSD mode shall operate as follows to receive an MSDU or management frame from the QAP:

a) non-AP QSTAs shall wake up early enough to receive the next beacon that corresponds to the wakeup period specified in the (re)association or action request frame the non-AP QSTA used to initiate APSD mode with the QAP.

b) When a non-AP QSTA detects that the bit corresponding to its AID is set in the TIM, the non-AP QSTA shall remain awake until it receives a directed MSDU or management frame with the More Data field cleared, or it receives a beacon with its TIM bit cleared.
c) A non-AP QSTA that misses it scheduled wakeup beacon shall remain awake until it receives a beacon with its TIM bit cleared, or a data frame with the More Data field set off.
d) A non-AP QSTA operating in APSD mode may send a PS-Poll frame at any time to retrieve a single frame buffered at the QAP.
e) When a non-AP QSTA transitions from active mode to APSD mode it shall remain awake until it receives a beacon with its TIM bit cleared.

Insert the following sections after section 11.3.4 and renumber 11.4 as 11.6:

11.4 Traffic Stream operation

11.4.1 Introduction (Informative)

A TSPEC describes the QoS characteristics of a traffic stream. 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 TSPEC is transported on the air by the ADDTS and DELTS QoS Action frames and across the MLME SAP by the MLME-ADDTS and MLME-DELTS 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.

It is always the responsibility of the non-AP QSTA to initiate the creation of a TS regardless of its direction.

In the direct link case, it is the responsibility of the non-AP QSTA that is going to send the Data to create the TS. In this case, the non-AP QSTA negotiates with the HC to gain TXOPs that it uses to send the Data. There is no negotiation between the originator and recipient non-AP QSTAs concerning the TS – the originator can discover the capabilities of the recipient (rates, GroupAck) using the DLP protocol. It does not negotiate the use of these on a per-TS basis with the intended recipient.

In the case of traffic relayed by a QAP, sending and receiving non-AP QSTA may both create individual TS for the traffic. Any traffic classifier created for the downlink TS applies equally regardless of whether the source is in the same BSS or reached through the DS.

11.4.2 TSPEC Construction (informative)

TSPECs are constructed within the MLME, 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 section 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 section describes how the parameters that are typically generated within the MAC may be derived.

Note that 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 shall be terminated.

Typically, then TSPEC elements not determined by the application are built upon the assumptions that there is:

- A probability *p* of not transmitting the frame (because it would have exceeded its Delay Bound)
- An MSDU length (which is 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. Note that for multiple streams, this Service Interval should be the aggregate of all service intervals requested, since the STA is assigned TXOPs, not any particular stream.

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 STA is assigned TXOPs, not any particular stream.

Typically, we can assume that the scheduler would attempt to schedule TXOPs uniformly 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 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 makes errors independently from frame to frame, and that 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 being 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.

To understand how the Surplus Bandwidth Allowance may be specified, consider the following:

Suppose one wished only to send 100 PPDUs, and that that delay was 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 PPDUS by $N_{excesss}$ 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 will yield 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 MPDUs 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 $1.6X10^{-15}$.

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 retires, and with a 10% PER, 1.111 times the bandwidth required to send MPDUs 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= 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 $N_{allocated}$ 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 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 spacing, polls, etc.) then determines the Surplus Bandwidth allowance.

In general, then the Surplus Bandwidth Allowance may be given by: $\frac{N_{allocated}}{N_{payload}}$.

Note that for the case of a Group Acknowledgement a similar calculation applies, although the calculations for the excess bandwidth need to take into account the probability of failing to receive a Group ACK, and so forth.

11.4.2 TS lifecycle

Figure 68.1 summarizes the TS lifecycle. (Using the HMSC syntax defined in ITU Z.120).

Initially a TS is inactive. A QSTA shall not transmit any QoS Data MPDUs using an inactive TS.

Following a successful TS Setup initiated by the non-AP QSTA, the TS becomes active, and either the non-AP QSTA or HC may transmit MPDUs using this TSID (according to the Direction field).

While the TS is active, the parameters of the TSPEC characterizing the TS can be re-negotiated, initiated by the non-AP QSTA. This negotiation can succeed – resulting in a change to the TSPEC, or can fail, resulting in no change to the TSPEC.

An active TS becomes inactive following a TS deletion process initiated at either non-AP QSTA or HC. It also becomes inactive following a TS timeout detected at the HC.

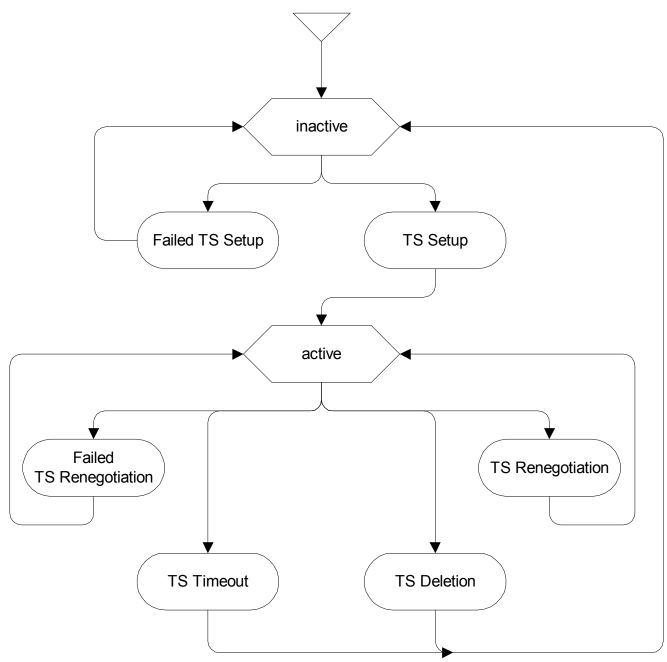


Figure 68.1 – TS Lifecycle

11.4.3 TS setup

Figure 68.2 shows the sequence of messages occurring at a TS setup.

The non-AP QSTA SME decides that a TS needs to be created. How it does this, and how it selects the TSPEC parameters are 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 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 shall be terminated.

The non-AP QSTA MAC transmits the TSPEC in an ADDTS QoS action request to the HC and starts a response timer called ADDTS timer of duration dot11ADDTSResponseTimout.

The HC MAC receives this MPDU 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, admit the TSPEC with a counter proposal or refuse the TSPEC and generates an MLME-ADDTS.response primitive containing the TSPEC and a status value. The contents of the TSPEC and status field contain values specified in table 23.2.

Table 23.2 – TSPEC and Status field contents in the MLME-ADDTS.response

Condition	TSPEC Contents	Status
HC SME grants requested TXOP	Exactly as the requested TXOP	Success
HC SME grants an altered TXOP	TSID and Direction field the same as the requested TXOP. Other fields can be modified	ALTERNATIVE
HC SME refuses TXOP	Exactly as the requested TXOP	REFUSED

The HC MAC transmits an ADDTS QoS action response containing this TSPEC and status.

The non-AP QSTA MAC receives this MPDU 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. In the "OK" and "ALTERNATIVE_GRANT" Status cases, the TS is in the active state. If an alternative grant is acceptable, the setup procedure ends here. 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. If the non-AP QSTA SME decides to terminate and that an ALTERNATIVE is inadequate, it is the responsibility of the non-AP QSTA SME to destroy the TS using the TS Deletion procedure.

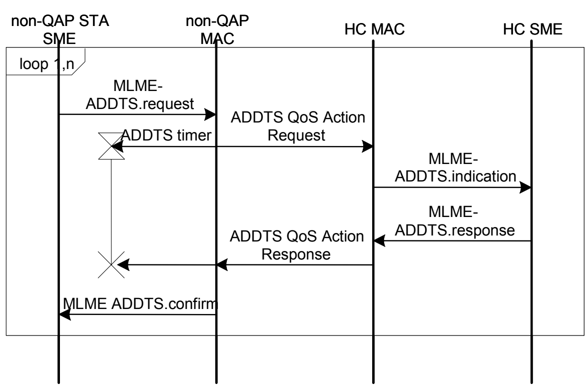


Figure 68.2 – TS Setup

11.4.4 Failed TS setup

There are three possible types of Failed TS setup:

1. An alternative grant is not acceptable to the non-AP STA SME
2. The ADDTS MPDU transmission failed

3. No ADDTS MPDU response is received from the HC (for example because of delay due to congestion, or because the response frame cannot be transmitted)

The first case is indistinguishable from success within the MAC sublayer[27], and so is not considered further here.

Cases 2 and 3 are considered to be the same as the non-AP STA cannot be sure that its transmission failed. Figure 68.3 summarizes this case. The non-AP STA MAC shall send a DELTS QoS Action request 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.

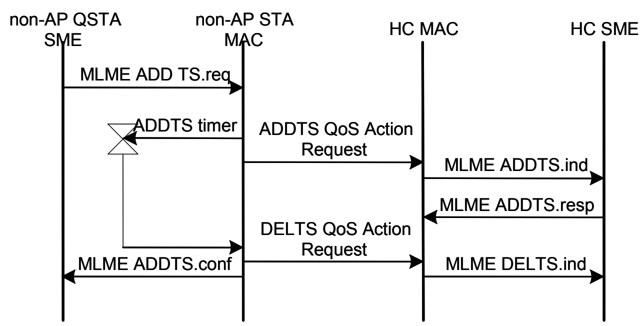


Figure 68.3 – Failed TS Setup detected within non-AP STA MAC

11.4.5 TS deletion

There are two types of TS deletion: non-AP STA-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 MPDU 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 STA. The case of HC-initiated TS deletion is the same with the non-AP STA and HC labels swapped over.

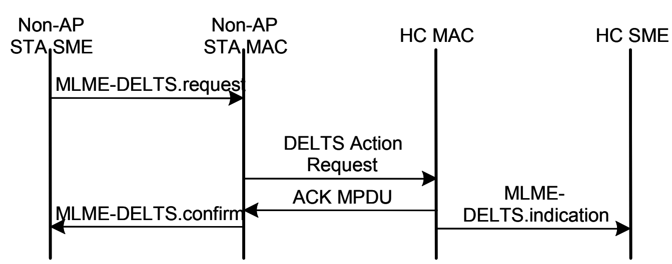


Figure 68.4 TS Deletion (non-AP STA-initiated)

11.4.6 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.

[27] This is only considered a failure in the SME of the non-AP STA.

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 MAC-UNITDATA.request primitives using this TS at the MAC service interface.

For a direct link TS, 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 or missing QoS NULL termination of a TXOP are both considered to be inactivity. 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 STA. Detection of inactivity of this type is optional.

In response to an inactivity timeout, the HC shall send a DELTS QoS Action request to the non-AP STA, and inform its SME using the MLME-DELTS.indication.

The case of uplink TS timeout is shown in figure 68.5.

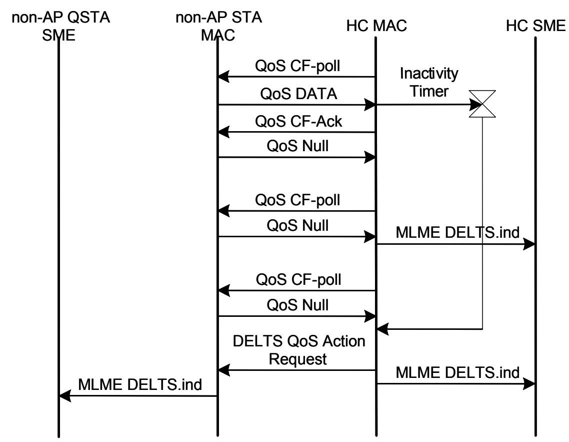


Figure 68.5 TS Timeout

11.5 Group Ack operation

Group Ack may be set up at the MAC or by the initiation of SME. The set up and deletion of Group Ack at the initiation of the SME is described in this subclause.

11.5.1 Set up and modification of the Group Ack parameters

The procedures for setting up and modifying the Group Ack parameters for originator and the recipient are described in 11.6.1.1 and 11.6.1.2 respectively and illustrated in Figure 68.6.

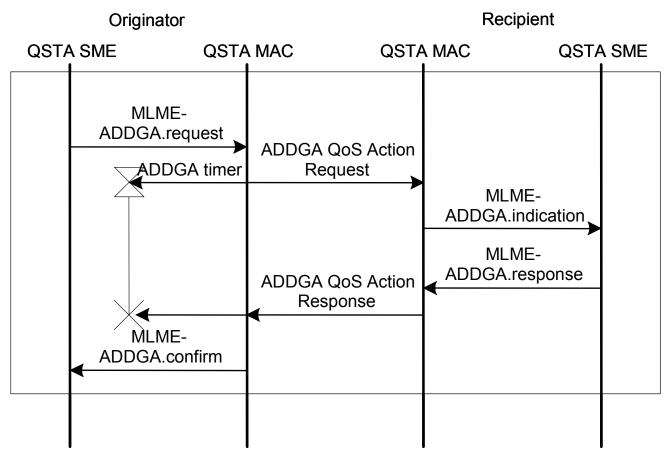


Figure 68.6 – Group Ack set up

11.5.1.1 Procedure at the originator

Upon receipt of MLME-ADDGA.request, an originating QSTA shall set up the Group Ack via the following procedure that has data traffic to send and intends to use Group Ack facility mechanism.

a) Check if the intended recipient QSTA is capable of participating in Group Ack mechanism by discovering and examining its “Group Ack” capability bit. If the recipient is capable of participating, the originator sends an ADDGA request QoS Action frame indicating the TID.

b) If an ADDGA response QoS Action frame is received with the matching Dialog Token and the TID, and with a Result Code set to a value of “SUCCESS”, the QSTA has established Group Ack mechanism with the receiving QSTA and the MLME shall issue an MLME-ADDGA.confirm indicating the successful completion of the operation.

c) If a ADDGA response QoS Action frame is received with the matching Dialog Token and the TID, and with a Result Code set to a value other than “SUCCESS”, the QSTA has not established Group Ack mechanism with the receiving QSTA and the MLME shall issue an MLME-ADDGA.confirm indicating the failure of the operation.

11.5.1.2 Procedure at the recipient

A receiver shall operate as follows in order to support Group Ack initialization and modification.

a) Whenever an ADDGA request QoS Action frame is received from another QSTA, the MLME shall issue an MLME-ADDGA.indication.

b) Upon receipt of the MLME-ADDGA.response, the QSTA shall respond by an ADDGA response QoS Action response frame with a Result code as defined in 7.4.3.2.

    1) If the Result code is “SUCCESS” the Group Ack is considered to be established with the originator. Contained in the frame are the type of Group Ack and the number of buffers that have been allocated for the support of this Group.

    2) If the Result code is “REFUSED” the Group Ack is not considered to have been established.

11.5.2 Tear down of the Group Ack mechanism

The procedure at the two QSTAs is described in 11.6.2.1 and 11.6.2.2 and illustrated in Figure 68.7.

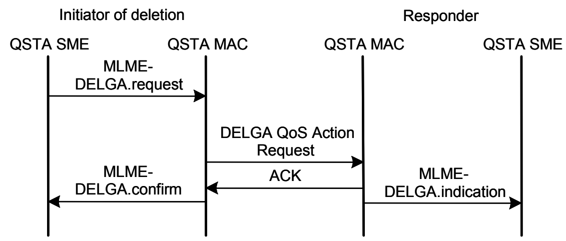


Figure 68.7 – Group Ack Deletion

11.5.2.1 Procedure at the initiator of the Group Ack teardown

Upon receipt of MLME-DELGA.request, the QSTA shall tear down the Group Ack vial the following procedure:

a) The QSTA shall transmit a DELGA request QoS Action frame.

b) Upon the receipt of an acknowledgement to the Delete Group Ack request QoS Action frame, the MLME issues a MLME-DELGA.confirm.

11.5.2.2 Procedure at the recipient of the DELGA request QoS Action frame

A QSTA shall issue a MLME-DELGA.indication when a DELGA request QoS Action frame is received:

11.5.3 Error recovery upon a peer failure

When a timeout of dot11PeerLivenessTimeout is detected, the QSTA shall act as though a DELGA request had been received and shall issue a MLME-DELGA.indication. The procedure is illustrated in Figure 68.8.

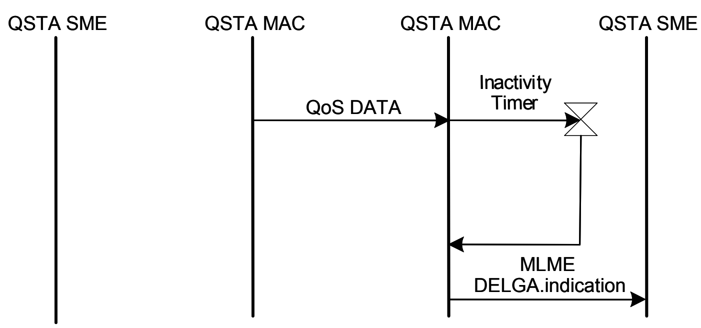


Figure 68.8 – Error Recovery by the Receiver upon a peer failure

11.6 Higher-layer timer synchronization

A MAC that supports the MLME-HL-SYNC primitives shall respond to a MLME-HL-SYNC.request with a MLME-HL-SYNC.confirm response of SUCCESS. This confirms to the requesting application that the specified group address has been entered into a MAC table of group addresses for which MLME-HL-SYNC.indication shall be provided.

In order to determine whether or not to provide an MLME-HL-SYNC.indication for a particular data frame, a MAC that supports MLME-HL-SYNC primitives compares the Address 1 field in a data frame's MAC header against a list of group addresses previously registered by an MLME-HL-SYNC.request. If the MAC and the AP are co-located within the same station, the MLME-HL-SYNC.indication shall occur when the last symbol of the data frame is transmitted. Otherwise, the indication shall occur when the last symbol of the data frame is received. In both cases, the MLME-HL-SYNC.indication provided is simultaneous (within implementation-dependent delay bounds) with the indication provided to other stations within the BSS for the same data frame.

Annex A

Add the following entries to the table in A.4.3:

A.4.3 IUT configuration

Item	IUT configuration	References	Status	Support
CF12	QoS supported	9.10, 9.11	O	Yes, No
CF13	Enhanced distributed channel access	9.10.1	CF12:M	Yes, No
CF14	QoS Parameterized - channel access	9.10.2	CF12:M	Yes, No

Insert the following at the end of A.4:

A.4.x QoS Base Functionality

Item	Protocol Capability	References	Status	Support
QB1	QoS frame format	7.1.3..1, 7.1.3.2, 7.1.3.4 - 7.1.3.6, 7.2.2, 7.2.3.1, 7.2.3.4 - 7.2.3.9, 7.2.3.12, 7.3.1.4, 7.3.1.7, 7.3.1.9, 7.3.1.4, 7.3.2.13 -7.3.2.20, 7.5	CF12:M	Yes, No
QB2	Per TID duplicate detection	7.1.3.4, 7.1.3.5, 9.10.1.6	CF12:M	Yes, No
QB3	Decode of no-ack policy in QoS data frames	7.1.3.5.2, 9.10.1.4, 9.10.1.5, 9.10.1.7.1, 9.10.1.7.2	CF12:M	Yes, No
QB4	Group acknowledgements	7.2.1.7, 7.2.1.8, 7.4.1.5, 7.4.1.6, 7.4.1.7, 9.11, 11.5	CF12:O	Yes, No
QB5	Automatic Power Save Delivery	7.3.2.18, 7.4.1.6, 11.2.3	CF12:O	Yes, No
QB6	Direct Link Protocol	7.3.2.20, 7.4.2, 10.3.13	(CF1 AND CF12):M (CF2 AND CF12):O	Yes, No

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	9.10.1.1	CF13:M	Yes, No
QD2	Per-channel access function differentiated channel access	9.10.1.2, 9.10.1.3, 9.10.1.5	CF13:M	Yes, No
QD3	Continuation TXOP support	9.10.1.4	CF13:M	Yes, No
QD4	Maintenance of within-queue ordering, exhaustive retransmission when sending non-QoS data frames	9.10.1.6	CF13:M	Yes, No
QD5	Distributed admission control	9.10.1.7	CF13:M	Yes, No

A.4.z QoS Parameterized Channel Access

Item	Protocol Capability	References	Status	Support
QP1	TSPEC and associated frame formats	7.4.1.1 – 7.4.1.4	CF14:M	Yes, No
QP2	HCF controlled channel access rules	9.1.3.2, 9.10.2.1 - 9.10.2.3	CF14:M	Yes, No
QP3	HCF controlled channel access schedule generation and management	9.10.2.4	(CF1 &CF14):M	Yes, No
QP4	HCF frame exchange sequences	9.10.3	CF14:M	Yes, No
QP5	TS Management	11.4	CF14:M	Yes, No
QP6	Minimum TSPEC parameter set	9.10.2.4	CF14:M	Yes, No
QP7	Power management	11.2.1.10	CF14:M	Yes, No

Annex C

Delete the text and SDL in the Annex.

Annex D

In "Major Sections" of Annex D, insert the following text to the end of MAC attributes:

```
        -- dot11EDCFTable                       ::= {dot11mac 4}
        -- dot11QosCountersTable                ::= {dot11mac 5}
```

In "SMT Station Config Table" of Annex D, insert the following text to the end of dot11StationConfigEntry sequence list:

```
    dot11QosOptionImplemented        TruthValue,
    dot11AssociateAsNonAPQSTA        TruthValue,
    dot11GroupAckOptionImplemented   TruthValue,
    dot11DirectOptionImplemented     TruthValue
```

Change the MIB attribute dot11DesiredBSSType as shown below:

```
dot11DesiredBSSType OBJECT-TYPE
      SYNTAX INTEGER { infrastructure(1), independent(2), any(3), qbss(4) }
      MAX-ACCESS read-write
      STATUS current
      DESCRIPTION
            "This attribute shall specify the type of BSS the station shall use
            when scanning for a BSS with which to synchronize. This value is
            used to filter Probe Response frames and Beacons. When set to
            infrastructure, the station shall only synchronize with a BSS whose
            Capability Information field has the ESS subfield set to 1. When set
            to independent, the station shall only synchronize with a BSS whose
            Capability Information field has the IBSS subfield set to 1. When
            set to any, the station may synchronize to either type of BSS. When
            set to QBSS, the station shall only synchronize with a BSS whose
            Capability Information field has the QoS subfield set to 1."
      ::= { dot11StationConfigEntry 10 }
```

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 }

dot11AssociateAsNonAPQSTA OBJECT-TYPE
      SYNTAX TruthValue
      MAX-ACCESS read-write
      STATUS current
      DESCRIPTION
            "This attribute, when TRUE, indicates that the station can be
            associated as a non-AP QSTA in a QBSS. This attribute, when FALSE,
            indicates that the station can not be associated as a non-AP QSTA in
            a QBSS. The default value of this attribute is TRUE."
      ::= { dot11StationConfigEntry 25 }

dot11GroupAckOptionImplemented OBJECT-TYPE
      SYNTAX TruthValue
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION
            "This attribute, when TRUE, indicates that the station
            implementation is capable of supporting Group Ack. The capability is
            disabled, otherwise. The default value of this attribute is FALSE."
      ::= { dot11StationConfigEntry 26 }

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 27 }
```

In "dot11OperationTable" of Annex D, insert the following text to the end of dot11OperationEntry sequence list:

```
dot11CAPLimit                           INTEGER,
dot11HCCWmin                            INTEGER,
dot11HCCWmax                            INTEGER,
dot11HCAIFS                             INTEGER,
dot11PeerLivenessTimeout                INTEGER,
dot11ADDTSResponseTimeout               INTEGER,
dot11ChannelUtilizationBeaconInterval   INTEGER,
dot11TXLimitDampingFactor               INTEGER,
dot11DLPIdleTimeOut                     INTEGER,
dot11ScheduleTimeout                    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 back-off. The default value of this attribute shall be 0."
      ::= { dot11OperationEntry 11 }

dot11HCCWax  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 back-off. The default value of this attribute shall be 0."
      ::= { dot11OperationEntry 12 }

dot11HCAIFS OBJECT-TYPE
      SYNTAX INTEGER (0..10)
      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 back-off. The default value of this
            attribute shall be 0."
      ::= { dot11OperationEntry 13 }

dot11PeerLivenessTimeout OBJECT-TYPE
      SYNTAX INTEGER (1..65535)
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION
            "This attribute shall specify the number of TUs that an originator
            or a destination shall wait  before  assuming that there has been a
            peer failure."
      ::= { dot11OperationEntry 14 }
```

```
dot11ADDTSResponseTimeout OBJECT-TYPE
      SYNTAX INTEGER (1..65535)
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION
            "This attribute shall specify the maximum number of seconds a TS is
            to be responded. The default value of this attribute shall be 1."
      ::= { dot11OperationEntry 15 }


dot11ChannelUtilizationBeaconInterval OBJECT-TYPE
      SYNTAX INTEGER (0..100)
      MAX-ACCESS read-write
      STATUS current
      DESCRIPTION
            "This attribute shall indicate the number of beacon intervals over
            which the averaging calculation should be calculated for purposed of
            calculating channel busy time.  The default value for this parameter
            shall be 50."
      ::= { dot11OperationEntry 16 }




dot11DLPIdleTimeOut OBJECT-TYPE
      SYNTAX INTEGER (0..65536)
      MAX-ACCESS read-write
      STATUS current
      DESCRIPTION
            "This attribute shall indicate the time out value in TUs at which if
            a direct link is found to be idle a transmitter will cease
            transmitting frames to the STA. The default value for this parameter
            shall be 500."
      ::= { dot11OperationEntry 17 }

dot11ScheduleTimeout OBJECT-TYPE
      SYNTAX INTEGER (0..100)
      MAX-ACCESS read-write
      STATUS current
      DESCRIPTION
            "This attribute shall indicate the duration in TUs after which a STA
            could go into power-save mode. The default value for this parameter
            shall be 10."
      ::= { dot11OperationEntry 18 }
```

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:

```
dot11QoSDiscardedFragmentCount OBJECT-TYPE
      SYNTAX Counter32
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION

            "This counter shall increment for each QoS Data MPDU that has been
            discarded."

      ::== { dot11CountersEntry 15 }
```

```
dot11AssociatedStationCount OBJECT-TYPE
      SYNTAX Counter32
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION

            “This counter, only available at QAP, shall increment when a station
            associates or reassociates. This counter shall decrement when a
            station disassociates.”

      ::== { dot11CountersEntry 16 }

dot11QoSCFPollsReceivedCount OBJECT-TYPE
      SYNTAX Counter32
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION

            “This counter shall increment for each QoS (+)CF-Poll that has been
            received.”

      ::== { dot11CountersEntry 17 }

dot11QoSCFPollsUnusedCount OBJECT-TYPE
      SYNTAX Counter32
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION

            “This counter shall increment for each QoS (+)CF-Poll that has been
            received but not used.”

      ::== { dot11CountersEntry 18 }

dot11QoSCFPollsUnusableCount OBJECT-TYPE
      SYNTAX Counter32
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION

            “This counter shall increment for each QoS (+)CF-Poll that has been
            received but could not be used due to the TXOP size being smaller
            than the time that is required for one frame exchange sequence.”

      ::== { dot11CountersEntry 19 }
```

Insert the following changes to the 802.11 MIB in Annex D, after the Group Addresses Table:

```
-- ********************************************************************
-- * SMT EDCF Config TABLE
-- ********************************************************************

dot11EDCFTable OBJECT-TYPE
      SYNTAX SEQUENCE OF Dot11EDCFTableEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
            “Conceptual value for EDCF default parameter values. This table
            shall contain the four entries of the EDCF parameters corresponding
            to four possible ACs.”
      REFERENCE “IEEE 802.11 Tge Draft Version, 9.10.1.X (to be filled later)”
   ::= { dot11mac 4}

dot11EDCFTableEntry OBJECT-TYPE
      SYNTAX Dot11EDCFTableEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
            “An Entry (conceptual row) in the EDCF Table.
```

```
            ifIndex – Each 802.11 interface is represented by an ifEntry.
            Interface tables in this MIB module are indexed by ifIndex."

        INDEX {ifIndex, dot11EDCFTableIndex}
    ::= { dot11EDCFTable 1 }

Dot11EDCFTableEntry ::=
        SEQUENCE {
            dot11EDCFTableIndex                     INTEGER,
            dot11EDCFTableCWmin                     INTEGER,
            dot11EDCFTableCWmax                     INTEGER,
            dot11EDCFTableAIFS                      INTEGER,
            dot11EDCFTableTXOPLimit                 INTEGER,
            dot11EDCFTableTXLimitDampingFactor      INTEGER
             }

dot11EDCFTableIndex OBJECT-TYPE
        SYNTAX INTEGER (1..4)
        MAX-ACCESS not-accessible
        STATUS current
        DESCRIPTION
            "The auxiliary variable used to identify instances of the columnar
            objects in the EDCF Table. The value of this variable is equal to AC
            + 1."
    ::= {dot11EDCFTableEntry 1 }

dot11EDCFTableCWmin OBJECT-TYPE
        SYNTAX INTEGER (0..255)
        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 a QSTA for a particular AC for
            generating a random number for the back-off. The default value for
            this attribute is
            1) aCWmin, if dot11EDCFTableIndex is 1 or 2.
            2) (aCWmin+1)/2 – 1, if dot11EDCFTableIndex is 3.
            3) (aCWmin+1)/4 – 1, if dot11EDCFTableIndex is 4."

    ::= {dot11EDCFTableEntry 2 }

dot11EDCFTableCWmax OBJECT-TYPE
        SYNTAX INTEGER (0..65535)
        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 a QSTA for a particular AC for
            generating a random number for the back-off. The default value for
            this attribute is
            1) aCWmax, if dot11EDCFTableIndex is 1 or 2.
            2) aCWmin, if dot11EDCFTableIndex is 3.
            3) (aCWmin+1)/2 – 1, if dot11EDCFTableIndex is 4."

    ::= {dot11EDCFTableEntry 3 }

dot11EDCFTableAIFS OBJECT-TYPE
        SYNTAX INTEGER (0..10)
        MAX-ACCESS read-write
        STATUS current
        DESCRIPTION
            "This attribute shall specify the number of slots, after a SIFS
            duration, that the QSTA, for a particular AC, shall sense the medium
            idle either before transmitting or executing a back-off. The default
            value for this attribute is
            1) 2, if dot11EDCFTableIndex is 1,
            2) 1, otherwise."

    ::= {dot11EDCFTableEntry 4 }
```

```
dot11EDCFTableTXOPLimit OBJECT-TYPE
      SYNTAX INTEGER (0..65535)
      MAX-ACCESS read-write
      STATUS current
      DESCRIPTION
            "This attribute shall specify the maximum number of microseconds a
            QSTA can burst in EDCF for a given UP. The default value for this
            attribute is
            1) 0 for all PHYs, if dot11EDCFTableIndex is 1; this implies that
               the sender can send one MSDU in an EDCF TXOP limit,
            2) 1500 microseconds for IEEE 802.11a/g PHY and 3000 microseconds
               for IEEE 802.11b PHY, if dot11EDCFTableIndex is 2,
            3) 3000 microseconds for IEEE 802.11a/g PHY and 6000 microseconds
               for IEEE 802.11b PHY, if dot11EDCFTableIndex is 3,
            4) 1500 microseconds for IEEE 802.11a/g PHY and 3000 microseconds
               for IEEE 802.11b PHY, if dot11EDCFTableIndex is 4."

   ::= {dot11EDCFTableEntry 5 }

dot11EDCFTableTXLimitDampingFactor OBJECT-TYPE
      SYNTAX INTEGER (0..100)
      MAX-ACCESS read-write
      STATUS current
      DESCRIPTION
            "This attribute shall indicate the highest value of the damper
            factor in the units of 0.01.  The default value for this parameter
            shall be 90."
      ::= { dot11EDCFTableEntry 6 }


-- *******************************************************************
-- * End of dot11EDCF TABLE
-- *******************************************************************

-- *******************************************************************
-- * dot11QosCounters TABLE
-- *******************************************************************

dot11QosCountersTable OBJECT-TYPE
      SYNTAX SEQUENCE OF Dot11QosCountersEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION

            "Group containing attributes that are MAC counters implemented as a
            table to allow for multiple instantiations on an agent."

      ::= { dot11mac 5 }

dot11QosCountersEntry OBJECT-TYPE
      SYNTAX Dot11QosCountersEntry
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION
            "An Entry (conceptual row) in the EDCF Table.

            ifIndex – Each 802.11 interface is represented by an ifEntry.
            Interface tables in this MIB module are indexed by ifIndex."

      INDEX {ifIndex, dot11QosCounterIndex}
   ::= { dot11QosCountersTable 1 }

Dot11QosCountersEntry ::=
      SEQUENCE {
            dot11QosCounterIndex                 INTEGER,
            dot11QosTransmittedFragmentCount     Counter32,
            dot11QosFailedCount                  Counter32,
            dot11QosRetryCount                   Counter32,
            dot11QosMultipleRetryCount           Counter32,
            dot11QosFrameDuplicateCount          Counter32,
            dot11QosRTSSuccessCount              Counter32,
```

```
            dot11QosRTSFailureCount                  Counter32,
            dot11QosACKFailureCount                  Counter32,
            dot11QosReceivedFragmentCount            Counter32,
            dot11QosTransmittedFrameCount            Counter32,
            dot11QosDiscardedFrameCount              Counter32,
            dot11QosMPDUsReceivedCount               Counter32,
            dot11QosRetriesReceivedCount             Counter32 }

dot11QosCounterIndex OBJECT-TYPE
      SYNTAX INTEGER (1..16)
      MAX-ACCESS not-accessible
      STATUS current
      DESCRIPTION

            "The auxiliary variable used to identify instances of the columnar
            objects in the QOSCounter Table. The value of this variable is equal
            to TID + 1."

   ::= {dot11QosCountersEntry 1 }


dot11QosTransmittedFragmentCount OBJECT-TYPE
      SYNTAX Counter32
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION

            "This counter shall be incremented for an acknowledged MPDU, for a
            particular priority, with an individual address in the address 1
            field or an MPDU with a multicast address in the address 1 field,
            either belonging to a particular TID. This counter has relevance
            only for TIDs between 0 and 7."

      ::= { dot11QosCountersEntry 2 }

dot11QosFailedCount OBJECT-TYPE
      SYNTAX Counter32
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION

            "This counter shall increment when an MSDU, for a particular
            priority, is not transmitted successfully due to the number of
            transmit attempts exceeding either the dot11ShortRetryLimit or
            dot11LongRetryLimit. This counter has relevance only for TIDs
            between 0 and 7."

      ::= { dot11QosCountersEntry 3 }

dot11QoSsRetryCountTC[0..7] OBJECT-TYPE
      SYNTAX Counter32
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION

            "This counter shall increment when an MSDU, of a particular
            priority, is successfully transmitted after one or more
            retransmissions. This counter has relevance only for TIDs between 0
            and 7."

      ::= { dot11QosCountersEntry 4 }

dot11QosMultipleRetryCount OBJECT-TYPE
      SYNTAX Counter32
      MAX-ACCESS read-only
      STATUS current
      DESCRIPTION

            "This counter shall increment when an MSDU, of a particular
            priority, is successfully transmitted after more than one
```

```
            retransmission. This counter has relevance only for TIDs between 0
            and 7."

        ::= { dot11QosCountersEntry 5}

dot11QosFrameDuplicateCount OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION

            "This counter shall increment when a frame, of a particular
            priority, is received that the Sequence Control field indicates is a
            duplicate. This counter has relevance only for TIDs between 0 and
            7."

        ::= { dot11QosCountersEntry 6 }

dot11QosRTSSuccessCount OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION

            "This counter shall increment when a CTS is received in response to
            an RTS that has been sent for the transmission of MPDU of a
            particular priority. This counter has relevance only for TIDs
            between 0 and 7."

        ::= { dot11QosCountersEntry 7 }

dot11QosRTSFailureCount OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION

            "This counter shall increment when a CTS is not received in response
            to an RTS that has been sent for the transmission of an MPDU of a
            particular priority. This counter has relevance only for TIDs
            between 0 and 7."

        ::= { dot11QosCountersEntry 8 }

dot11QosACKFailureCount OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION

            "This counter shall increment when an ACK is not received in
            response to an MPDU of a particular priority. This counter has
            relevance only for TIDs between 0 and 7."

        ::= { dot11QosCountersEntry 9 }

dot11QosReceivedFragmentCount OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION

            "This counter shall be incremented for each successfully received
            MPDU of type Data of a particular priority. This counter has
            relevance only for TIDs between 0 and 7."

        ::= { dot11QosCountersEntry 10 }

dot11QosTransmittedFrameCount OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
```

```
        STATUS current
        DESCRIPTION

                "This counter shall increment for each successfully transmitted MSDU
                of a particular priority. This counter has relevance only for TIDs
                between 0 and 7."

        ::= { dot11QosCountersEntry 11 }

dot11QosDiscardedFrameCount OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION

                "This counter shall increment for each Discarded MSDU of a
                particular priority. This counter has relevance only for TIDs
                between 0 and 7."

        ::== { dot11QosCountersEntry 12 }

dot11QosMPDUsReceivedCount OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION

                "This counter shall increment for each received MSDU of a particular
                TID."

        ::== { dot11QosCountersEntry 13 }

dot11QosRetriesReceivedCount OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION

                "This counter shall increment for each received MPDU of a particular
                TID with the retry bit set to 1."

        ::== { dot11QosCountersEntry 14 }

-- **********************************************************************
-- * 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.

Annex H

(Informative)

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	CBR Traffic	Bursty Traffic	Unspecified non-QoS Traffic
Nominal MSDU Size	S	S	X	DC
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
Maximum Service Interval	S	Delay Bound/Number of Retries (AV typically uses this).	Delay Bound/Number of Retries, if Delay Bound present	DC
Inactivity Interval	Always specified.			
Minimum Data Rate	Must be specified if Peak Data Rate is specified.	Equal to Mean Data Rate	X	DC
Mean Data Rate	S	S	DC	DC
Maximum Burst Size	X	X	S	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

Delay Bound	S	S	DC	X
Surplus Bandwidth Allowance	Must be specified if delay and jitter bounds present.			DC

Annex I

(Informative)

Traffic Classification (TCLAS) Element

Traffic Classification (TCLAS) element contains the set of parameters necessary to identify incoming MSDUs with a particular traffic stream (TS) to which they belong. MSDUs that are identified using this TCLAS belong to the TS of the affiliated TSPEC, i.e., the TSPEC contained in the same ADDTS Action frame or MLME-ADDTS.indication primitive as this TCLAS. The structure of this element is defined in Figure I.1.

Octets: 1	1	L
Element ID (14)	**Length** (L)	**Frame Classifier**

Figure I.1 –TCLAS element format

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 I.1. Three classifier types are defined below.

Table I.1 – Frame classifier type

Classifier Type	Classifier Parameters
0	Ethernet LLC 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 classifier 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 TSs 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 LLC packet header: Source Address, Destination Address, and Type. The Frame Classifier field for Classifier Type 0 is defined in Figure I.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 I.2 – Frame Classifier of Classifier Type 0 Field

For Classifier Type 1, 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: Type of Service (TOS) and Protocol ,or the following parameter in an IPv6 header: Flow Label. The Frame Classifier fields for Classifier Type 1 for traffic over IPv4 and IPv6 are defined in figures I.3 and I.4, respectively.

Octets: 1	1	4	4	2	2	1	1	1	1
Classifier Type (1)	**Classifier Mask**	**Source IP Address**	**Destination IP Address**	**Source Port**	**Destination Port**	**Version**	**TOS**	**Protocol**	**Reserved**

Figure I.3 – Frame Classifier of Classifier Type 1 Field for traffic over IPv4

Octets: 1	1	16	16	2	2	1	3
Classifier Type (1)	**Classifier Mask**	**Source IP Address**	**Destination IP Address**	**Source Port**	**Destination Port**	**Version**	**Flow Label**

Figure I.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 I.5.

Octets: 1	1	**2**		
		B0 B2	B3 B14	**B15**
Classifier Type(2)	**Classifier Mask**	**User Priority**	**VLAN ID**	**Reserved**

Figure I.5 – Frame Classifier of Classifier Type 2 Field