IEEE Standard for Wireless Access in Vehicular Environments (WAVE)— Multi-channel Operation

IEEE Vehicular Technology Society

Sponsored by the
Intelligent Transportation Systems Committee

IEEE
3 Park Avenue
New York, NY 10016-5997
USA

7 February 2011

IEEE Std 1609.4™-2010
(Revision of
IEEE Std 1609.4-2006)

IEEE Std 1609.4™-2010
(Revision of
IEEE Std 1609.4-2006)

IEEE Standard for Wireless Access in Vehicular Environments (WAVE)— Multi-channel Operation

Sponsor

Intelligent Transportation Systems Committee

of the

IEEE Vehicular Technology Society

Approved 30 September 2010

IEEE-SA Standards Board

Approved 26 August 2011

American National Standards Institute

Abstract: Multi-channel wireless radio operations, Wireless Access in Vehicular Environments (WAVE) mode, medium access control (MAC), and physical layers (PHYs), including the operation of control channel (CCH) and service channel (SCH) interval timers, parameters for priority access, channel switching and routing, management services, and primitives designed for multi-channel operations are described in this standard.

Keywords: 1609.4, channel coordination, multi-channel operation, user priority, Wireless Access in Vehicular Environments (WAVE)

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

PDF: ISBN 978-0-7381-6489-2 STD97033
Print: ISBN 978-0-7381-6490-8 STDPD97033

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied "**AS IS**."

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation, or every ten years for stabilization. When a document is more than five years old and has not been reaffirmed, or more than ten years old and has not been stabilized, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon his or her independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration. A statement, written or oral, that is not processed in accordance with the IEEE-SA Standards Board Operations Manual shall not be considered the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal interpretation of the IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position, explanation, or interpretation of the IEEE.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Recommendations to change the status of a stabilized standard should include a rationale as to why a revision or withdrawal is required. Comments and recommendations on standards, and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854
USA

Introduction

This introduction is not part of IEEE Std 1609.4-2010, IEEE Standard for Wireless Access in Vehicular Environments (WAVE)—Multi-channel Operation.

A WAVE system is a radio communications system intended to provide seamless, interoperable services to transportation. These services include those recognized by the U.S. National Intelligent Transportation Systems (ITS) Architecture [B3][a] and many others contemplated by the automotive and transportation infrastructure industries. These services include vehicle-to-roadside communication, vehicle-to-vehicle communications, and potentially communication among other devices. Multi-channel operation provides medium-access channel (MAC)-layer enhancements to the capabilities standardized in IEEE Std 802.11p.[b] This is but one component in the overall WAVE architecture, which is described in IEEE P1609.0.

Notice to users

Laws and regulations

Users of these documents should consult all applicable laws and regulations. Compliance with the provisions of this standard does not imply compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Copyrights

This document is copyrighted by the IEEE. It is made available for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making this document available for use and adoption by public authorities and private users, the IEEE does not waive any rights in copyright to this document.

Updating of IEEE documents

Users of IEEE standards should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect. In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit the IEEE Standards Association web site at http://ieeexplore.ieee.org/xpl/standards.jsp, or contact the IEEE at the address listed previously.

For more information about the IEEE Standards Association or the IEEE standards development process, visit the IEEE-SA web site at http://standards.ieee.org.

[a] The numbers in brackets correspond to those of the bibliography in Annex A.
[b] Information on references can be found in Clause 2.

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: http://standards.ieee.org/reading/ieee/updates/errata/index.html. Users are encouraged to check this URL for errata periodically.

Interpretations

Current interpretations can be accessed at the following URL: http://standards.ieee.org/reading/ieee/interp/index.html.

Patents

Attention is called to the possibility that implementation of standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

Participants

At the time this standard was completed, the Dedicated Short Range Communication Working Group had the following membership:

Thomas M. Kurihara, *Chair*
John Moring, *Vice Chair*

Scott Andrews
Roger Berg
Anthony Chou
Susan Dickey
Wayne Fisher
Ramez Gerges
Paul Gray
Refi-Tugrul Güner
Somak Datta Gupta
Gloria Gwynne
Ronald Hochnadel
Gavin Holland
Wendong Hu
Guo-Huang Hsu
Daniel Jiang
Carl Kain
Douglas Kavner
David Kelley
John Kenney
Minjung Kim
Jeremy Landt
Michael Li
Jason Liu
Jules Madey
Alastair Malarky
Justin McNew
Vinuth Rai
Randal Roebuck
Richard Roy
Kuo Sheng-Po
François Simon
George Vlantis
Timothy Weil
Aaron Weinfield
William Whyte

The following members of the balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Iwan Adhicandra
Lee Armstrong
H. Stephen Berger
Harry Bims
William Byrd
Kai T. Chen
Keith Chow
Michael Coop
Susan Dickey
Sourav Dutta
Wayne Fisher
Andre Fournier
Avraham Freedman
Ramez Gerges
Paul Gray
Randall Groves
Refi-Tugrul Güner
Gloria Gwynne
Ronald Hochnadel
Werner Hoelzl
Chung-Hsien Hsu
Wendong Hu
Daniel Jiang
Carl Kain
Piotr Karocki
Douglas Kavner
John Kenney
Stuart J. Kerry
Dmitri Khijniak
Max Kicherer
Gerald L. Kolbe
Hsu Kuo-Huang
Thomas M. Kurihara
Richard Lancaster
Jeremy Landt
Hsia-Hsin Li
Jun Liu
William Lumpkins
G. Luri
Dmitriy Lyubskiy
Alastair Malarky
Sean Maschue
Justin McNew
John Moring
Michael S. Newman
Satoshi Oyama
Venkatesha Prasad
Robert Robinson
Randal Roebuck
Richard Roy
Randall Safier
Reza Salehi
Bartien Sayogo
Gil Shultz
Walter Struppler
Mark Sturza
Dale Sumida
Jasja Tjink
George Vlantis
William Whyte

When the IEEE-SA Standards Board approved this standard on 30 September 2010, it had the following membership:

Robert M. Grow, *Chair*
Richard H. Hulett, *Vice Chair*
Steve M. Mills, *Past President*
Judith Gorman, *Secretary*

Karen Bartleson
Victor Berman
Ted Burse
Clint Chaplin
Andy Drozd
Alexander Gelman
Jim Hughes
Young Kyun Kim
Joseph L. Koepfinger*
John Kulick
David J. Law
Hung Ling
Oleg Logvinov
Ted Olsen
Ronald C. Petersen
Thomas Prevost
Jon Walter Rosdahl
Sam Sciacca
Mike Seavey
Curtis Siller
Don Wright

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Satish Aggarwal, *NRC Representative*
Richard DeBlasio, *DOE Representative*
Michael Janezic, *NIST Representative*

Michelle D. Turner
IEEE Standards Program Manager, Document Development

Matthew J. Ceglia
IEEE Standards Program Manager, Technical Program Development

Contents

IEEE Standard for Wireless Access in Vehicular Environments (WAVE)—Multi-channel Operation

IMPORTANT NOTICE: This standard is not intended to ensure safety, security, health, or environmental protection. Implementers of the standard are responsible for determining appropriate safety, security, environmental, and health practices or regulatory requirements.

This IEEE document is made available for use subject to important notices and legal disclaimers. These notices and disclaimers appear in all publications containing this document and may be found under the heading "Important Notice" or "Important Notices and Disclaimers Concerning IEEE Documents." They can also be obtained on request from IEEE or viewed at http://standards.ieee.org/IPR/disclaimers.html.

1. Overview

1.1 Scope

The scope of this standard is the specification of medium access control (MAC) sublayer functions and services that support multi-channel wireless connectivity between IEEE 802.11 Wireless Access in Vehicular Environments (WAVE) devices.

1.2 Purpose

To enable effective mechanisms that control the operation of upper layer data transfers across multiple channels, without requiring knowledge of physical layer (PHY) parameters, and describe the multi-channel operation channel routing and switching for different scenarios.

1.3 Conformance

Per the *IEEE Style Manual* [B1],[1] this standard includes normative and informative information. Normative text may describe mandatory or optional features. A mandatory feature may have optional as well as

[1] The numbers in brackets correspond to those of the bibliography in Annex A.

mandatory components. For example, a mandatory message may have optional fields. An optional feature may have components that are mandatory if the feature is supported. For example, an optional message might require a certain field. Additionally, a feature may be conditional on support of another feature. For example, if A is supported, at least one of B or C must be supported.

In this standard, features are designated as mandatory, optional, or conditional in the introduction to the subclause specifying the feature. The designation applies to the requirements found in the subclause itself as well as any subordinate subclauses, unless otherwise indicated. The word *shall*, when applied to a component of an optional feature, indicates that the component is mandatory if the feature is supported, i.e., conditional on support for the feature. The protocol implementation conformance statement (PICS) in Annex G summarizes the features and their components.

1.4 Document conventions

Unless otherwise noted, conventions follow those in IEEE Std 802.11, including conventions for the ordering of information within data items.

Numbers are decimal unless otherwise noted. Numbers preceded by 0x indicate hexadecimal numbers, so that 0xFF is equivalent to “FF hexadecimal.” Field lengths are in octets unless otherwise noted.

Words in italics refer to data items that appear in the management information base (MIB), primitives, or over the air frames.

Descriptive, informative information is generally found at the beginning of a clause or subclause, with normative text following. Figures are used for illustration and are informative, unless otherwise noted.

Unless otherwise noted, references to a MIB refer to the MLME MIB specified in Annex E.

2. Normative references

The following referenced documents and URLs are indispensable for the application of this document (i.e., they must be understood and used, so each referenced document is cited in text and its relationship to this document is explained). For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

IEEE P802.11REVmb™/D3.0, March 2010, IEEE Draft Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.[2]

IEEE P1609.0™/D0.9, April 2010, Draft Standard for Wireless Access in Vehicular Environments (WAVE)—Architecture.

IEEE P1609.2™, D5, June 2010, Draft IEEE Standard for Wireless Access in Vehicular Environments—Security Services for Applications and Management Messages.

[2] Numbers preceded by P are IEEE authorized standards projects that were not approved by the IEEE-SA Standards Board at the time this publication went to press. For information about obtaining drafts, contact the IEEE.

IEEE Std 802®, IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture.[3, 4, 5]

IEEE Std 802.11™, IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

IEEE Std 802.11k™-2008, IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 1: Radio Resource Measurement of Wireless LANs.

IEEE Std 802.11p™, IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments.

IEEE Std 1609.3™, IEEE Standard for Wireless Access in Vehicular Environments (WAVE)—Networking *Services*.

IETF RFC 1042, Standard for the Transmission of IP Datagrams over IEEE 802 Networks.[6]

ISO/IEC 8802-2 [IEEE Std 802.2™], IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 2: Logical Link Control.

3. Definitions, acronyms, and abbreviations

Definitions

For the purposes of this document, the following terms and definitions apply. IEEE P1609.0 and *The IEEE Standards Dictionary: Glossary of Terms & Definitions* should be consulted for terms not defined in this clause.[7]

channel coordination: The mechanisms specified in this standard that facilitate Wireless Access in Vehicular Environments (WAVE) operation of at least one physical layer (PHY) over multiple wireless channels, including synchronization.

control channel (CCH): A single radio channel, not a service channel, intended for the exchange of management information, including Wireless Access in Vehicular Environments (WAVE), Service Advertisements, and WAVE Short Messages.

[3] IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, Piscataway, NJ 08854, USA (http://standards.ieee.org/).
[4] The IEEE standards or products referred to in this clause are trademarks owned by the Institute of Electrical and Electronics Engineers, Incorporated.
[5] IEEE and 802 are registered trademarks in the U.S. Patent & Trademark Office, owned by The Institute of Electrical and Electronics Engineers, Incorporated.
[6] IETF publications are available from the Internet Engineering Task Force, IETF Secretariat, c/o Association Management Solutions, LLC (AMS), 48377 Fremont Boulevard, Suite 117, Fremont, CA 94538 (http://www.ietf.org).
[7] *The IEEE Standards Dictionary: Glossary of Terms & Definitions* is available at http://shop.ieee.org/.

control channel (CCH) interval: A specified periodic interval of time with respect to Coordinated Universal Time (UTC). During the CCH interval a device using alternating channel access will tune to the CCH, and outside the CCH interval the device will tune to a channel other than the CCH. A CCH interval and the subsequent service channel (SCH) interval comprise a sync interval.

Ethertype: The Ethernet Type field specified in IETF RFC 1042, used to identify the higher layer protocol above the Logical Link Control (LLC).

guard interval: A time interval at the start of each control channel (CCH) interval and service channel (SCH) interval during which devices that are switching channels do not transmit.

higher layer: A functionality (e.g., an application), residing, in the context of the Open System Interconnection (OSI) protocol suite reference model, in the management or data plane above the services defined in this standard.

managed Wireless Access in Vehicular Environments (WAVE) device: A WAVE device that supports management through the use of an IEEE 1609 standards-defined management information base.

management ID: Identifier of an IEEE 1609 management entity used to indicate the source and destination of management information. The values are specified in IEEE P1609.0.

multi-channel operation: The operation of a Wireless Access in Vehicular Environments (WAVE) device to utilize more than one wireless channel. May involve channel coordination.

multi-physical layer (PHY) device: A Wireless Access in Vehicular Environments (WAVE) device capable of simultaneous operation on multiple radio channels.

OCBEnabled: Indicates operation outside the context of a basic service set as specified in IEEE Std 802.11p (by setting *dot11OCBEnabled* to TRUE) (i.e., used for WAVE operation).

pseudonymity: A property of a pseudonymous entity. A pseudonymous entity has the property that its permanent or long-lived identities, and its long-term patterns of behavior, cannot be deduced from its network traffic and are only available to appropriately authorized parties.

service channel (SCH): Any channel that is not the control channel, intended for management frames and higher layer information exchanges (Wireless Access in Vehicular Environments [WAVE] Short Message [WSMs] and Internet Protocol version 6 [IPv6] packets). There may be more than one service channel defined in a given spectrum.

service channel (SCH) interval: A recurring time interval during which a device using alternating channel access will tune to an SCH and outside of which the device will tune to the CCH. An SCH interval and the preceding CCH interval comprise a sync interval.

single-physical layer (PHY) device: A Wireless Access in Vehicular Environments (WAVE) device not capable of simultaneous operation on multiple radio channels.

switching device: A WAVE device with at least one physical layer (PHY) that is switching between channels (e.g., service channel [SCH] and control channel [CCH]).

sync interval: A repeating time interval comprised of one control channel (CCH) interval followed by one service channel (SCH) interval.

Timing Advertisement frame: A management frame specified in IEEE Std 802.11p used to carry timing information.

transmitter profile: A data object containing communication parameters to be used by the transmitter, including SCH number, power level, data rate and the adaptable status of power level, for Internet Protocol (IP)-based data transfers.

Vendor Specific Action frame: A management frame specified in IEEE Std 802.11. Despite the name, when the first 36 bits of the *Organization Identifier* are equal to 0x0050C24A4, the contents are specified within the 1609 standards for use in carrying IEEE 1609 management information including Wireless Access in Vehicular Environments (WAVE) Service Advertisements.

Wireless Access in Vehicular Environments (WAVE) device: A device that is compliant to IEEE Std 1609.3, IEEE Std 1609.4, and IEEE Std 802.11, operating outside the context of a basic service set. (See IEEE Std 802.11p specification of station [STA] transmission of data frames outside the context of a basic service set.)

Wireless Access in Vehicular Environments (WAVE) Management Entity (WME): A set of management functions required to provide WAVE networking services. It is specified in IEEE Std 1609.3.

Wireless Access in Vehicular Environments (WAVE) Service Advertisement (WSA): A data structure specified in IEEE Std 1609.3 containing information including the announcement of the availability of services.

Wireless Access in Vehicular Environments (WAVE) Short Message Protocol (WSMP): A protocol for rapid exchange of messages in a rapidly varying radio frequency (RF) environment where low latency is also an important objective. It is specified in IEEE Std 1609.3.

wireless channel: An instance of one radio frequency (RF) channel within an operating class as defined in IEEE Std 802.11/IEEE Std 802.11p/Annex J of IEEE P802.11REVmb.

3.1 Acronyms and abbreviations

AC	access category
ACI	access category index
CCH	control channel
CW	contention window
EDCA	enhanced distributed channel access
GPS	Global Positioning System
IP	Internet Protocol
LLC	Logical Link Control
MAC	medium access control
MIB	management information base
MLME	MAC sublayer management entity
MPDU	MAC protocol data unit
MSDU	MAC service data unit
OCXO	oven stabilized crystal oscillator
PHY	physical layer
PICS	protocol implementation conformance statement
PLME	physical layer management entity
PPS	pulse per second
RCPI	Received Channel Power Indicator
RF	radio frequency
RFC	Request for Comments
SAP	service access point
SCH	service channel
SNAP	Subnetwork Access Protocol

STA	station
TA	Timing Advertisement (frame)
TSF	timing synchronization function
TXOP	transmit opportunity
UP	user priority
UTC	Coordinated Universal Time
VSA	Vendor Specific Action (frame)
WAVE	Wireless Access in Vehicular Environments
WME	WAVE Management Entity
WSA	WAVE Service Advertisement
WSM	WAVE Short Message
WSMP	WAVE Short Message Protocol

4. General description

4.1 Overview

WAVE provides a communication protocol stack optimized for the vehicular environment, employing both customized and general-purpose elements. IEEE P1609.0 provides a description of the WAVE system architecture and operations. To operate over multiple wireless channels while in operation with OCBEnabled, there is a need to perform channel coordination. OCBEnabled indicates operation outside the context of a basic service set as specified in IEEE Std 802.11p (by setting *dot11OCBEnabled* to TRUE) (i.e., WAVE operation).

Channel coordination is specified in this standard and consists of additional features for OCBEnabled operations in the MAC sublayer specified in IEEE Std 802.11p. Both data plane and management plane features are included as shown in Figure 1. These features are introduced in 4.2 and 4.3 and specified in Clause 5 and Clause 6.

4.2 Reference model

The basic reference model relevant to this standard is shown in Figure 1. WAVE supports both IP- and non–IP-based data transfers, although individual devices might support only one networking protocol. Non–IP-based data transfers are supported through the WAVE Short Message Protocol (WSMP) specified in IEEE Std 1609.3. Channel coordination is a collection of enhancements to the IEEE 802.11 MAC, and interacts with the IEEE 802.2 LLC and IEEE 802.11 PHY. The WAVE Management Entity (WME) and corresponding network services are specified in IEEE Std 1609.3. WAVE security services are specified in IEEE P1609.2.

The MAC and PHY layers conceptually include management entities, called MAC sublayer management and PHY layer management entities (MLME and physical layer management entity [PLME], respectively). These management entities provide the layer management service interfaces through which layer management functions may be invoked. In Clause 6, this document specifies extensions to the IEEE 802.11 MAC sublayer management entity (MLME; as amended by IEEE Std 802.11p) to provide channel coordination. The interactions between the various entities are through service access point (SAPs) across which defined primitives are exchanged; see Clause 7.

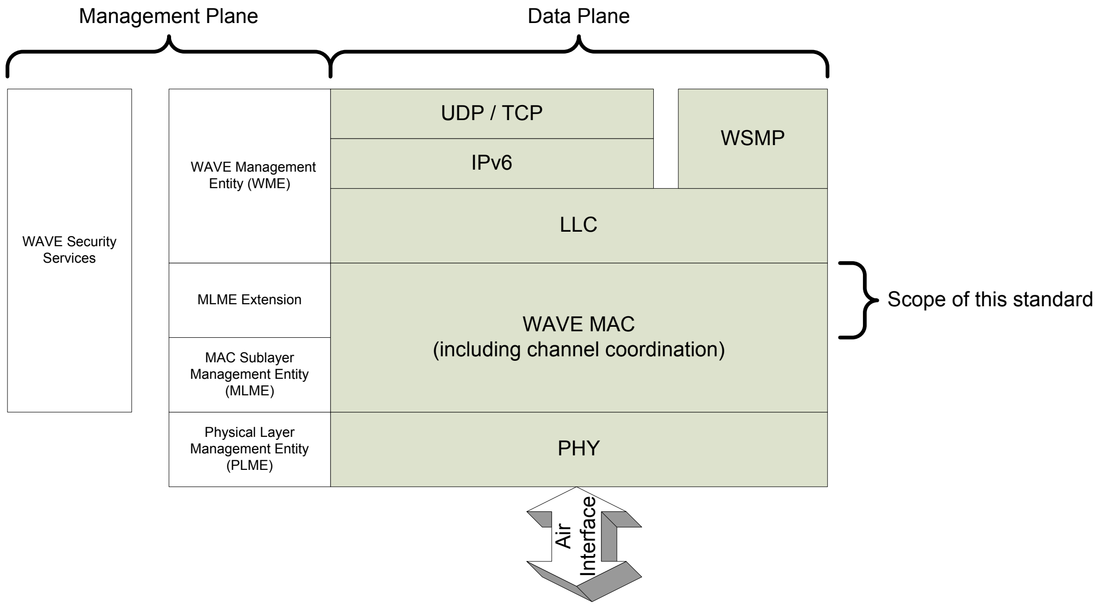


Figure 1 —Reference model

4.3 Overview of the services

4.3.1 General

The services provided by this standard are used to manage channel coordination and to support MAC service data unit (MSDU) delivery. Both data plane and management plane features are specified.

4.3.2 Data plane services

The MLME data services specified in Clause 5 comprise the following:

— Channel coordination

  The MAC sublayer coordinates channel intervals so that data packets are transmitted on the proper RF channel at the right time.

— Channel routing

  The MAC sublayer handles inbound and outbound higher layer data. This specification includes routing of data packets from the LLC to the designated channel, and setting parameters (e.g., transmit power) for WAVE transmissions.

— User priority

  WAVE supports a variety of safety and nonsafety applications with up to eight levels of priority as defined in IEEE Std 802.11. The use of user priority (UP) and related access category (AC) supports quality of service using enhanced distributed channel access (EDCA) functionality specified in IEEE Std 802.11.

4.3.3 Management plane services

The MLME management services specified in Clause 6 comprise the following:

— Multi-channel synchronization

The MLME uses information derived locally and received over the air to provide a synchronization function with the objective of aligning channel intervals among communicating WAVE devices. The MLME provides the capability to generate Timing Advertisement (TA) frames to distribute system timing information and monitor received TA frames.

— Channel access

The MLME controls the access to specific radio channels in support of communication requests received from the WME.

— Vendor Specific Action frames

The MLME will accept incoming VSA frames and pass them to the WAVE Management Entity; the MLME will also generate VSA frames for transmission at the request of WME.

— Other IEEE 802.11 services

The MLME allows access to IEEE 802.11 services, which may be invoked on a per-channel basis.

— MIB maintenance

The MLME maintains a management information base (MIB) containing configuration and status information.

— Readdressing

The MLME allows device address changes to be triggered in support of pseudonymity.

5. Data plane services

5.1 General

WAVE devices shall communicate outside the context of an IEEE 802.11 basic service set (i.e., with *dot11OCBEnabled* equal true), as specified in IEEE Std 802.11p.

An example of a WAVE multi-channel MAC internal architecture is shown in Figure 2. This reference design architecture is used to specify the following transmit operations: channel routing, data queueing and prioritization, and channel coordination; however, actual implementations may depart from this example design. A WAVE MAC entity uses one or more instances of the MAC sublayer entity specified in IEEE Std 802.11, as amended by IEEE Std 802.11p, operating outside the context of a basic service set, with the extensions defined herein. There are two IEEE Std 802.11p MAC entities in the example design described in this clause: one for the CCH and one for the SCH. Data is prioritized according to access category (directly related to user priority), as indicated by the queues shown in Figure 2, which provide different contention and transmission parameters for different priority data frames. Prioritization is further specified in 5.4.

There are three types of information exchanged on the WAVE medium: management frames, data frames, and control frames. Control frames may be used per IEEE Std 802.11 and are not addressed in this standard. Management frames used by WAVE enter the data plane at the MAC layer. The primary management frames are the TA frame specified in IEEE Std 802.11p and the VSA frame specified in IEEE Std 802.11 and amended by IEEE Std 802.11p. Timing Advertisement frames are used to distribute time synchronization information. VSA frames are used to exchange management information such as WAVE Service Advertisements (WSAs). VSAs and TAs may be transmitted on any channel.

For higher layer data exchanges, WAVE supports both the WSMP specified in IEEE Std 1609.3 and IP version 6. Data frames containing WSMs (as indicated by the *Ethertype*) may be exchanged among devices on either the CCH or an SCH; however, data frames containing IP datagrams are only allowed on service channels. The management process of accessing the CCH or an SCH is specified in 6.3.

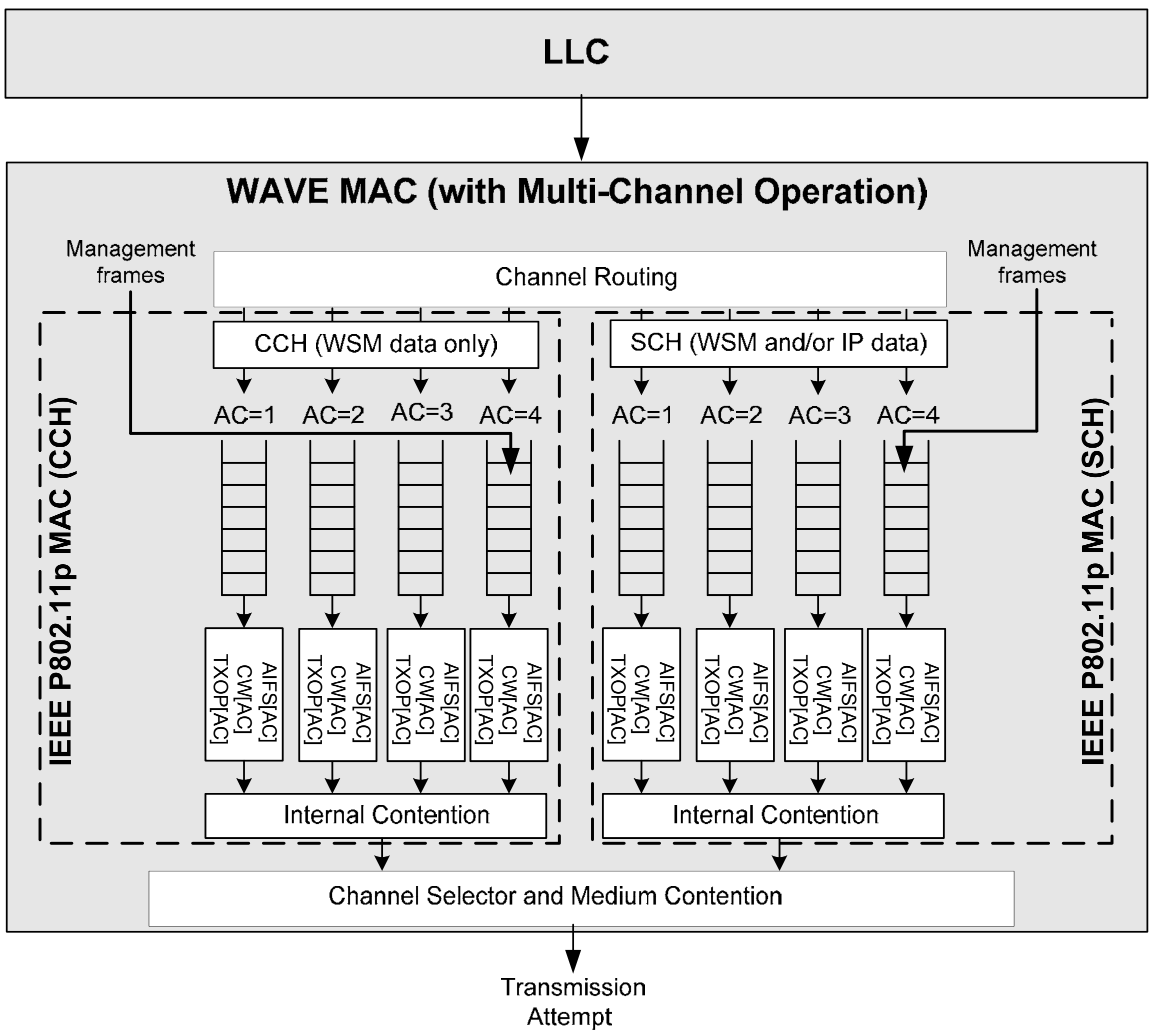


Figure 2 —Reference internal architecture of the transmit side of the MAC with channel coordination

5.2 Channel coordination

5.2.1 General

Channel coordination is designed to support data exchanges involving one or more switching devices with concurrent alternating operation on the CCH and an SCH. This allows, for example, a single-PHY device access to high-priority data and management traffic during the CCH interval, as well as general higher layer traffic during the SCH interval. Figure 3 illustrates two examples of channel access: continuous access, which requires no channel coordination, and alternating access, which does require channel coordination. Additional options of immediate and extended access are described in 6.3. CCH and SCH intervals are described in 5.2.2.

The channel coordination function also ensures that packets are transmitted on the intended RF channel.

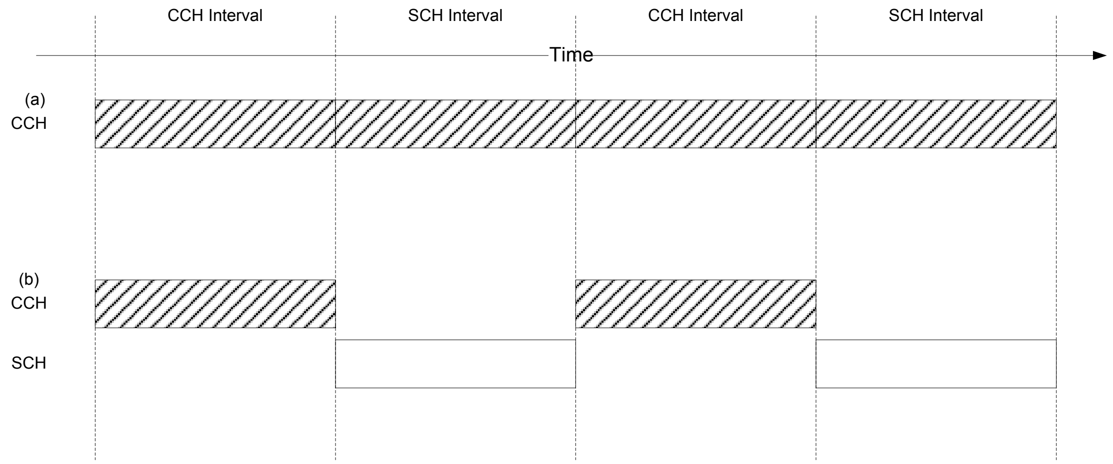


Figure 3 —Channel access examples: (a) continuous and (b) alternating

5.2.2 Channel intervals and guard interval

A sync interval and its CCH interval and SCH interval components are shown in Figure 4. The values are specified in Annex H. The duration of the CCH and SCH intervals are stored in the MIB attributes *CchInterval* and *SchInterval*, respectively, and the values of these attributes sum to the length of the sync interval. There shall be an integer number of sync intervals in 1 s. Coordinated Universal Time (UTC) defines the common time base for WAVE channel coordination. Subclause 6.2 includes requirements for switching devices to synchronize to UTC and a description of UTC estimation procedures. At each UTC second, the beginning of a sync interval shall align with the beginning of the UTC second, as shown in Figure 4.

At the beginning of each channel interval (CCH interval or SCH interval) is a guard interval, shown in Figure 4, used to account for radio switching and timing inaccuracies among different devices. See 6.2.5 for further specification of the guard interval.

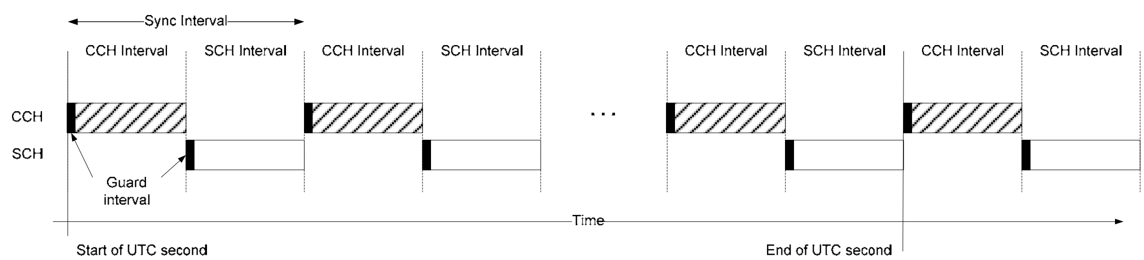


Figure 4 —Sync interval, guard interval, CCH interval, and SCH interval

5.2.3 Transmit restrictions on CCH and SCH

A WAVE device shall support transmit or receive operations, and it may support both. A WAVE device shall support at least one, and may support more than one, of the following IEEE 802.11 frame types:

- Management frame of type Timing Advertisement
- Management frame of type Action with subtype Vendor Specific
- Data frame, when carrying a WSM
- Data frame, when carrying an IP packet

This standard does not prohibit the transmission of other frame types. Any of the preceding frame types may be transmitted in either the CCH interval or the SCH interval. Frames carrying IP packets shall not be transmitted on the CCH and may be transmitted on an SCH. Supported control and service channels are identified in the MIB, with the CCH specified in Annex H. Other frame types may be transmitted on either the CCH or an SCH.

5.3 Channel routing

5.3.1 General

Channel numbers available for use by the device are listed in the MIB attribute *ChannelSetTable*. *ChannelSetTable* shall contain operating classes as specified in IEEE Std 802.11p and IEEE P802.11REVmb. The MAC sublayer receives data for transmission from the LLC sublayer in the form of MA-UNITDATA.requests for IP data or MA-UNITDATAX.requests for WSMP data, as illustrated in Figure 5 and Figure 6. IP data and WSMP data are also distinguished by the *Ethertype* field of the subnetwork access protocol (SNAP) header within the *Data* field, which is used by the LLC on the receiving side to deliver the data to the proper higher layer protocol.

For WSMs, the channel, transmit power, and data rate are set by higher layers on a per-message basis. For IP datagrams, the channel, transmit power, and data rate to be used are stored in a transmitter profile.

The channel routing for data and management frames is specified in 5.3.2 through 5.3.6.

5.3.2 Overview of data frame transmit operations

Figure 5 illustrates the contents of the MA-UNITDATAX.request from the LLC to the MAC for WSMP data. *Ethertype* is passed to the MAC in the SNAP header. Channel number, data rate, transmit power level, destination MAC address, and priority are passed to the MAC through parameters in the MA-UNITDATAX.request. A similar example for IP data is shown in Figure 6. A transmitting WAVE device shall support WSMP traffic, IP traffic, or both.

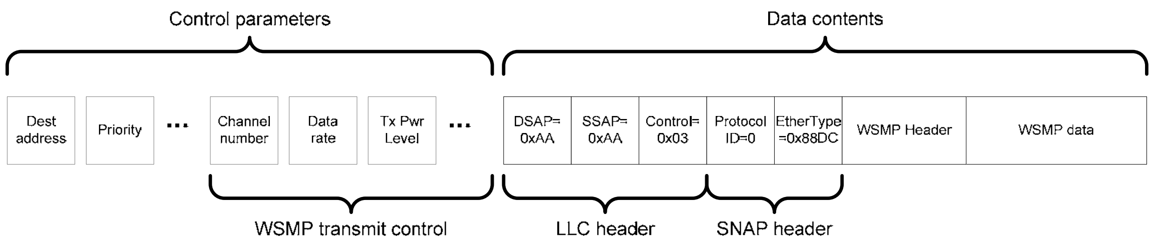


Figure 5 —MA-UNITDATAX.req fields for WSMP data

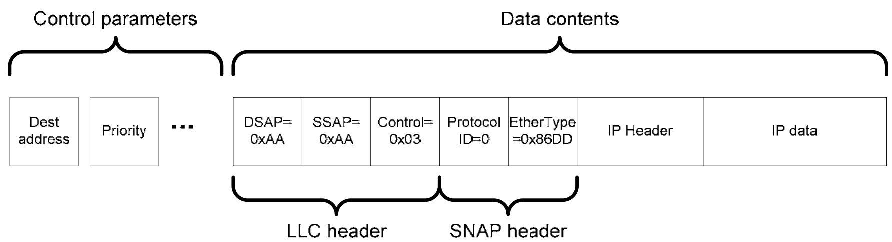
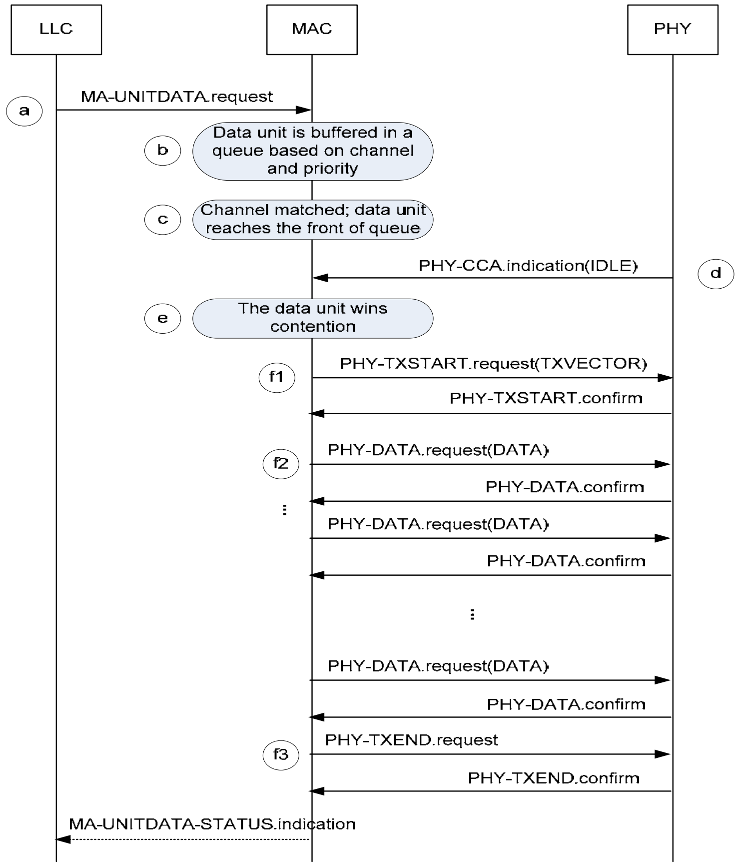


Figure 6 —MA-UNITDATA.req fields for IP data

An example IEEE 802.11 process flow for transmitting a packet is illustrated in Figure 7. MAC-PHY interactions are specified in IEEE Std 802.11. The MAC controls the data rate and power level used by the PHY via values in the *TXVECTOR* of the PHY-TXSTART.request.

Per IEEE Std 802.11 specification of Transmit Power Control procedures, the MLME prevents transmit power from exceeding the maximum allowed power. The maximum allowed power for a channel is stored in the IEEE 1609.3 MIB, either in attribute *UserAvailableTransmitPowerLevel* or attribute *ProviderChannelInfoTransmitPowerLevel*.



a) The LLC passes an MSDU to the MAC via an MA-UNITDATA.request or MA-UNITDATAX.request.

b) The MAC channel routing function examines the *Ethertype* field of the MSDU:

1) If the *Ethertype* field indicates WSMP, the MAC assigns the MSDU to the access category (see 5.4) based on the user priority from the MA-UNITDATAX.request. The MAC combines the MSDU with MAC protocol information (see IEEE Std 802.11) to form an MAC protocol data unit (MPDU) and places the MPDU in a queue appropriate for the assigned access category and the channel parameter passed in the primitive.

2) If the *Ethertype* field indicates IP, the MAC assigns the MSDU to an access category (see 5.4) based on the user priority. The MAC combines the MSDU with MAC protocol information (see IEEE Std 802.11) to form an MPDU and places the MPDU in a queue appropriate for the assigned access category and the SCH obtained from the transmitter profile.

c) The set of data queues associated with the current channel is identified.

1) If the frame contains a WSM, then the transmit power and data rate received in the MA-UNITDATAX.request are used.

2) If the frame contains an IP datagram, then the transmit power and data rate stored in the MIB attribute *TransmitterProfileTable* are used.

d) The PHY reports a clear channel to the MAC by issuing a PHY-CCA.indication (*IDLE*).

e) The MAC selects the MPDU that wins the internal contention on the current channel via EDCA.

f) The values of the power level and data rate are set in a *TXVECTOR*. The procedure to transfer an MPDU between the MAC and PHY is specified in IEEE Std 802.11, which is summarized as follows:

1) The MAC issues a PHY-TXSTART.request (*TXVECTOR*) to the PHY. A confirmation is issued back to the MAC after the PHY sets the desired power level and data rate.

2) The data is then exchanged between MAC/PHY through a series of PHY-DATA.request (*DATA*) primitives issued by the MAC, and PHY-DATA.confirm primitives issued by the PHY.

3) After the transmission of the final bit of the last MPDU octet, the transmission is terminated by the MAC through the primitive PHY-TXEND.request.

g) If transmission of an MPDU is not completed by the time the current channel interval ends, the transmission may be canceled by the MAC through the primitive PHY-TXEND.request. This premature termination can be avoided via implementing the safeguards described in Annex C.

Figure 7 —Example transmit process flow

5.3.3 Overview of data frame receive operations

When the MAC receives an MPDU, it extracts the MSDU and passes it to the LLC. It uses an MA-UNITDATA.indication primitive regardless of whether the frame contains a WSM or an IPv6 packet. If the PHY is in the process of receiving a data frame when channel switching occurs, causing an interruption of the reception, the PHY issues a PHY-RXEND.indication (*RXERROR* = CarrierLost) to the MAC sublayer, as specified in IEEE Std 802.11. The corresponding incomplete frame is discarded by the MAC (i.e., treated as a receive error), as specified in IEEE Std 802.11.

5.3.4 Routing for WSMP data

The MA-UNITDATAX.request primitive (7.4.2) includes the *priority*, *Channel Identifier*, *TxPwr_Lvl*, and *Data Rate* associated with the WSMP data packet. This permits a higher layer to control these PHY transmit parameters for each individual packet.

When the WSMP data is passed in an MA-UNITDATAX.request from the LLC to the MAC, the MAC shall route the packet to a proper queue corresponding to the *Channel Identifier* and *priority*. The data packet shall not be queued for transmission in the case that the *Channel Identifier* does not refer to a valid channel. A channel is invalid if it does not correspond to a channel for which the MLME is providing access via one of the mechanisms specified in 6.3.

For WSMP data, if *Expiry Time* is present in the MA-UNITDATAX.request, and the MAC supports *Expiry Time*, then the MAC sublayer shall not transmit the data if it has not been transmitted before the *Expiry Time*.

5.3.5 Routing for IP datagrams

Sending IP data requires that a transmitter profile first be registered in the MIB attribute *TransmitterProfileTable* via an MLMEX-REGISTERTXPROFILE.request or other mechanism. The transmitter profile contains an SCH number, power level, and the adaptable status of power level and data rate. The priority is set by the higher layer for each MSDU.

When IP data is passed in an MA-UNITDATA.request from the LLC to the MAC, the MAC shall route the packet to a data queue that corresponds to the SCH and requested priority. If there is no transmitter profile registered for the SCH, or if the MLME is not providing access to the SCH, the data shall be discarded.

5.3.6 Routing for management data

Routing of management information received in the form of MLME_VSPECIFIC.indication, whose *Organization Identifier* most significant bits equal 0x 00 50 C2 4A 4 (indicating IEEE 1609 usage), is specified in 6.4.2.

Routing of timing information received in the form of an MLME-TIMING_ADVERTISEMENT.indication is specified in 6.2.7. The recipient of the information is not specified in this standard.

Management frame transmissions shall be controlled by the data rate and power parameters in the MIB attribute *ChannelSetTable* entry for that channel.

5.4 User priority

5.4.1 General

The WAVE MAC shall use the IEEE 802.11 EDCA mechanism per channel as described in the following subclauses. The general architecture of prioritized access for data transmission is illustrated in Figure 2. When an MA-UNITDATA.request or MA-UNITDATAX.request arrives at the MAC sublayer, and the appropriate channel routing is completed (see 5.3), the MAC shall queue the data by mapping its UP (range, 0 to 7) to AC (range, 1 to 4) and access category index (ACI; range, 0 to 3). These mappings are specified in IEEE Std 802.11. Management frames are assigned the highest AC (i.e., 4, or AC_VO) per IEEE Std 802.11. Service providers may announce SCH EDCA parameters via the service advertisement mechanism described in IEEE Std 1609.3.

Each access category has an independent channel access function, with priority access governed by the appropriate EDCA parameter set values.

The values for a WAVE EDCA parameter set may differ from the default set specified in IEEE Std 802.11. These EDCA parameters are described here for convenience:

- Arbitration inter-frame space: The minimum time interval between the wireless medium becoming idle and the start of transmission of a frame.
- Contention window (CW): An interval from which a random number is drawn to implement the random transmit back-off mechanism.
- Transmit opportunity (TXOP) limit: The maximum time duration (in milliseconds) for which a station can transmit after obtaining a TXOP. If the TXOP limit is 0, the transmit opportunity only permits the transmission of a single MPDU.

The internal contention algorithm according to IEEE Std 802.11 calculates the back-off independently for each AC based on access parameters. The AC with the smallest back-off wins the internal contention, and the winning AC then contends externally for the wireless medium.

5.4.2 Control channel priority

The default EDCA parameter set specified in IEEE Std 802.11p for OCBEnabled operation is optimized for short message transfer and is recommended to be used when operating on the CCH. (See IEEE Std 802.11p specification of the EDCA Parameter Set element.)

5.4.3 Service channel priority

Both IP and WSMP data packets, as well as management frames, are permitted on an SCH. On an SCH, if an EDCA parameter set is received in the MLMEX-SCHSTART.request, this EDCA parameter set shall be used for transmissions on this SCH; otherwise, the default EDCA parameter set specified in IEEE Std 802.11p shall be used.

6. Management plane services

6.1 General

This standard specifies an extension to the MAC sublayer management entity (MLME) specified in IEEE Std 802.11 and IEEE Std 802.11p. It includes time synchronization and channel access features in support of channel coordination, as well periodic transmission of management frames.

6.2 Multi-channel synchronization

6.2.1 General

The synchronization function specified in this clause is based on a common time reference that allows WAVE devices to perform the channel coordination function specified in 5.2. Devices without a local timing source can acquire timing information over the air from other WAVE devices. Over-the-air synchronization procedures utilize the *Timestamp* field and *Time Advertisement* information element in the Timing Advertisement frame as inputs in estimating UTC time. Refer to IEEE Std 802.11 for the format of the *Timestamp* field, which indicates the value of the sender's timing synchronization function (TSF) timer. The value of the TSF timer is an integer modulo 2^{64}, incremented in units of microseconds. The *Time Advertisement* information element provides an offset from sending device TSF timer to general UTC time. Refer to IEEE Std 802.11p for the format of the *Time Advertisement* information element.

Synchronization to UTC, as specified in 6.2.5, is mandatory for WAVE devices switching channels on channel interval boundaries. The timing information used for synchronization may be derived by a timing management function from information received over the air from other WAVE devices, or may be obtained from a local time reference. The MLME maintains the estimated UTC time used for channel coordination; the timing management function that derives the UTC estimate may be internal or external to the MLME. An external timing management function may set the MLME UTC estimate (and its error characteristics) via the MLMEX-SETUTCTIME.request.

6.2.2 Common time base definition

The synchronization function uses UTC as the common time base. UTC is defined in ITU-R TF.460 [B2] and can be obtained from many sources including Global Positioning System (GPS). GPS receivers typically provide a precise 1 pulse per second (PPS) UTC signal (with an error less than 100 ns), and these precise 1 PPS signals can be used for timing and synchronization. The MLME shall use its UTC estimate to determine channel interval timing.

6.2.3 Common time base estimation

When performing synchronization, the MLME synchronizes to UTC time by implementing an estimator of UTC time and an estimator of the standard deviation of the error in the estimate of UTC time. This permits the synchronization of transmissions with channel intervals specified in 6.2.5. Standard deviation is defined as the square root of the second moment of the probability density function of the estimation error. Note this includes the effect of any biases in the estimate of UTC time. For example, GPS can be used as an input to a simple estimator that calculates the necessary TSF timer offset value and sets the standard deviation (see 6.2.4) to that of the standard deviation of the time output of the GPS device under the given operating conditions. A slightly more complex implementation could use information from the *Time Advertisement* information element in received Timing Advertisement frames (see 6.2.4) to update an internal estimator of UTC time along with an estimation error as outlined in Annex F.

6.2.4 Timing information

The *Time Advertisement* information element is carried within a Timing Advertisement frame and contains information that can be used by recipients to estimate UTC. This element has a *Timing Capabilities* subfield, a *Time Value* subfield, and a *Time Error* subfield as specified in IEEE Std 802.11p. The recipient of a Timing Advertisement frame can use the Timing Advertisement frame transmission *Timestamp* field, the *Time Value*, and the *Time Error*, along with the local TSF time at reception *Local Time*, to determine the relationship between UTC and local TSF time at *Local Time* point in time, which can then be used with local TSF time to provide an estimate of UTC at any time (e.g., using a procedure similar to that outlined in Annex F).

The *Timestamp* field conveys the TSF timer value of the transmitting device.

The *Timing Capabilities* subfield is used to indicate the transmitting device's source of external time.

The *Time Value* subfield specified in IEEE Std 802.11p, in conjunction with the *Timestamp* gives the receiving STA an estimate of the time standard at the time the frame was transmitted.

The *Time Error* subfield indicates the standard deviation of the error in the *Time Value*.

Local Time is returned in the MLME-TIMING_ADVERTISEMENT.indication.

The current local TSF timer value may be accessed via the IEEE 802.11 MLME-GETTSFTIME primitive.

When no valid time estimate is available at a device sending a TA frame (e.g., at startup), the *Time Value* shall be set to zero and the *Time Error* shall be set to its maximum value. Note that once any higher quality (lower variance) source of UTC time becomes available, that source and its error variance can be adopted for the UTC estimator. The value of the *Time Error* (i.e., the estimated uncertainty of the *Time Value*), is generally a function of various environmental and implementation factors and may vary with time.

6.2.5 Sync tolerance

Each CCH and SCH interval includes an initial guard interval as specified in 5.2. As shown in Figure 8, the guard interval is the sum of MIB system parameters *SyncTolerance* and *MaxChSwitchTime* (stored in the MIB per Annex E, with values specified in Annex H). *SyncTolerance*/2 is the value used to determine whether a WAVE device is synchronized to UTC or not, as described in the subsequent discussion. *MaxChSwitchTime* is the maximum time allowed for any WAVE device operating with multi-channel synchronization to change channels.

During the guard interval, a switching device might not be available for communication, as its switching PHY could be in a transition state between channels. For a device switching channels on channel interval boundaries, at the beginning of a guard interval, the MAC activities on the previous channel may be suspended; at the end of the guard interval, the prioritized access activities on the next channel shall be started, or resumed if they were suspended. To prevent multiple switching devices from attempting to transmit simultaneously at the end of a guard interval, a medium busy shall be declared during the guard interval so that all transmission attempts are subject to a random back-off at the start of each channel interval (see Annex B).

Figure 8 illustrates the components of the guard interval. *MaxChSwitchTime* represents the time when a switching device's radio might be in an unusable state, unable to transmit or receive. *SyncTolerance*/2 accounts for the fact that communicating devices may not be precisely aligned in time, and a second device may be switching during any portion of a first device's guard interval (but should not be switching outside the first device's guard interval). For a device switching channels, receive operation may be performed whenever the device's radio supports it (i.e., when it is not switching), but transmit operation is only performed outside the guard interval.

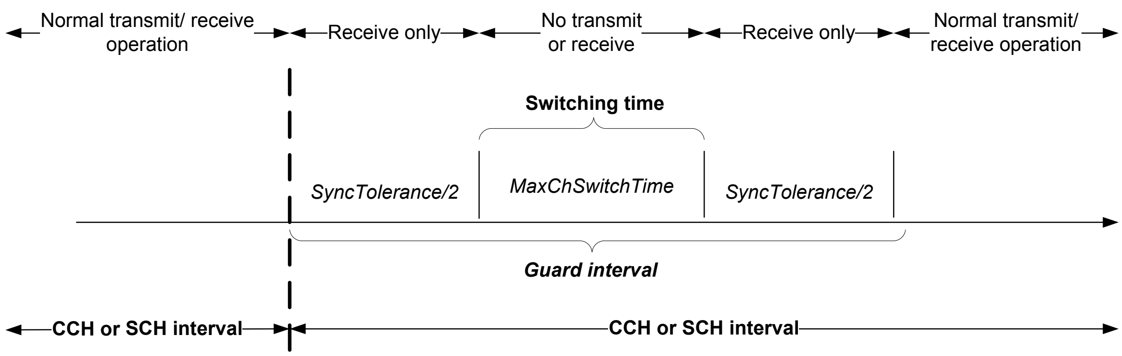


Figure 8 —Guard interval, sync tolerance, and max channel switch time

A WAVE device is defined to be synchronized to UTC if three times the value of its *Time Error* is less than *SyncTolerance*/2. When not synchronized to UTC, the MLME shall not provide alternating channel access (or immediate channel access when used without extended access) as specified 6.3.

6.2.6 Sending timing advertisements

Sending Timing Advertisement frames is optional. On receipt of an MLMEX-TA.request, the MLME may begin to generate MLME-TIMING_ADVERTISEMENT.request primitives at the rate indicated by *Repeat Rate* in the MLMEX-TA.request, as illustrated in Figure 9. Timing Advertisement frames shall be transmitted on the channel and in the channel interval indicated in the MLMEX-TA.request. If an external time management function has calculated the UTC offset, it is provided to the MLME in the MLMEX-TA.request; otherwise, the timing information is calculated by the MLME.

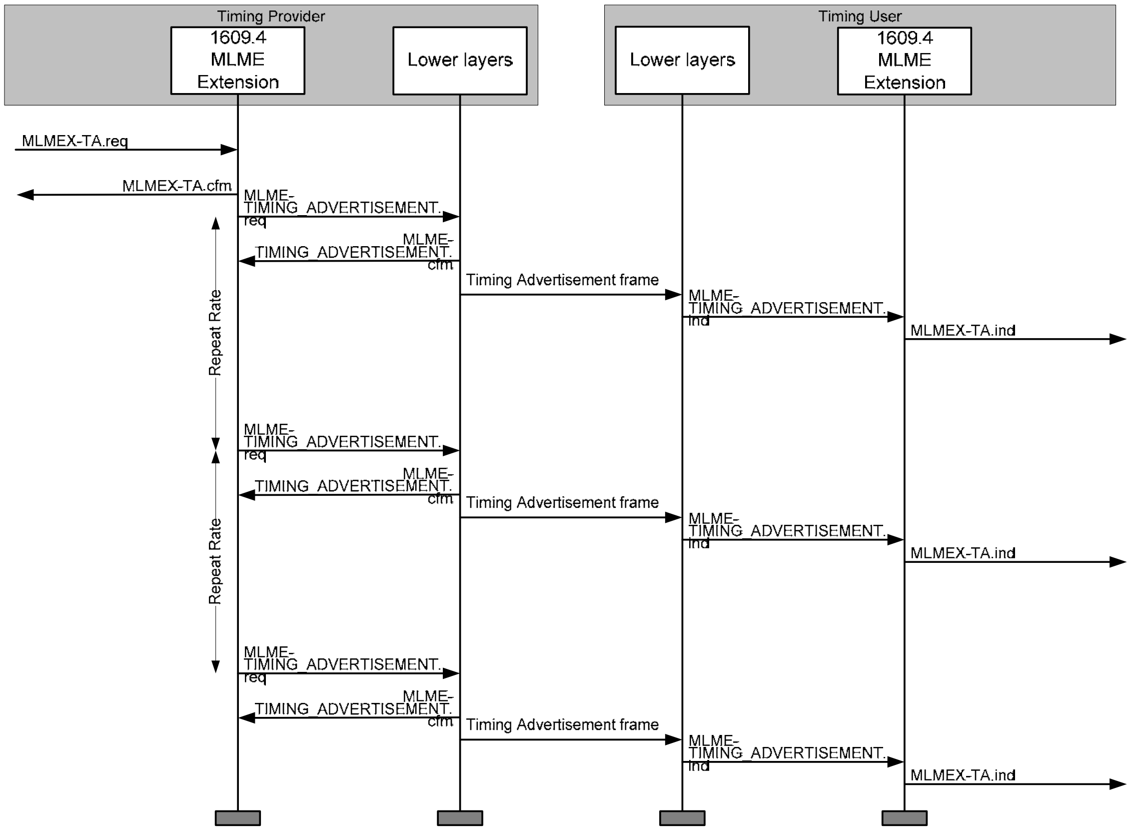


Figure 9 —Sending timing information

MLME-TIMING_ADVERTISEMENT.request parameters are set as specified in Table 1.

Table 1—MLME-TIMING_ADVERTISEMENT.request parameters

Parameter	Value
PeerMACAddress	Set to *Destination MAC address* received in the MLMEX-TA.request.
Capability	If received in the MLMEX-TA.request, set per the *TimingAdvertisementContents*; otherwise set as specified in IEEE Std 802.11 and IEEE Std 802.11p.
Time Advertisement	If received in the MLMEX-TA.request, set per the *TimingAdvertisementContents*; otherwise set per 6.2.4.
Country	If received in the MLMEX-TA.request, set per the *TimingAdvertisementContents*; otherwise set as specified in IEEE Std 802.11 and IEEE Std 802.11p.
Power Constraint	If received in the MLMEX-TA.request, set per the *TimingAdvertisementContents*; otherwise set as specified in IEEE Std 802.11 and IEEE Std 802.11p.
Extended Capabilities	If received in the MLMEX-TA.request, set per the *TimingAdvertisementContents*; otherwise not used
VendorSpecificInfo	If received in the MLMEX-TA.request, set per the *TimingAdvertisementContents*; otherwise not used.

6.2.7 Receiving timing information

Processing received time advertisements is optional. If supported, on receipt of an MLME-TIMING_ADVERTISEMENT.indication, the MLME shall generate an MLMEX-TA.indication primitive, as illustrated in Figure 9. MLMEX-TA.indication parameters are set to the values received in the MLME-TIMING_ADVERTISEMENT.indication, if present. Additionally, MLME indicates the channel on which the frame was received via the *Channel Identifier*.

The MLME may use the *Timestamp*, *Time Advertisement*, and *Local Time* received in the MLME-TIMING_ADVERTISEMENT.indication to generate an estimate of UTC time, as described in 6.2.3 and 6.2.4.

6.3 Channel access

6.3.1 General

The MLME provides access to specific radio channels under control of the WME. Channel access options include continuous access, alternating service channel and CCH access, immediate SCH access, and extended SCH access, as illustrated in Figure 10 and specified in 6.3.2 through 6.3.4. Immediate and extended options may be combined in a single access.

Service channel access allows higher layers to exchange data on a designated service channel. Channel access operations are specified here in the context of single-PHY devices. Multi-PHY devices may use modified operations as needed to achieve comparable results.

Figure 10—Channel access options: (a) continuous, (b) alternating, (c) immediate, and (d) extended

As specified in 6.3.2 through 6.3.5, the MLME accomplishes channel switching by sending an IEEE Std 802.11 PLME-SET.request with *dot11CurrentFrequency* set equal to the desired channel. Furthermore, if the current and new channels have different *Channel Starting Frequency* or *Channel Spacing* (e.g., are in different operating classes as specified in IEEE Std 802.11 and IEEE P802.11REVmb annex on country elements and operating classes), the MLME additionally sends PLME-SET.request setting the appropriate *dot11ChannelStartingFactor* and/or *dot11PhyOFDMChannelWidth*. This exchange is illustrated in Figure 11.

6.3.2 Alternating service channel access

On receipt of an MLMEX-SCHSTART.req, the MLME shall begin to provide access to the indicated *Channel Number* as follows.

Unless otherwise indicated by *ExtendedAccess* as specified below, at the beginning of each CCH interval, the MLME sends a PLME-SET.request with *dot11CurrentFrequency* set equal to the channel number of the CCH as determined by the MIB attribute *ChannelSetTable*.

Unless otherwise indicated by *ImmediateAccess* or *ExtendedAccess* as specified below, at the beginning of each SCH interval, the MLME sends a PLME-SET.request with *dot11CurrentFrequency* set equal to the *Channel Number* from the MLMEX-SCHSTART.request. This exchange is illustrated in Figure 11.

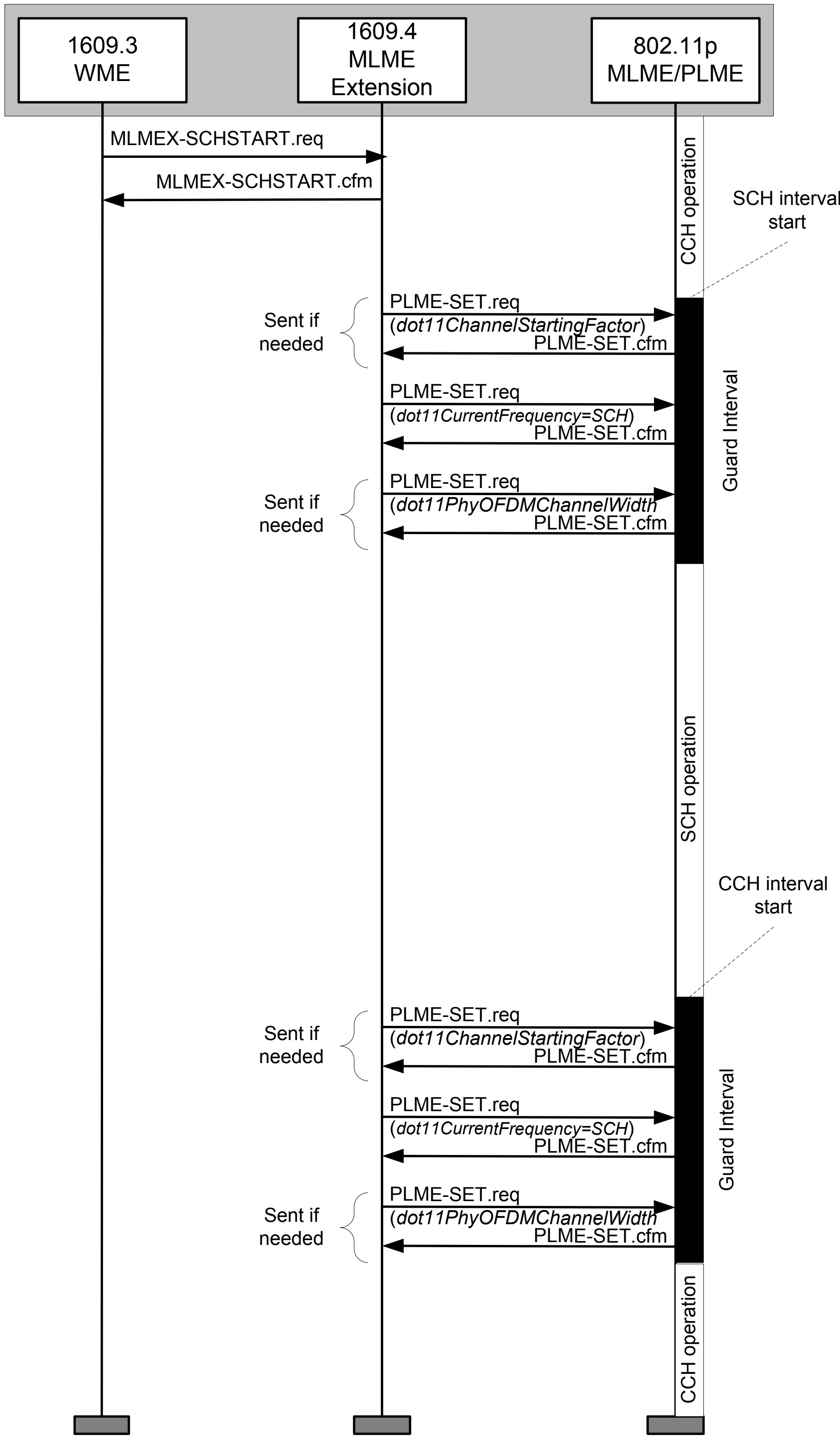


Figure 11—Alternating service channel access (timeline not to scale)

6.3.3 Immediate SCH access

Immediate SCH access allows immediate communications access to the SCH without waiting for the next SCH interval.

On receipt of an MLMEX-SCHSTART.request with *ImmediateAccess* set to True, the MLME shall begin to provide access to the indicated *Channel Number* as follows.

As soon as possible, the MLME sends a PLME-SET.request with *dot11CurrentFrequency* set equal to the *Channel Number* from the MLMEX-SCHSTART.request as illustrated in Figure 12.

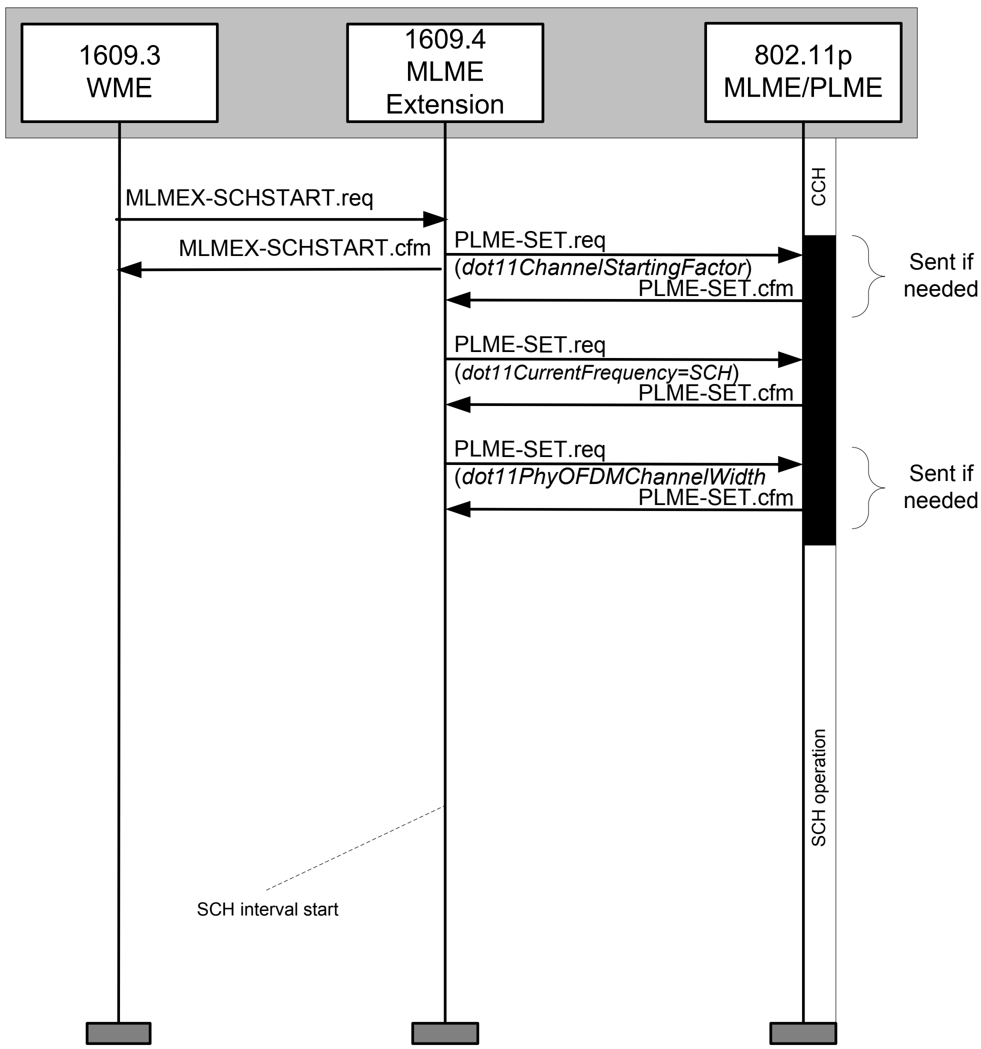


Figure 12 —Immediate SCH access

6.3.4 Extended SCH access

Extended SCH access allows communications access to the SCH without pauses for CCH access.

On receipt of an MLMEX-SCHSTART.request with *ExtendedAccess* greater than 0 (see 7.3.4.2), the MLME shall begin to provide access to the indicated *Channel Number* as follows.

The MLME sends a PLME-SET.request with *dot11CurrentFrequency* set equal to the *Channel Number* from the MLMEX-SCHSTART.req as illustrated in Figure 13. Unless a new MLMEX-SCHSTART.request or MLMEX-SCHEND.request is received, the MLME does not send any more PLME-SET.requests for a period of *ExtendedAccess* channel intervals, after which time the MLME returns to continuous or alternating service channel access as applicable.

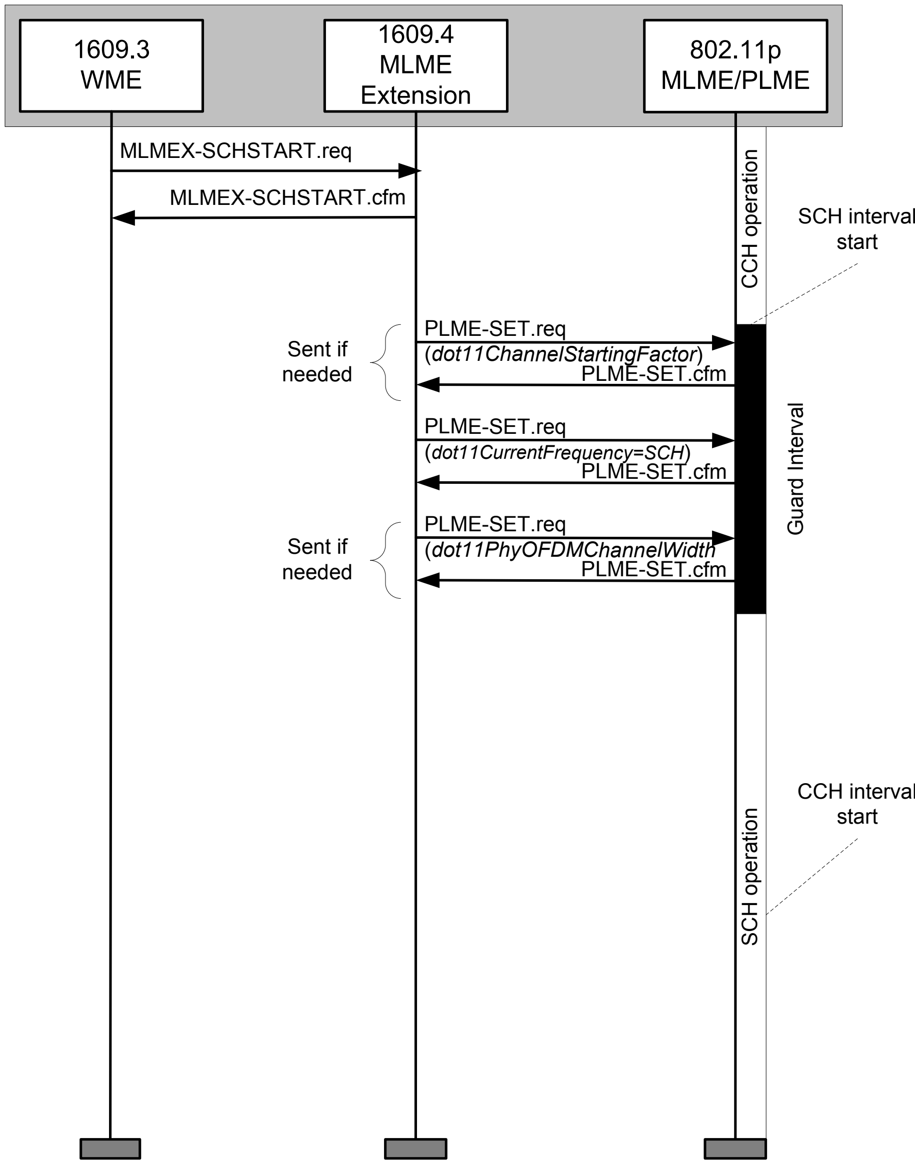


Figure 13—Extended SCH access

6.3.5 Service channel release

On receipt of an MLMEX-SCHEND.request, the MLME shall cease providing access to the indicated *Channel Number* as follows. The MLME sends a PLME-SET.request with *dot11CurrentFrequency* set equal to the CCH as determined by the MIB attribute *ChannelSetTable*. If the MLME unilaterally releases an SCH (e.g., due to loss of sync), it sends an MLMEX-SCHEND.indication to indicate the reason. Whether or not the release occurs on a CCH Interval boundary is not specified.

6.4 Vendor Specific Action frames

6.4.1 VSA transmission

6.4.1.1 Starting VSA transmission

Transmission of VSAs is optional. Receipt of an MLMEX-VSA.request causes the IEEE 1609.4 MLME to generate MLME-VSPECIFIC.request primitives, which in turn cause VSA frames to be transmitted per IEEE Std 802.11. Vendor Specific Action frames shall be transmitted on the channel and in the channel interval indicated in the MLMEX-VSA.request.

On receipt of an MLMEX-VSA.request, the MLME shall begin to generate VSAs as follows. If *Destination MAC Address* in the MLMEX-VSA.request indicates a unicast address, the MLME generates one MLME-VSPECIFIC.request primitive. Otherwise, the MLME generates a number of MLME-VSPECIFIC.request primitives at the *Repeat Rate* received in the MLMEX-VSA.request, as illustrated in Figure 14.

MLME-VSPECIFIC.request parameters are set as shown in Table 2.

Table 2—MLME-VSPECIFIC.request parameters

Parameter	Value
PeerMacAddress	Set to the *Destination MAC Address* from the MLMEX-VSA.request.
Organization Identifier	Set as specified in the text that follows the table.
Vendor Specific Content	Set to the *Vendor Specific Content* field from the MLMEX-VSA.request.

Organization Identifier is set to the *Organization Identifier* from the MLMEX-VSA.request, if present. Otherwise, *Organization Identifier* is set to 0x 00 50 C2 4A 4z, where 0x00 50 C2 4 indicates IEEE 1609 and *z* is set to the 4-bit *Management ID* from the MLMEX-VSA.request. *Management ID* values are specified in IEEE P1609.0.

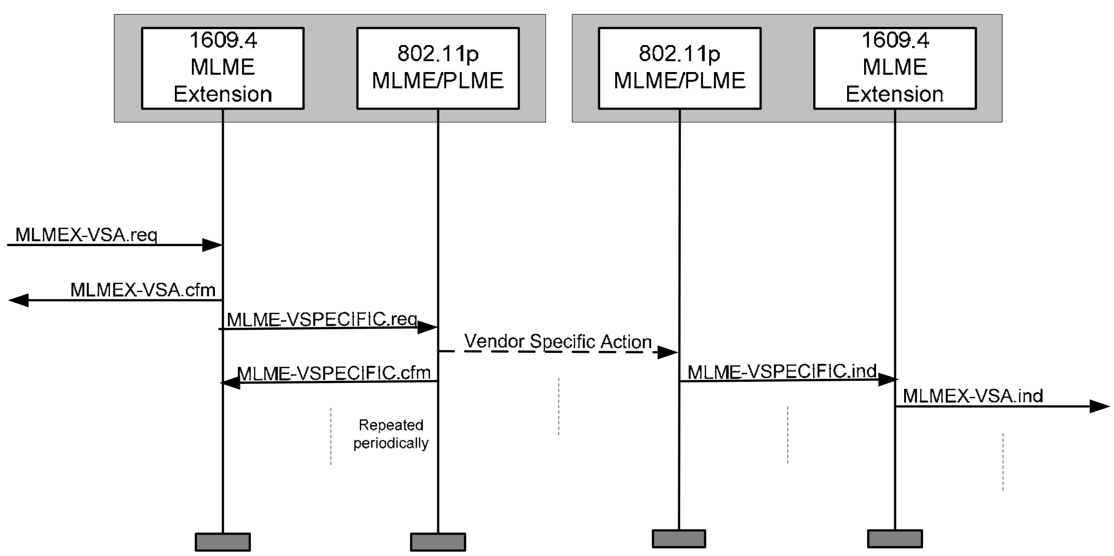


Figure 14—VSA transmission and delivery

6.4.1.2 Ending VSA transmission

When generating repeated VSAs, on receipt of an MLMEX-VSAEND.request, the MLME shall cease generating MLME-VSPECIFIC.requests. If the MLME is generating multiple different VSAs, an implementation specific mechanism may be used to identify the VSA being ended.

6.4.2 VSA reception

Processing of received VSAs is optional. Receipt of Vendor Specific Action frames with the most significant bits of *Organization Identifier* equal to 0x 00 50 C2 4A 4 are indicated to the IEEE 1609.4 MLME via the MLME-VSPECIFIC.indication per IEEE Std 802.11. The MLME in turn generates an MLMEX-VSA.indication as illustrated in Figure 14. On receipt of an MLME-VSPECIFIC.indication, for implementations supporting this feature, the MLME shall generate an MLMEX-VSA.indication as specified in Table 3.

Table 3—MLMEX-VSA.indication parameters

Parameter	Value
Source MAC Address	Set to the *PeerMACAddress* from the MLME-VSPECIFIC.indication.
Vendor Specific Content	*VendorSpecificContent* from the MLME-VSPECIFIC.indication.
RCPI	Set to Received Channel Power Indicator (*RCPI*) in the MLME-VSPECIFIC.indication.
Channel Identifier	The channel on which the frame was received.
Management ID	Set to 4 least significant bits of *Organization Identifier* in the MLME-VSPECIFIC.indication. *Management ID* values are specified in IEEE P1609.0.

MLME-VSPECIFIC.indication primitives with *Organization Identifier* not indicating IEEE 1609 are not delivered to the IEEE 1609.4 MLME, but rather to the entity indicated by the *Organization Identifier.*

6.5 Other IEEE 802.11 services

The MLME may support the sending of primitives that invoke channel-specific IEEE 802.11 services as specified here. This allows an external entity to trigger the transmission of a management frame, such as those used for link quality assessment, on a specific channel. Note that those IEEE 802.11 services that are specific to a particular channel, e.g., those that result in the transmission of a frame, may call for additional implementation-specific processing beyond that specified here.

On receipt of an MLMEX-SendPrimitive.request, the IEEE 1609.4 MLME generates the indicated IEEE 802.11 service request primitive. Figure 15 shows an example of this capability in the case of an IEEE 802.11 Measurement Request, and its resulting Measurement Report.

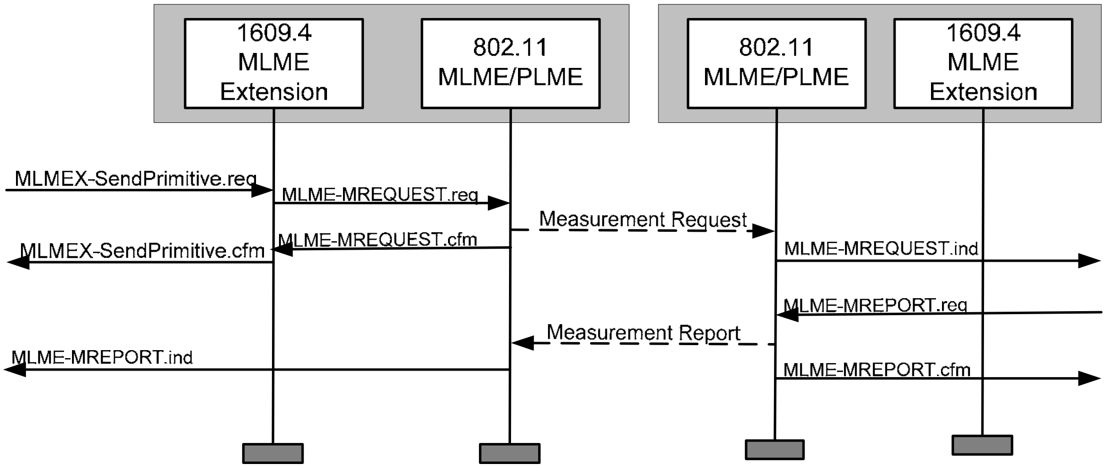


Figure 15—Example of invoking an IEEE 802.11 management frame

6.6 MIB maintenance

In a managed WAVE device, the MLME shall maintain a management information base (MIB) containing the configuration and status information identified in Annex D and specified in Annex E (Figure 16). The MIB is accessible to higher layers via the MLME-Get and MLME-Set primitives specified in IEEE Std 802.11. Additionally, the device may support other MIBs (e.g., MIBs containing vendor-specific information).

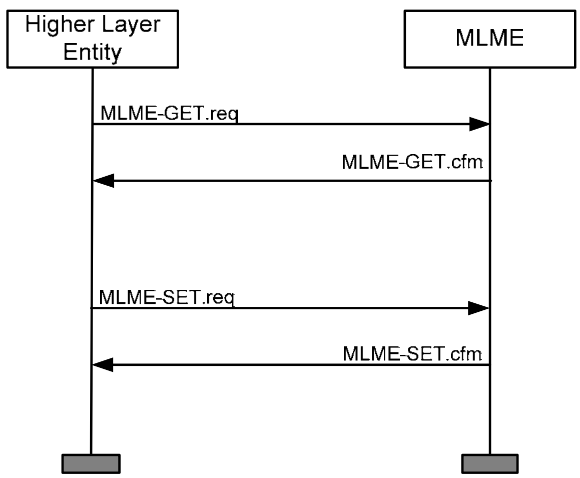


Figure 16—Information flow for MIB access

6.7 Readdressing

In support of pseudonymity, a device may support a readdressing feature. On receipt of an MLMEX-AddressChange.request, the MLME shall reset the device MAC address to a new value. If an MLMEX-AddressChange.request includes *MAC Address*, that value is used as the new address; otherwise, a randomly chosen address is used. The chosen address shall be a locally administered address as defined in IEEE Std 802. Note that an address change may disrupt any ongoing communications.

7. Service primitives

7.1 Overview of management model

Figure 17 depicts the relationships among data and management entities. For the purposes of describing the functionality specified by this standard, the MAC layer management entity is divided into two parts: The WAVE MAC management extension specified in this document and the MLME specified in IEEE Std 802.11. In implementation, these two entities would likely be combined into one MAC layer management entity. Management primitives specified in this standard have a prefix of MLMEX for convenience to the reader. Table 4 lists the primitives specified and invoked by this standard.

The implementation of the primitives and their exchange protocols are not specified but are left as design decisions.

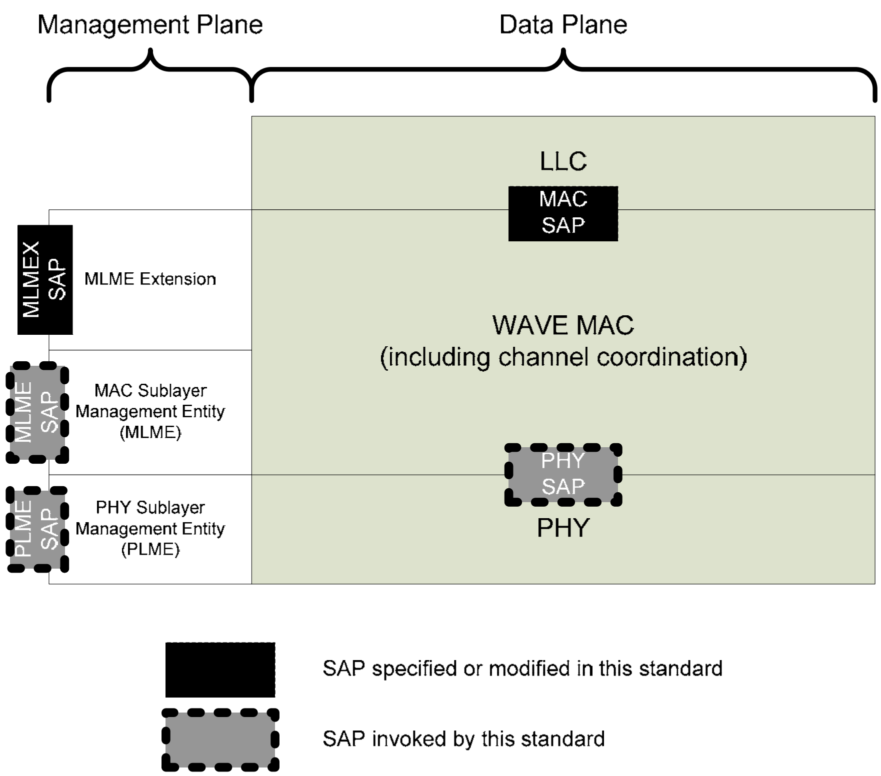


Figure 17 —SAP interfaces relevant to this standard

Table 4—Summary of primitives

SAP	Primitive	Specified in
MLME extension	MLMEX-TA	7.3.2
	MLMEX-TAEND	7.3.2
	MLMEX-VSA	7.3.3
	MLMEX-VSAEND	7.3.3
	MLMEX-SCHSTART	7.3.4
	MLMEX-SCHEND	7.3.4
	MLMEX-REGISTERTXPROFILE	7.3.5
	MLMEX-DELETETXPROFILE	7.3.5
	MLMEX-CANCELTX	7.3.6
	MLMEX-GETUTCTIME	7.3.7
	MLMEX-AddressChange	7.3.8
	MLMEX-SendPrimitive	7.3.9
MLME	MLME-GET	IEEE Std 802.11
	MLME-SET	IEEE Std 802.11
	MLME-VSPECIFIC	IEEE Std 802.11
	MLME-TIMING_ADVERTISEMENT	IEEE Std 802.11p
	MLME-GETTSFTIME	IEEE Std 802.11p
PLME	All	IEEE Std 802.11
MAC	MA-UNITDATA	IEEE Std 802.11
	MA-UNITDATAX	7.4
PHY	All	IEEE Std 802.11

7.2 Channel identification

IEEE Std 802.11 uses *Channel Number* (range, 0 to 200) to identify a specific radio channel within the context of a group of channels identified by *Operating Class* (see IEEE Std 802.11 and IEEE P802.11REVmb annex on country elements and operating classes) applicable to a geographic region (country) defined by *Country String* (see IEEE Std 802.11). The coordination of country and operating class among devices is outside the scope of this standard. *Channel Identifier* is used in the IEEE 1609 SAPs to indicate a fully specified channel, comprising *Country String*, *Operating Class,* and *Channel Number*. Where channel identification is sent over the air, its format is definitively specified.

7.3 WAVE MLME Extension SAP

7.3.1 General

Additional service primitives to those defined in the IEEE 802.11 and IEEE 802.11p MLME are used to manage operation. These service primitives are described in 7.3.2 through 7.3.9.

Where a mechanism is needed to identify one of multiple ongoing services (e.g., to end one of multiple active service channel accesses), the means is assumed to be implicit in the primitives, with the details left to the implementer. For example, an implementation could include an index parameter to distinguish among multiple services.

7.3.2 WAVE timing advertisement

7.3.2.1 General

The timing advertisement primitives control the sending and delivery of timing advertisements. The relationship between the MLMEX-TA primitives and the MLME-TIMING_ADVERTISEMENT primitives is described in 6.2.

7.3.2.2 MLMEX-TA.request

7.3.2.2.1 Function

The primitive is generated to request the MLME to begin transmission of timing advertisements.

7.3.2.2.2 Semantics of the service primitive

The primitive provides the following parameters:

MLMEX-TA.request (
Destination MAC address,
Repeat Rate,
Channel Identifier,
Channel Interval,
TimingAdvertisementContents
)

Table 5 specifies the MLMEXTA.request parameters.

Table 5—MLMEX-TA.request parameters

Name	Type	Valid range	Description
Destination MAC Address	MACAddress	Any valid address	The address of the peer MAC entity to which the TA is sent.
Repeat Rate	Integer	0–255	The number of Timing Advertisement frames to be transmitted per 5 s. A value of 0 indicates a single message is to be sent.
Channel Identifier	See 7.2	See 7.2	The channel (CCH or SCH) on which the Timing Advertisement frame is transmitted.
Channel Interval	Enumeration	CCH Interval, SCH Interval, both	The channel interval in which the transmissions are to occur.
TimingAdvertisementContents	As described in the text that follows the table.	As described in the text that follows the table.	As described in the text that follows the table.

TimingAdvertisementContents is optionally included and consists of the information needed to construct the MLME-TIMING_ADVERTISEMENT.request specified in IEEE Std 802.11p, including some or all of *Capability Information, Time Advertisement, Country, Power Constraint, Extended Capabilities,* and *VendorSpecificInfo.*

7.3.2.2.3 When generated

The primitive is generated to initiate the transmission of timing information.

7.3.2.2.4 Effect of receipt

Upon receipt of the primitive, the MLME schedules the transmission of timing information (see 6.2.6).

7.3.2.3 MLMEX-TA.confirm

7.3.2.3.1 Function

The primitive reports the result of an MLMEX-TA.request.

7.3.2.3.2 Semantics of the service primitive

The primitive provides the following parameter.

MLMEX-TA.confirm (
ResultCode
)

Table 6 specifies the MLMEXTA.confirm parameter.

Table 6—MLMEX-TA.confirm parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, NO SYNC, INVALID_PARAMETERS, UNSPECIFIED FAILURE	Reports the result of an MLMEX-TA.request.

7.3.2.3.3 When generated

The primitive is generated by the MLME in response to an MLMEX-TA.request.

7.3.2.3.4 Effect of receipt

The requester is notified of the result of an MLMEX-TA.request.

7.3.2.4 MLMEX-TA.indication

7.3.2.4.1 Function

This primitive indicates the receipt of a Timing Advertisement frame.

7.3.2.4.2 Semantics of the service primitive

This primitive provides the following parameters:

MLMEX-TA.indication (
Timestamp,
Local Time,
RCPI,
Source MAC address,
Channel Identifier,
TimingAdvertisementContents
)

Table 7 specifies the MLMEX-TA.indication parameters.

Table 7—MLMEX-TA.indication parameters

Name	Type	Valid range	Description
Timestamp	Integer	N/A	The timestamp of the received frame, as specified in IEEE Std 802.11p.
Local Time	Integer	N/A	The value of the station's TSF timer at the start of reception of the first octet of the *Timestamp* field of the received Timing Advertisement frame, per IEEE Std 802.11p.
RCPI	Integer	0–255	RCPI is the measured power value of the received WAVE management frame, per IEEE Std 802.11k.
Source MAC address	MAC address	Specified in IEEE Std 802.11	The address of the MAC entity from which the Timing Advertisement frame was received.
Channel Identifier	See 7.2	See 7.2	The channel on which the Timing Advertisement frame was received.
TimingAdvertisementContents	As described in the text that follows the table.	As described in the text that follows the table.	As described in the text that follows the table.

TimingAdvertisementContents contains the information received in the MLME-TIMING_ ADVERTISEMENT.indication specified in IEEE Std 802.11p, including some or all of *Capability Information, Time Advertisement, Country, Power Constraint, Extended Capabilities,* and *Vendor SpecificInfo.*

7.3.2.4.3 When generated

The primitive is generated by the MLME when a Timing Advertisement frame is received.

7.3.2.4.4 Effect of receipt

Upon receipt of this primitive, the recipient may process the information.

7.3.2.5 MLMEX-TAEND.request

7.3.2.5.1 Function

The primitive is generated to instruct the MLME to cease transmission of timing advertisements.

7.3.2.5.2 Semantics of the service primitive

The parameters of the primitive are as follows:

MLMEX-TAEND.request (
Channel Identifier
)

Table 8 specifies the MLMEX-TAEND.request parameter.

Table 8—MLMEX-TAEND.request parameters

Name	Type	Valid range	Description
Channel Identifier	See 7.2	See 7.2	The channel (CCH or SCH) on which the Timing Advertisement frame is transmitted.

7.3.2.5.3 When generated

The primitive is generated to instruct the MLME to cease transmission of timing advertisements on the indicated channel.

7.3.2.5.4 Effect of receipt

Upon receipt of the primitive, the MLME ceases transmission of timing advertisements and generates an MLMEX-TAEND.confirm.

7.3.2.6 MLMEX-TAEND.confirm

7.3.2.6.1 Function

The primitive is used to confirm the outcome of an MLMEX-TAEND.request.

7.3.2.6.2 Semantics of the service primitive

The parameters of the primitive are as in Table 9:

MLMEX-TAEND.confirm (
ResultCode
)

Table 9—MLMEX-TAEND.request parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, UNSPECIFIED FAILURE	Reports the result of an MLMEX-TAEND.request.

7.3.3 Vendor Specific Action frames

7.3.3.1 General

The VSA primitives support the transfer of management data between WAVE devices. The relationship between the MLMEX_VSA primitives and the MLME frame generation primitives is described in 6.4.

7.3.3.2 MLMEX-VSA.request

7.3.3.2.1 Function

The VSA request is generated to request the MLME to begin transmission of VSAs.

7.3.3.2.2 Semantics of the service primitive

The primitive provides the following parameters:

MLMEX-VSA.request (
Destination MAC address,
Management ID,
Organization Identifier,
Vendor Specific Content,
Repeat Rate,
Channel Identifier,
Channel Interval
)

Table 10 specifies the MLMEX-VSA.request parameters.

Table 10 —MLMEX-VSA.request parameters

Name	Type	Valid range	Description
Destination MAC Address	MACAddress	Any valid address	The address of the peer MAC entity to which the VSA is sent.
Management ID	Integer	0–15	Identifies the source of the data when the source is an IEEE 1609 entity. Values are specified in IEEE P1609.0.
Organization Identifier	As defined in IEEE Std 802.11p	As defined in IEEE Std 802.11p	Identifies the source of the data when the source is not an IEEE 1609 entity. See IEEE Std 802.11p.
Vendor Specific Content	Octet string	Unspecified	Information to be sent as vendor specific content.
Repeat Rate	Integer	0–255	The number of Vendor Specific Action frames to be transmitted per 5 s. A value of 0 indicates a single message is to be sent. If *Destination MAC Address* is an individual address, *Repeat Rate* is ignored.
Channel Identifier	See 7.2	See 7.2	The channel on which the transmissions are to occur.
Channel Interval	Enumeration	CCH Interval, SCH Interval, both	The channel interval in which the transmissions are to occur.

7.3.3.2.3 When generated

The VSA primitive is generated to start the transmission of Vendor Specific Action frames.

7.3.3.2.4 Effect of receipt

Upon receipt of the primitive, the MLME schedules the transmission of Vendor Specific Action frames (see 6.4).

7.3.3.3 MLMEX-VSA.confirm

7.3.3.3.1 Function

This primitive reports the result of an MLMEX-VSA.request.

7.3.3.3.2 Semantics of the service primitive

This primitive provides the following parameter.

MLMEX-VSA.confirm (
ResultCode
)

Table 11 specifies the MLMEX-VSA.confirm parameter.

Table 11 —MLMEX-VSA.confirm parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, NO SYNC, INVALID_PARAMETERS, UNSPECIFIED FAILURE	Reports the result of an MLMEX-VSA.request.

7.3.3.3.3 When generated

This primitive is generated by the MLME in response to an MLMEX-VSA.request.

7.3.3.3.4 Effect of receipt

The requester is notified of the result of the request.

7.3.3.4 MLMEX-VSA.indication

7.3.3.4.1 Function

This primitive indicates the receipt of a Vendor Specific Action frame.

7.3.3.4.2 Semantics of the service primitive

This primitive provides the following parameters:

MLMEX-VSA.indication (
Source MAC Address,
Management ID,
Vendor Specific Content,
Channel Identifier,
RCPI
)

Parameters are defined in Table 12.

Table 12—MLMEX-VSA.indication parameters

Name	Type	Valid range	Description
Source MAC address	MAC address	Specified in IEEE Std 802.11	The address of the MAC entity from which the management frame was received.
Management ID	Integer	0–15	Identifier of the originator of the data. Values are specified in IEEE P1609.0.
Vendor Specific Content	Octet string	Not specified	The data contents of the Vendor Specific Action frame.
Channel Identifier	See 7.2	See 7.2	The channel on which the frame was received.
RCPI	Integer	0–255	RCPI is the measured power value on the received management frame, per IEEE Std 802.11k.

7.3.3.4.3 When generated

The MLMEX-VSA.indication primitive is generated by the MLME when a VSA is received.

7.3.3.4.4 Effect of receipt

WME may take action on receipt of this primitive.

7.3.3.5 MLMEX-VSAEND.request

7.3.3.5.1 Function

The primitive is generated to instruct the MLME to cease transmission of VSAs.

7.3.3.5.2 Semantics of the service primitive

The parameters of the primitive are as follows:

MLMEX-VSAEND.request ( )

7.3.3.5.3 When generated

The primitive is generated to instruct the MLME to cease transmission of VSAs.

7.3.3.5.4 Effect of receipt

Upon receipt of the primitive, the MLME ceases transmission of VSAs and generates an MLMEX-VSAEND.confirm.

7.3.3.6 MLMEX-VSAEND.confirm

7.3.3.6.1 Function

The primitive is used to confirm the outcome of an MLMEX-VSAEND.request.

7.3.3.6.2 Semantics of the service primitive

The parameters of the primitive are as follows:

MLMEX-VSAEND.confirm (
ResultCode
)

Table 13 specifies the MLMEX-VSAEND.request parameter.

Table 13—MLMEX-VSAEND.request parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, UNSPECIFIED FAILURE	Reports the result of an MLMEX-VSAEND.request.

7.3.3.6.3 When generated

The outcome of an MLMEX-VSAEND.request is confirmed.

7.3.3.6.4 Effect of receipt

The recipient may take action on receipt.

7.3.4 WAVE service channel access

7.3.4.1 General

These primitives direct the MLME to allocate radio resources to the indicated service channel.

7.3.4.2 MLMEX-SCHSTART.request

7.3.4.2.1 Function

The MLMEX-SCHSTART.request primitive requests the MLME to provide access to the indicated SCH.

7.3.4.2.2 Semantics of the service primitive

The primitive provides the following parameters:

MLMEX-SCHSTART.request (
Channel Identifier,
OperationalRateSet,
EDCA Parameter Set,
ImmediateAccess,
ExtendedAccess
)

Table 14 specifies the MLMEX-SCHSTART.request parameters.

Table 14—MLMEX-SCHSTART.request parameters

Name	Type	Valid range	Description
Channel Identifier	See 7.2	See 7.2	The SCH to be made available for communications.
OperationalRateSet	Set of Integers	1–127	If present, as specified in IEEE Std 802.11.
EDCA ParameterSet	As specified in IEEE Std 802.11	As specified in IEEE Std 802.11	If present, as specified in IEEE Std 802.11.
ImmediateAccess	Boolean	TRUE or FALSE	Indicates that the MLME should provide immediate access to the SCH and not wait until the next SCH interval.
ExtendedAccess	Integer	0–255	Indicates that the MLME should provide continuous access (during both SCH interval and CCH interval) to the SCH for *ExtendedAccess* control channel intervals. A value of 255 indicates indefinite access.

7.3.4.2.3 When generated

The primitive is generated as needed.

7.3.4.2.4 Effect of receipt

Upon receipt of the primitive, the MLME begins providing the requested channel access.

7.3.4.3 MLMEX-SCHSTART.confirm

7.3.4.3.1 Function

The primitive reports the result of an MLMEX-SCHSTART.request.

7.3.4.3.2 Semantics of the service primitive

The primitive provides the following parameters, elaborated in Table 15:

MLMEX-SCHSTART.confirm (
ResultCode
)

Table 15 —MLMEX-SCHSTART.confirm parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, PARTIAL SUCCESS – No Extended Access, PARTIAL SUCCESS – No Immediate Access, PARTIAL SUCCESS – No Extended or Immediate Access, NO SYNC, INVALID_PARAMETERS, UNSPECIFIED FAILURE	Reports the outcome of an MLMEX-SCHSTART.request

7.3.4.3.3 When generated

The primitive is generated by the MLME to report the result of an MLMEX-SCHSTART.request.

7.3.4.3.4 Effect of receipt

The WME may take action on receipt.

7.3.4.4 MLMEX-SCHEND.request

7.3.4.4.1 Function

The primitive is generated to instruct the MLME to cease providing access to the indicated SCH.

7.3.4.4.2 Semantics of the service primitive

The parameters of the primitive are as follows.

MLMEX-SCHEND.request (
Channel Identifier
)

Table 16 specifies the MLMEX-DELETEXPROFILE.request parameter.

Table 16—MLMEX-DELETETXPROFILE.request parameter

Name	Type	Valid range	Description
Channel Identifier	See 7.2	See 7.2	RF channel number for which access is no longer needed.

7.3.4.4.3 When generated

The primitive is generated as needed (see IEEE Std 1609.3).

7.3.4.4.4 Effect of receipt

Upon receipt of the MLMEX-SCHEND.request primitive, the MLME ceases providing access to the indicated service channel as specified in 6.3.5.

7.3.4.5 MLMEX-SCHEND.confirm

7.3.4.5.1 Function

The primitive is used to confirm the outcome of an MLMEX-SCHEND.request.

7.3.4.5.2 Semantics of the service primitive

The parameters of the primitive are as in Table 17:

```
MLMEX-SCHEND.confirm (
                ResultCode
                )
```

Table 17—MLMEX-SCHEND.confirm parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, INVALID PARAMETERS	Reports the result of an MLMEX-SCHEND.request.

7.3.4.5.3 When generated

The outcome of an MLMEX-SCHEND.request is confirmed.

7.3.4.5.4 Effect of receipt

The recipient may take action on receipt.

7.3.4.6 MLMEX-SCHEND.indication

7.3.4.6.1 Function

The primitive is used to inform a higher layer that a service channel has been released by the MLME.

7.3.4.6.2 Semantics of the service primitive

The parameters of the primitive are as shown in Table 18:

MLMEX-SCHEND.indication (
Reason
)

Table 18—MLMEX-SCHEND.indication parameter

Name	Type	Valid range	Description
Reason	Enumeration	LOSS OF SYNC, UNSPECIFIED REASON	Reports the reason the MLME released the SCH.

7.3.4.6.3 When generated

The primitive is generated when the MLME unilaterally releases a service channel (e.g., due to loss of synchronization).

7.3.4.6.4 Effect of receipt

The recipient may take action on receipt.

7.3.5 Transmitter profile

7.3.5.1 General

This mechanism supports the registration and deletion of a transmitter profile for IP-based communications.

7.3.5.2 MLMEX-REGISTERTXPROFILE.request

7.3.5.2.1 Function

The primitive is generated to register a transmitter profile in the MLME.

7.3.5.2.2 Semantics of the service primitive

The parameters of the primitive are as follows and in Table 19:

MLMEX-REGISTERTXPROFILE.request (
Channel Identifier,
Adaptable,
TxPwr_Level,
DataRate
)

Table 19 —MLMEX-REGISTERTXPROFILE.request parameters

Name	Type	Valid range	Description
Channel Identifier	See 7.2	See 7.2	RF channel number for the SCH.
Adaptable	Boolean	TRUE or FALSE	If TRUE, the actual power level and data rate for transmission are adaptable. *TxPwr_Level* is the maximum transmit power that sets the upper bound for the actual transmit power; *DataRate* is the minimum data rate that sets the lower bound for the actual data rate. If FALSE, the actual power level and data rate for transmission are nonadaptable. *TxPwr_Level* and *DataRate* are the actual values to be used for transmission.
TxPwr_Level	Integer	1–8	Index of the transmit power level as defined in the PHY MIB attribute *dot11PhyTxPowerTable* (IEEE Std 802.11).
DataRate	Integer	1–8	Index of the data rate as defined in the PHY MIB (IEEE Std 802.11).

7.3.5.2.3 When generated

The primitive is generated to register a transmitter profile in the MLME before the associated IP-based data transfer starts.

7.3.5.2.4 Effect of receipt

Upon receipt of the primitive, the MLME stores the values associated with the transmitter profile in its MIB attribute *TransmitterProfileTable* and responds with an MLMEX-REGISTERTXPROFILE.confirm.

7.3.5.3 MLMEX-REGISTERTXPROFILE.confirm

7.3.5.3.1 Function

The primitive is used to confirm the outcome of an MLMEX-REGISTERTXPROFILE.request.

7.3.5.3.2 Semantics of the service primitive

The parameters of the primitive are as follows:

MLMEX-REGISTERTXPROFILE.confirm (
ResultCode
)

Table 20 specifies the MLMEX-REGISTERTXPROFILE.confirm parameter.

Table 20—MLMEX-REGISTERTXPROFILE.confirm parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, INVALID PARAMETERS	Reports the result of the transmitter profile registration request.

7.3.5.3.3 When generated

The primitive confirms the outcome of an MLMEX-REGISTERTXPROFILE.request

7.3.5.3.4 Effect of receipt

The receipt indicates the result of the MLMEX-REGISTERTXPROFILE.request primitive to the WME.

7.3.5.4 MLMEX-DELETETXPROFILE.request

7.3.5.4.1 Function

The primitive is generated by the WME to delete a transmitter profile in the MLME.

7.3.5.4.2 Semantics of the service primitive

The parameters of the primitive are as follows and in Table 21:

MLMEX-DELETETXPROFILE.request (
Channel Identifier
)

Table 21—MLMEX-DELETETXPROFILE.request parameter

Name	Type	Valid range	Description
Channel Identifier	See 7.2	See 7.2	RF channel number for an SCH.

7.3.5.4.3 When generated

The MLMEX-DELETETXPROFILE.request primitive is generated by the WME to delete a registered transmitter profile in the MLME after the associated IP-based data transfer is complete.

7.3.5.4.4 Effect of receipt

Upon receipt of the primitive, the MLME deletes the transmitter profile and generates an MLMEX-DELETETXPROFILE.confirm primitive.

7.3.5.5 MLMEX-DELETETXPROFILE.confirm

7.3.5.5.1 Function

The primitive is used to confirm the outcome of an MLMEX-DELETETXPROFILE.request.

7.3.5.5.2 Semantics of the service primitive

The parameters of the primitive are as follows and in Table 22:

MLMEX-DELETETXPROFILE.confirm (
ResultCode
)

Table 22—MLMEX-DELETETXPROFILE.confirm parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, INVALID PARAMETERS	Reports the result of the transmitter profile deletion request.

7.3.5.5.3 When generated

The outcome of an MLMEX-DELETETXPROFILE.request is confirmed.

7.3.5.5.4 Effect of receipt

The receipt indicates the result of the MLMEX-DELETETXPROFILE.request primitive to the WME.

7.3.6 Cancel transmission

7.3.6.1 General

This mechanism is used to cancel all transmissions with a particular access category per channel in the MAC. A PHY-TXEND.request (defined in IEEE Std 802.11) may be subsequently issued by the MAC to cancel transmissions in process by the PHY.

When multiple higher layer entities are utilizing the same ACI queue for a channel, flushing the ACI queue results in packet losses for each higher layer entity utilizing that ACI. Care should be taken in this situation to avoid canceling information exchange associated with multiple higher layer entities, and remedies should be left to the implementer.

7.3.6.2 MLMEX-CANCELTX.request

7.3.6.2.1 Function

The primitive requests that the MAC entity cancel all transmissions with a specific access category on a channel.

7.3.6.2.2 Semantics of the service primitive

The primitive provides the following parameters, defined in Table 23:

MLMEX-CANCELTX.request (
Channel Identifier,
ACI
)

Table 23—MLMEX-CANCELTX.request parameters

Name	Type	Valid range	Description
Channel Identifier	See 7.2	See 7.2	RF channel.
ACI	Integer	0–3	See 5.4.

7.3.6.2.3 When generated

The MLMEX-CANCELTX.request primitive is generated to cancel all transmissions with the specified access category on a specified channel if needed.

7.3.6.2.4 Effect of receipt

The MAC entity cancels all transmissions with the specified access category on the specified channel (the corresponding ACI queues are purged).

7.3.6.3 MLMEX-CANCELTX.confirm

7.3.6.3.1 Function

The primitive reports the result of a request to cancel all transmissions with an access category on a channel.

7.3.6.3.2 Semantics of the service primitive

The primitive provides the following parameters, defined in Table 24:

MLMEX-CANCELTX.confirm (
ResultCode
)

Table 24—MLMEX-CANCELTX.confirm parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, INVALID_PARAMETERS, UNSPECIFIED FAILURE	Reports the outcome of MLMEX-CANCELTX request.

7.3.6.3.3 When generated

The primitive is generated by the MLME to report the result of an MLMEX-CANCELTX.request.

7.3.6.3.4 Effect of receipt

The requestor is notified of the result of an MLMEX-CANCELTX request.

7.3.7 UTC time

7.3.7.1 General

This mechanism is used to request and set the current value of the estimate of UTC time that the MLME maintains.

7.3.7.2 MLMEX-GETUTCTIME.request

7.3.7.2.1 Function

The primitive requests that the MLME return the current value of its estimate of UTC time.

7.3.7.2.2 Semantics of the service primitive

The primitive provides the following parameters:

MLMEX-GETUTCTIME.request ( )

7.3.7.2.3 When generated

The primitive is generated to request the value of UTC time.

7.3.7.2.4 Effect of receipt

The MLME issues an MLMEX-GETUTCTIME.confirm.

7.3.7.3 MLMEX-GETUTCTIME.confirm

7.3.7.3.1 Function

The primitive reports the result of a request to get UTC time.

7.3.7.3.2 Semantics of the service primitive

The primitive provides the following parameters, defined in Table 25:

MLMEX-GETUTCTIME.confirm (
ResultCode,
Timestamp,
Time Value,
Time Error
)

Table 25—MLMEX-GETUTCTIME.confirm parameters

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, FAILURE	Reports the outcome of GETUTCTIME request.
Timestamp	Integer	0 to (2^{64} – 1)	Value of the TSF timer as defined in IEEE Std 802.11.
Time Value	Specified in IEEE Std 802.11p	Specified in IEEE Std 802.11p	An estimate of the relationship between the *Timestamp* and UTC. As specified in IEEE Std 802.11p for *Timing Capabilities* equal to 1.
Time Error	Specified in IEEE Std 802.11p	Specified in IEEE Std 802.11p	An estimate of the error of the *Time Value*. As specified in IEEE Std 802.11p for *Timing Capabilities* equal to 1.

7.3.7.3.3 When generated

The primitive is generated by the MLME to report the result of an MLMEX-GETUTCTIME.request.

7.3.7.3.4 Effect of receipt

The requestor is notified of the result of an MLMEX-GETUTCTIME.request, and if successful, has all the information necessary to calculate an estimate of UTC time at the instant the TSF timer was read, and its uncertainty. For information, the UTC time estimate is the sum of the *Timestamp* and the *Time Value*.

7.3.7.4 MLMEX-SETUTCTIME.request

7.3.7.4.1 Function

The primitive requests that the MLME set the current value of its estimate of UTC time.

7.3.7.4.2 Semantics of the service primitive

The primitive provides the following parameters, defined in Table 26:

MLMEX-SETUTCTIME.request (
Time Value,
Time Error
)

Table 26—MLMEX-SETUTCTIME.request parameters

Name	Type	Valid range	Description
Time Value	Specified in IEEE Std 802.11p	Specified in IEEE Std 802.11p	An estimate of the relationship between the unit's TSF timer and UTC. As specified in IEEE Std 802.11p for *Timing Capabilities* equal to 1.
Time Error	Specified in IEEE Std 802.11p	Specified in IEEE Std 802.11p	An estimate of the error of the *Time Value*. As specified in IEEE Std 802.11p for *Timing Capabilities* equal to 1.

7.3.7.4.3 When generated

The primitive is generated to set the value of UTC time.

7.3.7.4.4 Effect of receipt

The MLME sets its estimate of UTC time and issues an MLMEX-SETUTCTIME.confirm.

7.3.7.5 MLMEX-SETUTCTIME.confirm

7.3.7.5.1 Function

The primitive reports the result of a request to set UTC time.

7.3.7.5.2 Semantics of the service primitive

The primitive provides the following parameters, defined in Table 27:

MLMEX-SETUTCTIME.confirm (
ResultCode
)

Table 27—MLMEX-GETUTCTIME.confirm parameters

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, INVALID_PARAMETERS, UNSPECIFIED FAILURE	Reports the outcome of SETUTCTIME request.

7.3.7.5.3 When generated

The primitive is generated by the MLME to report the result of an MLMEX-SETUTCTIME.request.

7.3.7.5.4 Effect of receipt

The requestor is notified of the result of an MLMEX-SETUTCTIME.request.

7.3.8 Address change

7.3.8.1 MLMEX-AddressChange.request

7.3.8.1.1 Function

The primitive indicates that an immediate MAC-layer address change is required (e.g., to support device pseudonymity).

7.3.8.1.2 Semantics of the service primitive

The parameters of the primitive are as follows:

MLMEX -AddressChange.request (
MAC Address (optional)
)

Name	Type	Valid range	Description
MAC Address	MACAddress	Any valid MAC address	If present, it indicates a new MAC address to be adopted by the device.

7.3.8.1.3 When generated

The primitive is passed to the MLME when an address change is needed.

7.3.8.1.4 Effect of receipt

On receipt, the MLME changes its identifying address and responds with a confirmation.

7.3.8.2 MLMEX-AddressChange.confirm

7.3.8.2.1 Function

The primitive indicates the results of a MAC-layer address change.

7.3.8.2.2 Semantics of the service primitive

The parameters of the primitive are as follows:

MLMEX-AddressChange.confirm (
ResultCode
)

Name	Type	Valid range	Description
ResultCode	Enumerated	Accepted, Rejected	Indicates the result of the associated request.

7.3.8.2.3 When generated

The primitive is generated in response to an MLMEX-AddressChange.request.

7.3.8.2.4 Effect of receipt

No effect is specified.

7.3.9 IEEE 802.11 MLME primitive generation

7.3.9.1 MLMEX-SendPrimitive.request

7.3.9.1.1 Function

The primitive is generated to request the invocation of one of the MLME service request primitives specified in IEEE Std 802.11.

7.3.9.1.2 Semantics of the service primitive

This primitive provides the following parameters:

MLMEX-SendPrimitive.request (
Channel Identifier,
Channel Interval,
PrimitiveContents
)

Table 28 specifies the MLMEX-SendPrimitive.request parameters.

Table 28—MLMEX-SendPrimitive.request parameters

Name	Type	Valid range	Description
Channel Identifier	See 7.2	See 7.2	The channel (CCH or SCH) on which the resulting management frame is transmitted.
Channel Interval	Enumeration	CCH Interval, SCH Interval, both	The channel interval in which the transmissions are to occur.
PrimitiveContents	As described in the text that follows the table	As described in the text that follows the table	As described in the text that follows the table.

PrimitiveContents consists of the information needed to construct an MLME service request primitive specified in IEEE Std 802.11, including an identification of which primitive is to be generated.

7.3.9.1.3 When generated

The primitive is generated to trigger the sending of an IEEE 802.11 MLME service request primitive.

7.3.9.1.4 Effect of receipt

Upon receipt of the primitive, the MLME sends the indicated service request primitive (see 6.5).

7.3.9.2 MLMEX-SendPrimitive.confirm

7.3.9.2.1 Function

The primitive reports the result of an MLMEX-SendPrimitive.request.

7.3.9.2.2 Semantics of the service primitive

The primitive provides the following parameter.

MLMEX-SendPrimitive.confirm (
ResultCode
)

Table 29 specifies the MLMEX-SendPrimitive.confirm parameter.

Table 29—MLMEX-SendPrimitive.confirm parameter

Name	Type	Valid range	Description
ResultCode	Enumeration	SUCCESS, NO SYNC, FEATURE NOT SUPPORTED, CHANNEL UNAVAILABLE, INVALID_PARAMETERS, UNSPECIFIED FAILURE	Reports the result of an MLMEX-SendPrimitive .request.

7.3.9.2.3 When generated

The primitive is generated by the MLME in response to an MLMEX-SendPrimitive.request.

7.3.9.2.4 Effect of receipt

No behavior is specified.

7.4 WAVE MAC SAP

7.4.1 General

The WAVE MAC SAP is identical to the IEEE 802.11 MAC SAP with the exception of parameters added to the MA-UNITDATA.request. These parameters are included in the MA-UNITDATAX.request specified below and allow higher layers to control the transmission characteristics of WSMP data.

7.4.2 MA-UNITDATAX.request

7.4.2.1 Function

This primitive requests a transfer of an MSDU from a local LLC sublayer entity to a single peer LLC sublayer entity, or to multiple peer LLC sublayer entities in the case of multicast addresses. The MA-UNITDATA.request specified in IEEE Std 802.11 is extended here by the addition of *Channel Identifier*, *Data Rate*, *TxPwr_Level*, and *Expiry Time* which are used in the case of WAVE Short Message transmissions.

7.4.2.2 Semantics of the service primitive

The parameters of the primitive are as follows:

MA-UNITDATAX.request (
source address,
destination address,
routing information,
data,
priority,
service class,
Channel Identifier,
Data Rate,
TxPwr_Level,
ExpiryTime
)

Channel Identifier is specified in 7.2. *Channel Number* (a component of *Channel Identifier*), *Data Rate*, *TxPwr_Level*, and *ExpiryTime* are used for WSMP transmissions as specified in 5.3.4. Otherwise the MA-UNITDATAX.request is processed as the MA-UNITDATA.request specified in IEEE Std 802.11.

7.4.2.3 When generated

This primitive is generated by the LLC sublayer entity when a WSMP MSDU is to be transferred to a peer LLC sublayer entity or entities.

7.4.2.4 Effect of receipt

On receipt of this primitive, 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-UNITDATAX.confirm 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 appends all MAC specified fields, passes the properly formatted frame to the lower layers for transfer to a peer MAC sublayer entity or entities (see IEEE Std 802.11), and indicates this action to the LLC sublayer entity using an MA-UNITDATAX.confirm primitive with transmission status set to Successful.

7.4.3 MA-UNITDATAX.confirm

7.4.3.1 Function

This primitive has local significance and provides the LLC sublayer with status information for the corresponding preceding MA-UNITDATAX.request primitive.

7.4.3.2 Semantics of the service primitive

The parameters of the primitive and their meanings are identical to those of the IEEE Std 802.11 MA-UNITDATA.confirm:

MA-UNITDATAX.confirm (
source address,
destination address,
transmission status,
provided priority,
provided service class
)

7.4.3.3 When generated

The MA-UNITDATAX.confirm primitive is passed from the MAC sublayer entity to the LLC sublayer entity to indicate the status of the service provided for the corresponding MA-UNITDATAX.request primitive.

7.4.3.4 Effect of receipt

The effect of receipt of this primitive by the LLC sublayer is dependent upon the type of operation employed by the LLC sublayer entity.

Annex A

(informative)

Bibliography

[B1] *IEEE Standards Style Manual*. Piscataway, NJ: IEEE, 2009.[8]

[B2] ITU-R TF.460-6 (2002), Standard-Frequency and Time-Signal Emissions.[9]

[B3] U.S. National Intelligent Transportation Systems (ITS) Architecture, Version 6.1.[10]

[8] IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, Piscataway, NJ 08854, USA (http://standards.ieee.org/).
[9] ITU-T publications are available from the International Telecommunications Union, Place des Nations, CH-1211, Geneva 20, Switzerland/Suisse (http://www.itu.int/).
[10] Available at http://www.its.dot.gov/arch/.

Annex B

(informative)

Channel congestion phenomenon following a channel switch

B.1 Summary

WAVE devices that perform channel switching may experience a period of above-average channel congestion immediately following a channel switch, possibly resulting in an unexpectedly high collision rate. This phenomenon can occur as follows:

A WAVE device might queue an MPDU at a time when the intended channel is not available. For example, an MPDU intended to be sent on the CCH during the CCH interval might be placed in its queue during the SCH interval. In an even worse scenario, MAC-layer queues can build up in the device if there are multiple MPDUs to be sent. Subclause 5.2 states that an MPDU that is at the head-of-line of a queue at the end of a guard interval will undergo a random back-off before attempting to access the channel. This helps avoid, but does not prevent, collisions between MPDUs in different devices. In particular, if two such MPDUs in neighboring devices choose the same back-off length they are likely to collide. If a large number of neighboring devices have packets queued at the end of a guard interval, the probability of collisions can be much higher than normal; in the case where there are multiple MPDUs in the MAC-layer queue, the duration of the packet collision phenomenon is even longer. These collisions can lead to a serious degradation in application performance.

Although the IEEE 802.11 MAC is designed to minimize collisions during such periods of high load, it takes time to adapt (see the specification of Random backoff time and Distributed coordination function access procedure in IEEE Std 802.11), and it is not as effective for broadcast and multicast MPDUs as it is for unicast (see the specification of Broadcast and multicast MPDU transfer procedure in IEEE Std 802.11).

This annex discusses underlying causes of this problem as a benefit to protocol implementers (i.e., to help avoid possible technical pitfalls that might result in less efficient implementations). There is a variety of approaches to solving this problem. For example, if higher layers are exposed to channel interval timing information they can provide the packet to the MAC layer during the interval in which it is intended to be sent. The specific approach that a device uses to avoid start-of-interval contention is beyond the scope of this standard.

B.2 Causes

There are two primary protocol design characteristics that enable this problem:

— Synchronous channel switching

— IEEE 802.11 congestion control and error recovery

As specified in 5.2, channel switching is a synchronous operation performed by multi-channel capable WAVE devices that share the same medium. The synchronicity of the switching ensures that MPDUs queued for the same channel in such devices are likely to collide once the channel becomes available to them, even if there is only one queued packet per device. If there is a lack of coordination between WAVE devices and higher layer operations this congestion phenomenon is exacerbated (i.e., packet collisions caused by synchronized transmissions immediately following a channel switch). The IEEE 802.11 congestion control and error recovery protocol for unicast packets, specified in 9.2 of that standard, adapts to the channel conditions using an automatic retransmission mechanism with random back-off delay.

Unfortunately, there is no error recovery mechanism for broadcast or multicast MPDUs (see 9.2.7 of IEEE Std 802.11). This is primarily because the unicast adaptation mechanism relies on acknowledgments, which are impractical for MPDUs sent to multiple destinations. Therefore, broadcast and multicast packets are highly susceptible to collision losses during periods of high load. The lack of error recovery mechanisms means that collisions involving broadcast or multicast MPDUs are less transparent to the applications that use their payloads, and thus there is a greater likelihood that this MAC behavior will impact application performance.

The probability of collisions after a channel switch increases with the number of devices that are switching and contending for the same medium, especially if the MPDU generation rate is high enough to result in queuing. Note that the number of contending devices is a function of many variables, including data rate and transmission power.

Annex C

(informative)

Avoiding transmission at scheduled guard intervals

The maximum transmit time for a full-size MSDU (2312 octets) is approximately 6.5 ms. Transmission of an MSDU, by a switching device, beginning shortly before the end of a channel interval could prevent the transmission from completing before the transmitter begins switching during the guard interval. According to the procedures defined in 5.3, if transmission is terminated at the end of a channel interval, the packet is not received successfully. The following method can be used to avoid transmission termination:

a) Before delivering an MSDU to the PHY, the MAC issues a PLME-TXTIME.request and the PHY returns a PLME-TXTIME.confirm with the required transmit time.

b) If the required transmit time exceeds the remaining duration of the channel interval, the MSDU should be queued in the MAC sublayer until a return to the proper channel occurs.

Annex D

(informative)

MIB table

This clause provides a summary of the contents of the MIB specified in Annex E. The MIB has been verified to compile using the “smitools” package from the Institute of Operating Systems and Computer Networks at the Technical University of Braunschweig, Germany. Using these tools, the MIB compiled without generating warnings of severity 4 or lower.[11]

Table D.1—MIB contents

MIB Segment	MIB item	Table entry	Contents	Type
General	*Capabilities*	SEQUENCE of *dot4StationConfigEntry*	*MutiPhyOperation* *ImmediateAccessImplemented* *ExtendedAccessImplemented* *AlternatingAccessImplemented* *GpsTimeImplemented* *VsaFramesImplemented* *TaFramesImplemented* *DeviceReaddressingImplemented* *ExpiryTimeImplemented*	Capability
General	*Switching*		*CchInterval* *SchInterval* *SyncTolerance* *MaxChSwitchTime*	Capability
General	*ChannelSetTable*	SEQUENCE of *ChannelSetTableEntry* (first entry is CCH)	*ChannelSetTableIndex* *OperatingClass* *ChannelNumber* *ManagementAdaptablePowerAndRate* *ManagementDataRate* *ManagementPowerLevel*	Capability
Channel	*EDCACchTable*	SEQUENCE OF *EDCACchEntry*	*EDCACchTableIndex* *EDCACchTableCWmin* *EDCACchTableCWmax* *EDCACchTableAIFSN* *EDCACchTableTXOPLimit* *EDCACchTableMSDULifetime*	Status
Channel	*EDCASchTable*	SEQUENCE OF *EDCASchEntry*	*EDCASchTableIndex* *EDCASchTableChannelNumber* *EDCASchTableCWmin* *EDCASchTableCWmax* *EDCASchTableAIFSN* *EDCASchTableTXOPLimit* *EDCASchTableMSDULifetime*	Status
Channel	*TransmitterProfileTable*	SEQUENCE OF *TransmitterProfileEntry*	*TransmitterProfileIndex* *ServiceChannelOperatingClass* *ServiceChannelNumber* *AdaptablePowerAndRate* *DataRate* *PowerLevel*	Status
Timing	*TimingInformation*		*TimingCapabilities* *TimeValue* *TimeError*	Status

[11] These tools may be accessed online using the following URL: http://www.ibr.cs.tu-bs.de/bin/smitools.cgi.

Annex E

(normative)

ASN.1 encoding of the IEEE 1609.4 MIB

```
-- **********************************************************************
-- * IEEE P1609.4 Management Information Base
-- **********************************************************************
IEEE1609dot4-MIB DEFINITIONS ::= BEGIN
IMPORTS
MODULE-IDENTITY, OBJECT-TYPE, Integer32, Unsigned32 FROM SNMPv2-SMI
MacAddress, TruthValue, TimeStamp FROM SNMPv2-TC
MODULE-COMPLIANCE, OBJECT-GROUP FROM SNMPv2-CONF
ifIndex FROM RFC1213-MIB
Ipv6Address, Ipv6AddressPrefix FROM IPV6-TC;
-- **********************************************************************
-- * MODULE IDENTITY
-- **********************************************************************
ieee1609dot4v2mib MODULE-IDENTITY
LAST-UPDATED "201007270000Z"
ORGANIZATION "IEEE P1609"
CONTACT-INFO
"WG E-mail: stds-p1609@ieee.org
Chair: Tom Kurihara
Postal: 3800 N. Fairfax Drive, #207
Arlington, VA USA 22203-1759
Tel: +1 703-516-9650
Fax: +1 703-516-4688
E-mail: tkstds@mindspring.com
Editor: John moring
Tel: +1 760 633 1790
E-mail: john@moring.net"
DESCRIPTION
"The MIB module for IEEE P1609.4 full use entities.
iso(1) iso-identified-organization(3) ieee(111)
standards-association-numbered-series-standards(2) ieeeP1609(1609) dot4(4)
v2mib(2)"
REVISION "2010067270000Z"
DESCRIPTION
"Consistent with full use version 1609.4"
::= { ieeeP1609 4 2}
ieeeP1609 OBJECT IDENTIFIER ::=
{1 iso-identified-organization (3) ieee (111)
standards-association-numbered-series-standards (2) 1609}
-- **********************************************************************
-- This MIB includes channel related info and timing info.
-- General info:
-- - Switching parameters (CCHInterval, SCHInterval, SyncTolerance,
MaxChSwitchTime)
-- Channel related info:
-- - Channel Set (Operating Class, Channel Number)
-- - EDCA_CCH Table (default EDCA parameter set for CCH)
-- - EDCA_SCH Table (default EDCA parameter set for SCH)
```

```
--     -     Transmitter     Profile     Table     (ServiceChannelNumber,
AdpatablePowerAndRate,
-- PowerLevel, DataRate)
-- Timing related info:
-- -Timing information (capabilities, time value, time error)

dot4GeneralMib OBJECT IDENTIFIER ::= { ieee1609dot4v2mib 1 }
dot4ChannelMib OBJECT IDENTIFIER ::= { ieee1609dot4v2mib 2 }
dot4TimingMib OBJECT IDENTIFIER ::= { ieee1609dot4v2mib 3 }
dot4Conformance OBJECT IDENTIFIER ::= { ieee1609dot4v2mib 4 }

-- ******************************************************************
-- ******************************************************************
-- * General Segment
-- ******************************************************************
-- ******************************************************************

-- ******************************************************************
-- * Device Capabilities Information
-- ******************************************************************

dot4StationConfigTable OBJECT-TYPE
SYNTAX SEQUENCE OF Dot4StationConfigEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Station Configuration attributes pertinent to channel coordination.
In tabular form to allow for multiple instances on an agent."
::= { dot4GeneralMib 1 }

dot4StationConfigEntry OBJECT-TYPE
SYNTAX Dot4StationConfigEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "An entry in the dot4StationConfigTable. Each entry
            corresponds to one of multiple IEEE 802.11 interfaces
(if present)."
INDEX { dot4StationConfigIndex }
::= { dot4StationConfigTable 1 }

Dot4StationConfigEntry ::=
SEQUENCE {
            dot4StationConfigIndex INTEGER,
            dot4MutiPhyOperation INTEGER,
            dot4ImmediateAccessImplemented TruthValue,
            dot4ExtendedAccessImplemented TruthValue,
            dot4AlternatingAccessImplemented TruthValue,
            dot4GpsTimeImplemented TruthValue,
            dot4VsaFramesImplemented TruthValue,
            dot4TaFramesImplemented TruthValue,
            dot4DeviceReaddressingImplemented TruthValue,
            dot4ExpiryTimeImplemented TruthValue}

dot4StationConfigIndex OBJECT-TYPE
SYNTAX INTEGER (1..32)
MAX-ACCESS not-accessible
```

```
STATUS current
DESCRIPTION
"Index to the Station Config Table."
::= { dot4StationConfigEntry 1 }

dot4MutiPhyOperation OBJECT-TYPE
SYNTAX INTEGER {singlePhy(1), multiPhy(2)}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports one or more
            IEEE 802.11 PHYs."
::= { dot4StationConfigEntry 2 }

dot4ImmediateAccessImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports Immediate Access."
::= { dot4StationConfigEntry 3 }

dot4ExtendedAccessImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports Extended Access."
::= { dot4StationConfigEntry 4 }

dot4AlternatingAccessImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports Alternating Access."
::= { dot4StationConfigEntry 5 }

dot4GpsTimeImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device acquires timing from the GPS."
::= { dot4StationConfigEntry 6 }

dot4VsaFramesImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports sending and/or
            receiving Vendor Specific Action frames."
::= { dot4StationConfigEntry 7 }

dot4TaFramesImplemented OBJECT-TYPE
SYNTAX TruthValue
```

```
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports sending and/or
            receiving Timing Advertisement frames."
::= { dot4StationConfigEntry 8 }

dot4DeviceReaddressingImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports MAC readdressing for
            pseudonymity."
::= { dot4StationConfigEntry 9 }

dot4ExpiryTimeImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports purging WSMP data
            queued for transmission in accordance with Expiry Time."
::= { dot4StationConfigEntry 10 }

-- **********************************************************************
-- * End Device Capabilities Information
-- **********************************************************************
-- **********************************************************************
-- * Switching Information
-- **********************************************************************
dot4SwitchingInfo OBJECT IDENTIFIER ::= { dot4GeneralMib 2}

dot4CchInterval OBJECT-TYPE
SYNTAX INTEGER (0..255)
UNITS "millisecond"
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Control Channel interval"
DEFVAL { 50 }
::= { dot4SwitchingInfo 1 }

dot4SchInterval OBJECT-TYPE
SYNTAX INTEGER (0..255)
UNITS "millisecond"
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Service Channel interval"
DEFVAL { 50 }
::= { dot4SwitchingInfo 2 }

dot4SyncTolerance OBJECT-TYPE
SYNTAX INTEGER (0..255)
UNITS "millisecond"
MAX-ACCESS read-write
```

```
STATUS current
DESCRIPTION
            "sync tolerance"
DEFVAL { 2 }
::= { dot4SwitchingInfo 3 }

dot4MaxChSwitchTime OBJECT-TYPE
SYNTAX INTEGER (0..255)
UNITS "millisecond"
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "maximum channel switch time"
DEFVAL { 2 }
::= { dot4SwitchingInfo 4 }

-- **********************************************************************
-- * End Switching Information
-- **********************************************************************
-- **********************************************************************
-- **********************************************************************
-- * End General Segment
-- **********************************************************************
-- **********************************************************************


-- **********************************************************************
-- **********************************************************************
-- * Channel Segment
-- **********************************************************************
-- **********************************************************************

-- **********************************************************************
-- * Channel List
-- **********************************************************************
dot4ChannelList OBJECT IDENTIFIER ::= { dot4ChannelMib 1}

dot4ChannelSetTable OBJECT-TYPE
SYNTAX SEQUENCE OF Dot4ChannelSetTableEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "This table lists the set of channel numbers
            available for use. Each entry contains a value representing
            an operating class, followed by a channel number. The first
            entry indicates the Control Channel while the other
            entries indicate service channels."
::= { dot4ChannelList 1 }

dot4ChannelSetTableEntry OBJECT-TYPE
SYNTAX Dot4ChannelSetTableEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "An entry (conceptual row) in the Channel Set Table."
            INDEX {dot4ChannelSetTableIndex}
::= { dot4ChannelSetTable 1 }
```

```
Dot4ChannelSetTableEntry ::=
SEQUENCE {
            dot4ChannelSetTableIndex INTEGER,
            dot4OperatingClass INTEGER,
            dot4ChannelNumber INTEGER,
            dot4ManagementAdaptablePowerAndRate TruthValue,
            dot4ManagementDataRate INTEGER,
            dot4ManagementPowerLevel INTEGER
}

dot4ChannelSetTableIndex OBJECT-TYPE
SYNTAX INTEGER (1..32)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "Index to the Channel Set Table."
::= {dot4ChannelSetTableEntry 1 }

dot4OperatingClass OBJECT-TYPE
SYNTAX INTEGER
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Operating Class per 802.11"
::= {dot4ChannelSetTableEntry 2 }

dot4ChannelNumber OBJECT-TYPE
SYNTAX INTEGER (0..200)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Channel Number"
::= {dot4ChannelSetTableEntry 3 }

dot4ManagementAdaptablePowerAndRate OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Adaptable status of tx power and data rate for management
            frames on this channel. If True, data rate represents a
            minimum value and power represents a maximum levels. If False,
            they represent singular values."
::= { dot4ChannelSetTableEntry 4 }

dot4ManagementDataRate OBJECT-TYPE
SYNTAX INTEGER (2..127)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Transmission data rate for management frames on this channel
            represented by a count from X'02-X'7f, corresponding to data
            rates in increments of  500 Kbps to 63.5 Mbps. The default
            value of this attribute is 12."
DEFVAL { 12 }
::= { dot4ChannelSetTableEntry 5 }
```

```
dot4ManagementPowerLevel OBJECT-TYPE
SYNTAX INTEGER (1..8)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "One of 8 Tx Power levels reserved for management frame
            transmission on this channel as defined in the 802.11 MIB
            dot11PhyTxPowerTable. The default value of this attribute
            is 4 (representing an intermediate power level)."
DEFVAL { 4 }
::= { dot4ChannelSetTableEntry 6 }


-- *******************************************************************
-- * End of Channel List Information
-- *******************************************************************
-- *****************************************************************
-- * EDCA_CCH Config TABLE
-- *****************************************************************
dot4EDCACchTable OBJECT-TYPE
SYNTAX SEQUENCE OF Dot4EDCACchEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "Value for EDCA default parameter values used on the control
            channel. This table shall contain the four entries of the EDCA
            parameters corresponding to four possible ACs."
::= { dot4ChannelMib 2}

dot4EDCACchEntry OBJECT-TYPE
SYNTAX Dot4EDCACchEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "An Entry (conceptual row) in the EDCA_CCH Table."
INDEX {dot4EDCACchTableIndex}
::= { dot4EDCACchTable 1 }

Dot4EDCACchEntry ::=
SEQUENCE {
            dot4EDCACchTableIndex INTEGER,
            dot4EDCACchTableCWmin INTEGER,
            dot4EDCACchTableCWmax INTEGER,
            dot4EDCACchTableAIFSN INTEGER,
            dot4EDCACchTableTXOPLimit INTEGER,
            dot4EDCACchTableMSDULifetime INTEGER
}

dot4EDCACchTableIndex 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 EDCA Table. The value of this
            variable is equal to AC + 1."
```

```
::= {dot4EDCACchEntry 1 }

dot4EDCACchTableCWmin OBJECT-TYPE
SYNTAX INTEGER (0..255)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Default value per IEEE Std 802.11p."
::= {dot4EDCACchEntry 2 }

dot4EDCACchTableCWmax OBJECT-TYPE
SYNTAX INTEGER (0..65535)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Default value per IEEE Std 802.11p."
::= {dot4EDCACchEntry 3 }

dot4EDCACchTableAIFSN OBJECT-TYPE
SYNTAX INTEGER (1..15)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Default value per IEEE Std 802.11p."
::= {dot4EDCACchEntry 4 }

dot4EDCACchTableTXOPLimit OBJECT-TYPE
SYNTAX INTEGER (0..65535)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Default value per IEEE Std 802.11p."
::= {dot4EDCACchEntry 5 }

dot4EDCACchTableMSDULifetime OBJECT-TYPE
SYNTAX INTEGER (0..500)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "This attribute shall specify (in TUs) the maximum duration
            an MSDU, for a given AC, would be retained by the MAC before
            it is discarded. The default value for this parameter shall be
500."
::= {dot4EDCACchEntry 6 }
-- *******************************************************************
-- * End of EDCA_CCH Config TABLE
-- *******************************************************************
-- *******************************************************************
-- * EDCA_SCH Config TABLE
-- *******************************************************************
dot4EDCASchTable OBJECT-TYPE
SYNTAX SEQUENCE OF Dot4EDCASchEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "Value for EDCA default parameter values used on service
            channels. This table contains the four entries of the EDCA
```

```
            parameters corresponding to four possible ACs."
::= { dot4ChannelMib 3}

dot4EDCASchEntry OBJECT-TYPE
SYNTAX Dot4EDCASchEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "An Entry (conceptual row) in the EDCA_SCH Table."
            INDEX {dot4EDCASchTableIndex}
::= { dot4EDCASchTable 1 }

Dot4EDCASchEntry ::=
SEQUENCE {
            dot4EDCASchTableOperatingClass INTEGER,
            dot4EDCASchTableChannelNumber INTEGER,
            dot4EDCASchTableIndex INTEGER,
            dot4EDCASchTableCWmin INTEGER,
            dot4EDCASchTableCWmax INTEGER,
            dot4EDCASchTableAIFSN INTEGER,
            dot4EDCASchTableTXOPLimit INTEGER,
            dot4EDCASchTableMSDULifetime INTEGER
}

dot4EDCASchTableIndex 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 EDCA Table. The value of this variable
            is equal to AC + 1."
::= {dot4EDCASchEntry 1 }

dot4EDCASchTableOperatingClass OBJECT-TYPE
SYNTAX INTEGER
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Operating Class per 802.11"
::= {dot4EDCASchEntry 2 }

dot4EDCASchTableChannelNumber OBJECT-TYPE
SYNTAX INTEGER (0..200)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Channel Number"
::= {dot4EDCASchEntry 3 }

dot4EDCASchTableCWmin OBJECT-TYPE
SYNTAX INTEGER (0..255)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Value to be used on this channel; default value is per
            IEEE Std 802.11p."
```

```
::= {dot4EDCASchEntry 4 }

dot4EDCASchTableCWmax OBJECT-TYPE
SYNTAX INTEGER (0..65535)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Value to be used on this channel; default value is per
            IEEE Std 802.11p."
::= {dot4EDCASchEntry 5 }

dot4EDCASchTableAIFSN OBJECT-TYPE
SYNTAX INTEGER (1..15)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Value to be used on this channel; default value is per
            IEEE Std 802.11p."
::= {dot4EDCASchEntry 6 }

dot4EDCASchTableTXOPLimit OBJECT-TYPE
SYNTAX INTEGER (0..65535)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "Value to be used on this channel; default value is per
            IEEE Std 802.11p."
::= {dot4EDCASchEntry 7 }

dot4EDCASchTableMSDULifetime OBJECT-TYPE
SYNTAX INTEGER (0..500)
MAX-ACCESS read-write
STATUS current
DESCRIPTION
            "This attribute shall specify (in TUs) the maximum duration
            an MSDU, for a given AC, would be retained by the MAC before
            it is discarded. The default value for this parameter shall
            be 500."
::= {dot4EDCASchEntry 8 }

-- ***********************************************************************
-- * End of EDCA_SCH Config TABLE
-- ***********************************************************************
-- *************************************************************************
-- * Transmitter Profile Information
-- *************************************************************************
dot4TransmitterProfileTable OBJECT-TYPE
SYNTAX SEQUENCE OF Dot4TransmitterProfileEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "Transmitter Profile Table."
::= { dot4ChannelMib 4}

dot4TransmitterProfileEntry OBJECT-TYPE
SYNTAX Dot4TransmitterProfileEntry
MAX-ACCESS not-accessible
```

```
STATUS current
DESCRIPTION
            "Transmitter Profile Entry."
INDEX {dot4TransmitterProfileIndex}
::= { dot4TransmitterProfileTable 1}

Dot4TransmitterProfileEntry ::= SEQUENCE {
            dot4TransmitterProfileIndex INTEGER,
            dot4ServiceChannelOperatingClass INTEGER,
            dot4ServiceChannelNumber INTEGER,
            dot4AdaptablePowerAndRate TruthValue,
            dot4DataRate INTEGER,
            dot4PowerLevel INTEGER}

dot4TransmitterProfileIndex OBJECT-TYPE
SYNTAX INTEGER (0..127)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "Table index."
::= { dot4TransmitterProfileEntry 1 }

dot4ServiceChannelOperatingClass OBJECT-TYPE
SYNTAX INTEGER
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Service Channel number for IP-based applications"
::= { dot4TransmitterProfileEntry 2 }

dot4ServiceChannelNumber OBJECT-TYPE
SYNTAX INTEGER (0..200)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Service Channel number for IP-based applications"
::= { dot4TransmitterProfileEntry 3 }

dot4AdaptablePowerAndRate OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Adaptable status of tx power and data rate"
::= { dot4TransmitterProfileEntry 4 }

dot4DataRate OBJECT-TYPE
SYNTAX INTEGER (2..127)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Transmission data rate represented by a count from X'02-X'7f,
            corresponding to data rates in increments of 500 Kbps to
            63.5 Mbps. The default value of this attribute is 12."
DEFVAL { 12 }
::= { dot4TransmitterProfileEntry 5 }
```

```
dot4PowerLevel OBJECT-TYPE
SYNTAX INTEGER (1..8)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "One of 8 Tx Power levels reserved for WAVE PHY transmission
            as defined in the 802.11 MIB dot11PhyTxPowerTable. The default
            value of this attribute is 4 (representing an intermediate
            power level."
DEFVAL { 4 }
::= { dot4TransmitterProfileEntry 6 }

-- *********************************************************************
-- * End of Transmitter Profile Information
-- *********************************************************************

-- *********************************************************************
-- *********************************************************************
-- * End of Channel Segment
-- *********************************************************************
-- *********************************************************************

-- *********************************************************************
-- *********************************************************************
-- * Timing Segment
-- *********************************************************************
-- *********************************************************************

-- *********************************************************************
-- * WAVE Timing Information
-- *********************************************************************
dot4TimingInfo OBJECT IDENTIFIER ::= { dot4TimingMib 1}

dot4TimingCapabilities OBJECT-TYPE
SYNTAX INTEGER (0..7)
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Timing Capabilities per IEEE 802.11.
            0 indicates no standardized external time source
            1 indicates offset based on UTC
            2 and higher - see IEEE 802.11"
::= { dot4TimingInfo 1 }

dot4TimeValue OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(10))
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Time Value per IEEE 802.11"
::= { dot4TimingInfo 2 }

dot4TimeError OBJECT-TYPE
SYNTAX OCTET STRING (SIZE(5))
UNITS "nanosecond"
MAX-ACCESS read-only
STATUS current
```

```
DESCRIPTION
            "40-bit Timing Error per IEEE 802.11"
::= { dot4TimingInfo 3 }

-- ***********************************************************************
-- ***********************************************************************
-- * End of Timing Segment
-- ***********************************************************************
-- ***********************************************************************


-- *********************************************************************
-- * Conformance Segment
-- *********************************************************************
dot4Compliances OBJECT IDENTIFIER ::= { dot4Conformance 1 }
dot4Groups OBJECT IDENTIFIER ::= { dot4Conformance 2 }

dot4Compliance MODULE-COMPLIANCE
            STATUS current
            DESCRIPTION
            "Describes the requirements for conformance to the IEEE
            1609.4 MIB."
            MODULE -- this module
            MANDATORY-GROUPS {
            dot4CapabilitiesGroup,
            dot4ChannelSetGroup
            }

GROUP dot4EDCACchGroup
DESCRIPTION
"This group is mandatory for WAVE devices capable of operation on the
control channel."

GROUP dot4EDCASchGroup
DESCRIPTION
"This group is mandatory for WAVE devices capable of operation on
service channels."

GROUP dot4TransmitterProfileGroup
DESCRIPTION
"This group is mandatory for WAVE devices that support IPv6."

GROUP dot4TimingGroup
DESCRIPTION
"This group is mandatory for WAVE devices that support channel switching."

::= { dot4Compliances 1 }

-- *********************************************************************
-- * Conformance Groups
-- *********************************************************************
dot4CapabilitiesGroup OBJECT-GROUP
            OBJECTS {
            dot4MutiPhyOperation,
            dot4ImmediateAccessImplemented,
            dot4ExtendedAccessImplemented,
            dot4AlternatingAccessImplemented,
```

```
            dot4GpsTimeImplemented,
            dot4VsaFramesImplemented,
            dot4TaFramesImplemented,
            dot4DeviceReaddressingImplemented,
            dot4ExpiryTimeImplemented
            }
            STATUS current
            DESCRIPTION
            "Capabilities objects"
            ::= { dot4Groups 1 }

dot4ChannelSetGroup OBJECT-GROUP
            OBJECTS {
            dot4OperatingClass,
            dot4ChannelNumber,
            dot4ManagementAdaptablePowerAndRate,
            dot4ManagementDataRate,
            dot4ManagementPowerLevel
            }
            STATUS current
            DESCRIPTION
            "Channel set objects"
            ::= { dot4Groups 2 }

dot4EDCACchGroup OBJECT-GROUP
            OBJECTS {
            dot4EDCACchTableCWmin,
            dot4EDCACchTableCWmax,
            dot4EDCACchTableAIFSN,
            dot4EDCACchTableTXOPLimit,
            dot4EDCACchTableMSDULifetime
            }
            STATUS current
            DESCRIPTION
            "EDCA for the CCH objects"
            ::= { dot4Groups 3 }

dot4EDCASchGroup OBJECT-GROUP
            OBJECTS {
            dot4EDCASchTableOperatingClass,
            dot4EDCASchTableChannelNumber,
            dot4EDCASchTableCWmin,
            dot4EDCASchTableCWmax,
            dot4EDCASchTableAIFSN,
            dot4EDCASchTableTXOPLimit,
            dot4EDCASchTableMSDULifetime
            }
            STATUS current
            DESCRIPTION
            "EDCA for the CCH objects"
            ::= { dot4Groups 4 }

dot4TransmitterProfileGroup OBJECT-GROUP
            OBJECTS {
            dot4ServiceChannelOperatingClass,
            dot4ServiceChannelNumber,
            dot4AdaptablePowerAndRate,
```

```
            dot4DataRate,
            dot4PowerLevel
            }
            STATUS current
            DESCRIPTION
            "Capabilities objects"
            ::= { dot4Groups 5 }

dot4TimingGroup OBJECT-GROUP
            OBJECTS {
            dot4CchInterval,
            dot4SchInterval,
            dot4SyncTolerance,
            dot4MaxChSwitchTime,
            dot4TimingCapabilities,
            dot4TimeValue,
            dot4TimeError
            }
            STATUS current
            DESCRIPTION
            "Timing objects"
            ::= { dot4Groups 6 }

-- *********************************************************************
-- * End of 1609.4 MIB
-- *********************************************************************
END
```

Annex F

(informative)

Precise timing sources and timing quality estimation

When synchronizing to an external source of absolute time, this standard specifies that UTC will be used. UTC is commonly derived from GPS. This informative annex assumes GPS is the source of precise UTC and gives an example of how GPS can be used in conjunction with internal clocks and a Kalman filter to estimate UTC, as well as an estimate of the standard deviation of the error of the internal estimate of UTC (timing quality). The following is intended as an illustrative example. Details of how these steps are accomplished are implementation dependent, and other implementations are possible.

The following are elements of UTC time estimation in a WAVE device:

a) GPS is a universal standard available for generating UTC and is assumed “correct” by definition.

b) GPS satellites use crystal oscillators stabilized by rubidium standards. These are in turn corrected by a ground time standard.

c) Standard GPS units provide a time pulse (e.g., every 1 s). The pulse timing has some small errors, depending on atmospherics, but by definition, it does not drift.

d) GPS timing accuracy is also dependent on accuracy of position knowledge and upon direct ray communication with the satellite(s). As such, for mobile units whose RF environment can change, there may be periods where GPS is unavailable, so any design relying on GPS time for synchronization should account for this possibility.

e) For maintaining time between timing pulses, a low (phase) noise, oven stabilized crystal oscillator (OCXO) is often used in situations where very precise subsecond time is required. For WAVE, the use of such OCXOs is not required nor anticipated, since very precise subsecond timing is not required.

f) A crystal oscillator in the WAVE receiver may be stable (noise wise), but these oscillators do drift.

g) GPS may be used to stabilize this crystal oscillator, usually using a version of a continuous, discrete Kalman filter.

h) The continuous-discrete Kalman filter typically employs a dynamic model of the oscillator output whose states are the clock offset from UTC (error), its rate of change (drift), and possibly the rate of change of drift. Using the model, current estimates of the drift parameters, and some statistical assumptions of the uncertainty in the dynamic model, it predicts UTC and the variance of that estimate at the time at which the next pulse from the GPS unit arrives. Upon receipt of a pulse, the Kalman filter compares the oscillator’s count between time pulses with the “proper” count (e.g., if a 1 MHz oscillator were used, the proper count would be 10^6) and computes a filtered (best) estimate of UTC offset along with estimates of the drift parameters. In addition, it computes the filtered estimate error covariance. The standard deviation (square root of the variance) of the error in the estimate of UTC time is the 1 sigma width of the probability distribution of the error associated with the time base generation process, and it is a time-varying quantity that increases monotonically between pulse measurements.

i) The previous description assumed that the timing pulse was available from some internal source such as a GPS unit. While GPS units are commonly available and can be used for this purpose in addition to accurate location estimation, they are not required to be used. Other sources of UTC are possible, including the use of UTC time present in management frames that are transmitted over the air. This particular method is a bit more complex than described, but does not require an onboard GPS unit.

j) For reference, a typical error budget for a fixed (nonmobile) GPS time reference, when lock on four or more satellites is achieved, is as follows:

Inherent GPS measurement error	(very small)	(ephemeris error, atmospherics, time dilution of precision [TDOP], etc.)
GPS antenna position error	5 m ~5 ns	(reasonable position survey ~5 ft)
Cable error	~1 ns	(reasonable knowledge of cable length and dielectric constant)
GPS time/frequency tracking error	~5 ns	(loop stress—typical value)
Knowledge of tracking error	~2 ns	(sigma unknown—typical value)
RSS of errors	~7 ns to 8 ns	

Such a system typically takes several hours to settle from a cold start (most GPS receivers do not require a cold start unless the antenna has been moved significantly since the last acquisition of GPS, or are being started for the first time). These timing budget figures go up dramatically (by a factor of 5 or more) for mobile timing reference, as position and timing knowledge is significantly degraded even with differential correction, and the filter loop bandwidths have to be wider to accommodate variations caused by mobility. Since the accuracies required for WAVE operations allow errors several orders of magnitude larger than these, GPS internal errors are typically not an issue.

Annex G

(normative)

Protocol Implementation Conformance Statement (PICS) proforma[12]

Notes:

— Entries in the Item column may be hierarchical. Thus, an entry of the form M<a>.<b> indicates the item is part of the group identified by the item M<a> where all members of the group are subject to the conditions of applicability of M<a> [i.e., features lower in the numbering hierarchy (M<a>.<b>) are only applicable if the next higher level feature (M <a>) is identified in the Conformance column as being present].

— Parentheses in the Value column indicate the user should enter information as specified in the accompanying footnote.

— An entry of the form <pred>:<S> in the Status column indicates that the status <S> applies if the item identified by <pred> is identified in the Conformance column as being present.

— Valid status values in the Status column are M, O, O<n>, and C<n>.

  — A status of M indicates a mandatory feature.

  — A status of O indicates an optional feature.

  — A status of O<n> indicates a mutual conditionality such that the feature is optional but that support of at least one of the items that have status O<n> is mandatory.

  — A status of C<n> indicates a mutual conditionality such that support of one and one only of the items that have status C<n> is mandatory.

[12] *Copyright release for PICS proforma:* Users of this standard may freely reproduce the PICS proforma in this annex so that it can be used for its intended purpose and may further publish the completed PICS.

Item	Feature	Value	Reference	Status	Support
	General				
M1.	**Single-PHY device**		5.1, 6.3.1	C1	
M2.	**Multi-PHY device**	( )[a]	6.3.1	C1	
M3.	Operation on CCH	( )[b]	5.2	O4	
M4.	Operation on SCH	( )[c]	5.2	O4	
M5.	**Channel access**		6.3	O	
M5.1.	Continuous CCH access		6.3.3	O	
M5.2.	Continuous SCH access		6.3.3	O	
M5.3.	Immediate SCH access		6.3.3	O	
M5.4.	Extended SCH access		6.3.4	O	
M5.5.	Alternating SCH and CCH access		6.3.2	O	
M5.5.1.	Use UTC time estimate		5.2.2, 6.2.1	M	
M5.5.1.1.	Derive timing from GPS		6.2.3	O5	
M5.5.1.2.	Derive timing from Timing Advertisement frame		6.2.3	O5	
M5.5.1.3.	Derive timing from other timing source	( )[d]	6.2.3	O5	
M5.5.2.	Guard interval on transmit		6.2.5	M	
M5.5.3.	Medium busy declaration at end of guard interval		6.2.5	M	
M6.	**Transmit**		5.3.2	O2	
M6.1.	EDCA and user priority		5.4	M	
M6.2.	Cancel transmissions		7.3.6	O	
M6.3.	Send VSA		6.4.1	O	
M6.3.1.	Send VSA with non-1609 *Organization Identifier*		6.4.1	O	
M6.4.	Send TA		6.2.6	O	
M6.5.	Send other IEEE 802.11 frames	( )[e]	6.5	O	
M6.6.	Send WSM		5.3.4	O3	
M6.6.1.	Expiry time		5.3.4	O	
M6.7.	Send IPv6		5.3.5	O3	
M7.	**Receive**		5.3.3	O2	
M7.1.	Receive VSA		6.4.2, 5.3.6	O	
M7.2.	Receive TA		6.2.7	O	
M7.3.	Receive WSM		5.3.4	O3	
M7.4.	Receive IPv6		5.3.5	O3	
M8.	**Device readdressing**		6.7	O	
M9.	**MIB maintenance**		6.6	—	
M9.1.	Managed WAVE device		0, 6.6	O	
M9.2.	Per Annex E		6.6	M9.1: M	
M9.3.	Other MIB	( )[f]	6.6	O	

[a]Enter number of simultaneous channels supported.
[b]List supported control channel(s), including country and operating class.
[c]List supported service channel(s), including country and operating class.
[d]Indicate device's timing source(s).
[e]Enter IEEE 802.11 management frames/service request primitives supported.
[f]Enter references to other management information bases supported.

Annex H

(normative)

System characteristic values

For interoperability, communicating WAVE devices need to operate with common system characteristics. The following set of default system parameters shall be employed. In the future, to accommodate different WAVE systems (e.g., in different regulatory domains), different sets of characteristics may be specified in this standard.

Table H.1—System characteristics

Characteristic	**Value**
OperatingClass (CCH)	17
ChannelNumber (CCH)	178
CchInterval	50 ms
SchInterval	50 ms
SyncTolerance	2 ms
MaxChSwitchTime	2 ms