IEEE STANDARDS ASSOCIATION

IEEE Standard for Wireless Access in Vehicular Environments (WAVE)—Networking Services

IEEE Vehicular Technology Society

Sponsored by the
Intelligent Transportation Systems Committee

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

30 December 2010

IEEE Std 1609.3™-2010
(Revision of
IEEE Std 1609.3-2007)

IEEE Std 1609.3™-2010
(Revision of
IEEE Std 1609.3-2007)

IEEE Standard for Wireless Access in Vehicular Environments (WAVE)—Networking Services

Sponsor

Intelligent Transportation Systems Committee

of the

IEEE Vehicular Technology Society

Approved 30 September 2010

IEEE-SA Standards Board

Abstract: Wireless Access in Vehicular Environments (WAVE) Networking Services provides services to WAVE devices and systems. Layers 3 and 4 of the open system interconnect (OSI) model and the Internet Protocol (IP), User Datagram Protocol (UDP), and Transmission Control Protocol (TCP) elements of the Internet model are represented. Management and data services within WAVE devices are provided.

Keywords: Provider Service Identifier (PSID), Wireless Access in Vehicular Environments (WAVE), WAVE Short Message (WSM)

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

PDF: ISBN 978-0-7381-6483-0 STD97030
Print: ISBN 978-0-7381-6484-7 STDPD97030

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 1609.3-2010, IEEE Standard for Wireless Access in Vehicular Environments (WAVE)—Networking Services.

This document replaces IEEE Std 1609.3™-2007.

Regulatory bodies can designate particular (radio frequency) channels in Intelligent Transportation Systems allocated spectrum for specific dedicated purposes (e.g., vehicle-to-vehicle safety applications or emergency vehicle preemption). The channel access assignment mechanisms specified in this standard can be configured to satisfy such regulatory requirements.

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

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.

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 individual 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
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
Dale Sumida
Jasja Tijink
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	Young Kyun Kim	Ronald C. Petersen
Victor Berman	Joseph L. Koepfinger*	Thomas Prevost
Ted Burse	John Kulick	Jon Walter Rosdahl
Clint Chaplin	David J. Law	Sam Sciacca
Andy Drozd	Hung Ling	Mike Seavey
Alexander Gelman	Oleg Logvinov	Curtis Siller
Jim Hughes	Ted Olsen	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)—Networking Services

1. Overview

1.1 General

A WAVE system is a radio communications system intended to provide interoperable services to transportation. These services include those recognized by the U.S. National Intelligent Transportation Systems (ITS) Architecture and many others contemplated by the automotive and transportation infrastructure industries. These services include communications between vehicles and roadside units, vehicles and other vehicles, and perhaps communications among other WAVE devices. WAVE Networking Services provide management services and data delivery services between WAVE devices. This is but one component in the overall WAVE architecture, which includes IEEE P1609.1™ [B1],[1] IEEE P1609.2™,[2] IEEE Std 1609.4™, IEEE Std 802.11™, and IEEE Std 802.11p™. The WAVE architecture is described in IEEE P1609.0™.

1.2 Scope

The scope of this standard is the specification of network and transport layer protocols and services that support multi-channel wireless connectivity between IEEE 802.11 Wireless Access in Vehicular Environments (WAVE) devices.

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

1.3 Purpose

WAVE Networking Services represent layers 3 and 4 of the open system interconnect (OSI) communications stack. The purpose of this standard is to provide addressing and data delivery services within a WAVE system, providing multiple higher layer entities access to WAVE communication services. Upper layer support includes in-vehicle applications offering safety and convenience to their users.

1.4 Conformance

Per the *IEEE Style Manual* [B2], this standard includes normative and informative information. Normative text may describe mandatory or optional features. A mandatory feature may have optional as well as 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 C summarizes the features and their components.

1.5 Document organization

Clause 1 provides an overview of the document. Clause 2 and Clause 3 contain references, definitions, and abbreviations, respectively. Clause 4 provides a general description of WAVE Networking Services. Clause 5 specifies the data plane elements of WAVE Networking Services, which carry higher layer data through the system. Clause 6 specifies the management plane functions that support system operations. Clause 7 defines the primitives used to communicate between WAVE Networking Services and other internal functional entities. Clause 8 specifies the format of WAVE Service Advertisements and Short Messages. Annex A and Annex B contain a description, and formal definition, of the management information employed by WAVE Networking Services. Annex C provides a PICS proforma. Annex D provides some examples of Networking Services usage. Annex E contains allocated *WAVE Element ID* values. Annex F specifies a supplement to the WAVE Short Message Protocol for safety applications. Annex G shows the contents of example WAVE packets. Annex H shows example Provider Service Identifiers. Annex I provides an informative bibliography and definitions.

1.6 Document conventions

Unless otherwise noted, conventions follow those in IEEE Std 802.11, including conventions for the ordering of information within data items as specified in Clause 8 of this standard.

Numbers are decimal unless otherwise noted, except Internet Protocol (IP) addresses, which are hexadecimal per the conventions defined in IETF RFC 2373. Numbers preceded by 0x indicate hexadecimal numbers, so that 0xFF is equivalent to “FF hexadecimal.” Numbers preceded by 0b are binary.

Words in italics refer to data items that are defined as either a field in an interface primitive, a field in an over the air message, or a data item in the management information base (MIB).

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. Refer to the *IEEE Style Manual* [B2] for IEEE standards conventions.

Unless otherwise noted, references to a MIB refer to the WME MIB specified in Annex B.

1.7 System overview

The system described herein supports high-rate, low-latency communications between WAVE devices. Generic Internet Protocol version 6 (IPv6) traffic is supported (not Internet Protocol version 4), as well as a specialized WAVE Short Message service, although a given device might not support both protocols. A control channel provides a common channel for signaling. IP data is restricted to service channels; short message data may be sent on either type of channel. Higher layer entities benefiting from the WAVE communications can reside on the WAVE devices or reside on devices located on other networks connected to these devices. A more complete description of a WAVE system is provided in IEEE P1609.0.

1.8 Applicability

WAVE Networking Services support wireless communications between any and all WAVE devices. These devices may be mobile, portable, or stationary. The mobile devices include vehicles operating at high speeds on open highways. A common characteristic of WAVE systems is the need for extremely low communications latency (measured in milliseconds) from initially encountering a device that provides services to completing a set of data transfers.

This standard is consistent with the vehicle-to-roadside and vehicle-to-vehicle communications needs of various ITS architectures. Its intent is to enable interoperability and robust safety/public safety communications among these WAVE devices.

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.[1]

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

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

[1] 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.[2, 3, 4]

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 1609.4™, IEEE Standard for Wireless Access in Vehicular Environments (WAVE)—Multi-Channel Operation.

IETF RFC 768, User Datagram Protocol.[5]

IETF RFC 793, Transmission Control Protocol.

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

IETF RFC 2460, Internet Protocol, Version 6 (IPv6) Specification.

IETF RFC 3513, Internet Protocol Version 6 (IPv6) Addressing Architecture.

IETF RFC 4861, Neighbor Discovery for IP Version 6 (IPv6).

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.

SAE J2735, Dedicated Short Range Communications (DSRC) Message Set Dictionary.[6]

[2] IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, Piscataway, NJ 08854, USA (http://standards.ieee.org/).
[3] The IEEE standards or products referred to in this clause are trademarks owned by the Institute of Electrical and Electronics Engineers, Incorporated.
[4] IEEE and 802 are registered trademarks in the U.S. Patent & Trademark Office, owned by The Institute of Electrical and Electronics Engineers, Incorporated.
[5] 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).
[6] SAE publications are available from the Society of Automotive Engineers, 400 Commonwealth Drive, Warrendale PA 15096, USA (http://www.sae.org/).

3. Definitions, acronyms, and abbreviations

3.1 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]

air interface: A radio frequency communication interface, from the physical layer of one device to that of a remote device, such as the interface defined by Wireless Access in Vehicular Environments (WAVE).

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

data plane: A set of communication protocols defined to carry higher layer and management data. The data plane consists of protocol stack(s) for the transfer of data through a device for transmission over the air.

Ethertype: The Ethernet Type field defined in IETF RFC 1042, used to identify the protocol above the Logical Link Control.

hexadecimal representation: (From IEEE Std 802®) The representation of a sequence of octet values in which the values of the individual octets are displayed in order from left to right, with each octet value represented as a two-digit hexadecimal numeral, and with the resulting pairs of hexadecimal digits separated by hyphens. The order of the hexadecimal digits in each pair, and the mapping between the hexadecimal digits and the bits of the octet value, are derived by interpreting the bits of the octet value as a binary numeral using the normal mathematical rules for digit significance.

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.

management plane: A collection of functions performed in support of the communication functions provided by the data plane but not directly involved in passing higher layer data. The management plane may employ lower layers of the data plane to transfer management information between devices.

Networking Services: A collection of management plane and data plane functions at the network layer and transport layer, as specified in this standard, supporting Wireless Access in Vehicular Environments (WAVE) communications.

notification: An indication of an event of interest, sent from a management entity to a higher layer entity.

[7] *The IEEE Standards Dictionary: Glossary of Terms & Definitions* is available at http://shop.ieee.org/.

provider: Advertiser of a Wireless Access in Vehicular Environments (WAVE) service. *See also:* **user**; **Wireless Access in Vehicular Environments (WAVE) Service Advertisement**.

Provider Service Context (PSC): A field associated with a Provider Service Identifier (PSID) containing supplementary information related to the service. The internal format of the PSC is PSID dependent.

Provider Service Identifier (PSID): An octet string that identifies a service provided by a higher layer entity.

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 priority: A priority level associated with a service request that determines which service requests are given precedence by the WAVE Management Entity (WME) when assigning channels.

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

user: One that acts on receipt of a WAVE Service Advertisement. *See also:* **provider; Wireless Access in Vehicular Environments (WAVE) Service Advertisement**.

user priority: A priority level assigned to a packet that is ready for transmission, which determines its treatment at the medium access control (MAC) layer.

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 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 P1609.3, IEEE Std 1609.4, and IEEE Std 802.11, operating outside the context of a basic service set. (See IEEE P1609.0 and IEEE Std 802.11p.)

Wireless Access in Vehicular Environments (WAVE) medium access control (MAC): A WAVE MAC is a medium access control sublayer conformant to IEEE Std 1609.4.

Wireless Access in Vehicular Environments (WAVE) Management Entity (WME): A set of management functions required to provide WAVE networking services.

Wireless Access in Vehicular Environments (WAVE) Routing Advertisement (WRA): Network configuration information broadcast as part of a WAVE Service Advertisement.

Wireless Access in Vehicular Environments (WAVE) Service Advertisement (WSA): A data structure containing information including the announcement of the availability of a service.

Wireless Access in Vehicular Environments (WAVE) Short Message (WSM): A packet consisting of WSM data and a WAVE Short Message Protocol (WSMP) header.

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.

3.2 Acronyms and abbreviations

ACI	access category index
ACM	admission control mandatory
AIFSN	arbitration interframe space number
ASCII	American Standard Code for Information Interchange
ASN	Abstract Syntax Notation
CCH	control channel
CW	contention window
DNS	domain name system
EDCA	enhanced distributed channel access
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPv6	Internet Protocol version 6
ITS	Intelligent Transportation Systems
LLC	Logical Link Control
LSAP	link service access point
MAC	medium access control
MIB	management information base
MLME	MAC sublayer management entity
OSI	open system interconnect
PHY	physical layer
PICS	protocol implementation conformance statement
PSC	Provider Service Context
PSID	Provider Service Identifier
RCPI	Received Channel Power Indicator
RF	radio frequency
RFC	Request for Comments
SAE	Society of Automotive Engineers
SAP	service access point
SCH	service channel
SNAP	Subnetwork Access Protocol
STA	station
TA	Timing Advertisement (frame)
TCP	Transmission Control Protocol
TXOP	transmission opportunity
UDP	User Datagram Protocol
VSA	Vendor Specific Action (frame)
WAVE	Wireless Access in Vehicular Environments
WGS	world geodetic system
WME	WAVE Management Entity
WRA	WAVE Routing Advertisement
WSA	WAVE Service Advertisement
WSM	WAVE Short Message
WSMP	WAVE Short Message Protocol
WSMP-I	WAVE identity supplement
WSMP-S	WSMP safety supplement

4. General description

4.1 Overview

WAVE provides a communication protocol stack optimized for the vehicular environment, employing both customized and general-purpose elements as shown in Figure 1. IEEE P1609.0 provides a description of the WAVE system architecture and operations.

WAVE Networking Services is specified in this standard and consists of data plane layers and the associated management plane entity (WAVE management entity, WME) as shown by the dashed lines in Figure 2. These are introduced in 4.2 and 4.3 and specified in Clause 5 and Clause 6.

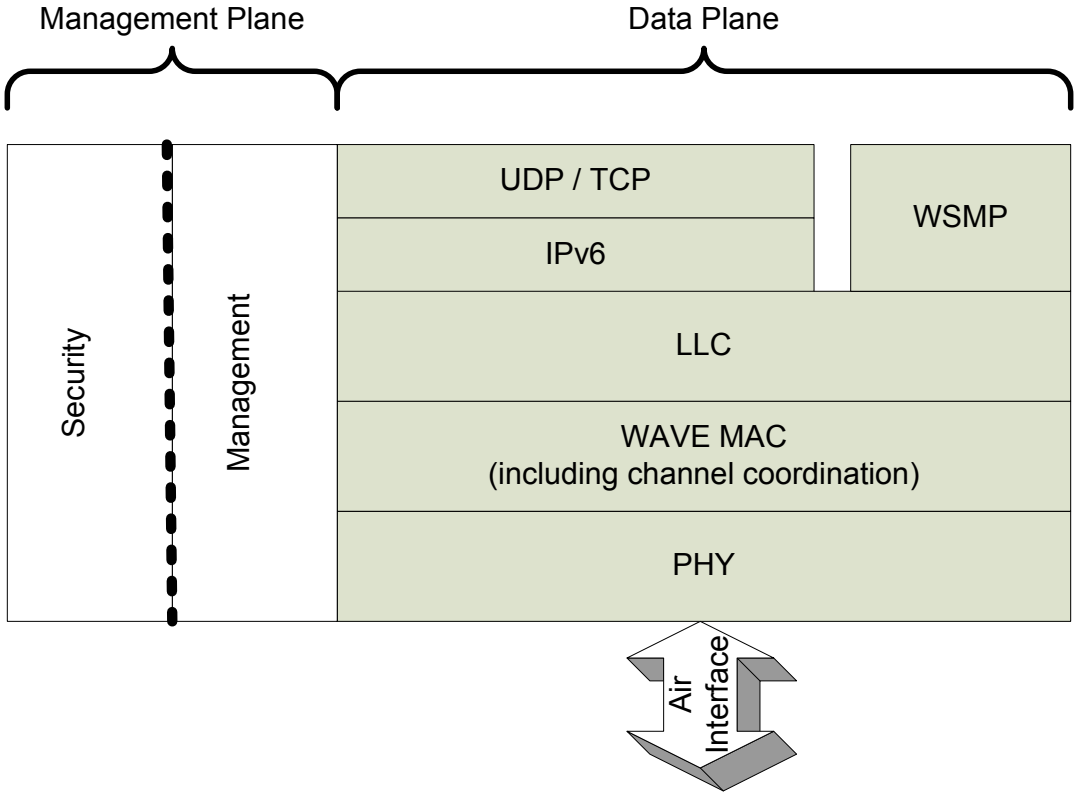


Figure 1—WAVE protocol stack

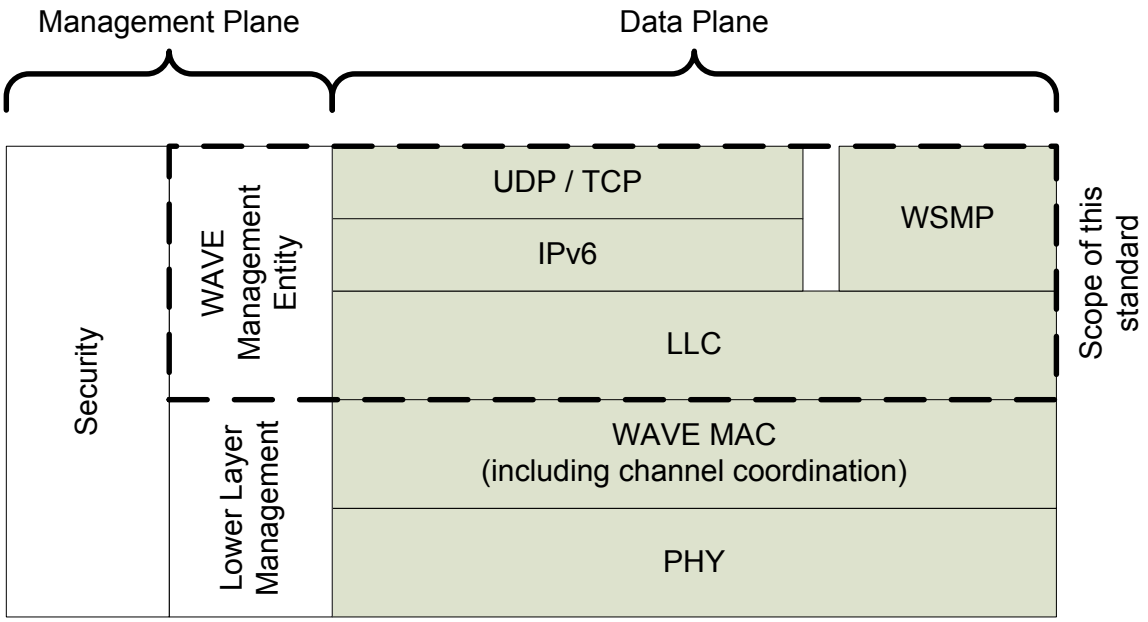


Figure 2—Scope of this standard

4.2 Data services

Data plane components of WAVE Networking Services are specified in Clause 5 and consist of the following, not all of which may be present in a given device:

— Logical Link Control

Networking Services includes the LLC sublayer, for both IP and WSMP traffic.

— Internet Protocol version 6, and higher layers such as User Datagram Protocol (UDP) and Transmission Control Protocol (TCP)

Networking Services accepts data from higher layers for transmission over the Internet Protocol stack and delivers received Internet Protocol data to higher layers.

— WSMP

Networking Services accepts data from higher layers for transmission over the WAVE Short Message Protocol and delivers received WSM data to higher layers.

4.3 Management services

Management plane components of WAVE Networking Services are specified in Clause 6 and consist of the following functions performed by the WAVE Management Entity:

— Service requests and channel access assignment

The WME processes service requests from higher layers, provides service channel access to satisfy service requests, and causes WAVE Service Advertisements, Timing Advertisements, and general management data to be transmitted.

— Management data delivery

The WME accepts management data received over the air, and either processes it or passes it to a designated management entity.

— WAVE Service Advertisement monitoring

The WME monitors and verifies services advertised by other WAVE devices for use by higher layers and management functions.

— IPv6 configuration

The WME configures the local IP protocol stack using data received from other WAVE devices.

— MIB maintenance

The WME maintains a MIB containing configuration and status information.

Other device management functions may be accessed directly, i.e., not via the Networking Services specified in this standard, via the primitives specified in IEEE Std 802.11 or the MLMEX-SendPrimitive.request specified in IEEE Std 1609.4.

5. Data plane

5.1 General

This clause specifies the components of the Networking Services data plane, which are optimized for air interface efficiency and low latency in support of vehicular applications. WSMP is specified in this standard. For WSMs, the channel, transmit power, and data rate are set by higher layers on a per-message basis. Networking services also specifies the use of the Internet Protocol IPv6 and supports transport protocols such as UDP and TCP. For IP datagrams, the channel, transmit power, and data rate to be used are stored in a transmitter profile. A Networking Services implementation shall support either IPv6 or WSMP or both.

Figure 3 and Figure 4 illustrate the flow of information associated with IP and WSM data, respectively. The primitives defined for exchanging information between the data plane components are described in Clause 7.

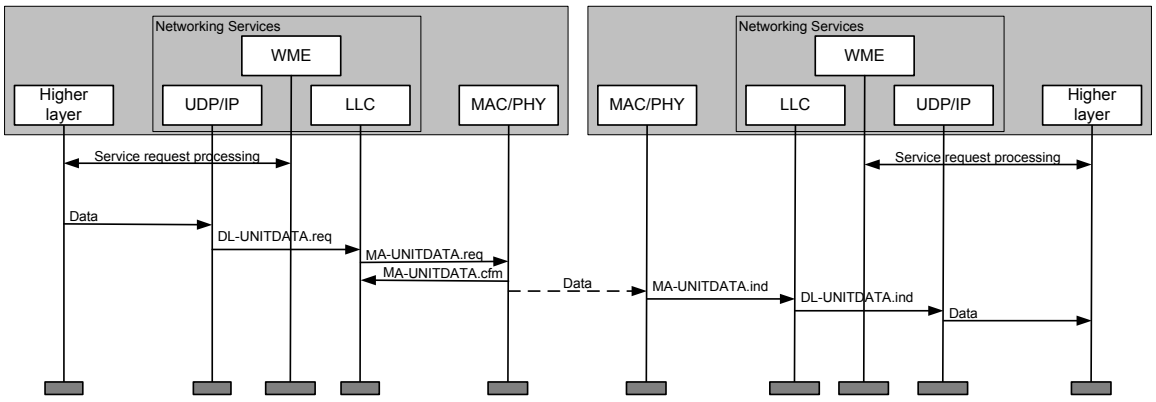


Figure 3—IP data flow

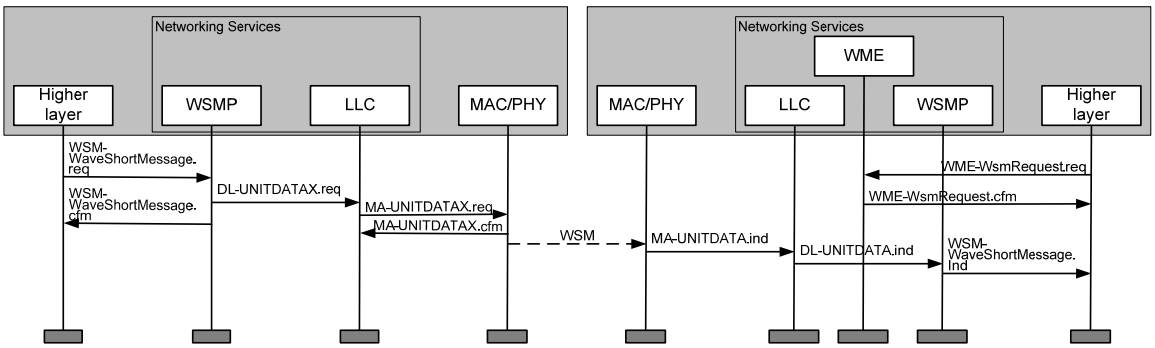


Figure 4—WSM flow

5.2 Logical Link Control

A Networking Services implementation shall support the connectionless unacknowledged Type 1 Operation of the LLC as specified in ISO/IEC 8802-2 [IEEE Std 802.2™], the Subnetwork Access Protocol (SNAP) specified in IEEE Std 802, and the standard for transmission of IP datagrams over IEEE 802 networks specified in IETF RFC 1042. Different Ethernet Type (*Ethertype*) values identify the different network layer protocols used in WAVE.

When the LLC receives an MAC service data unit via an IEEE 802.2 MA-UNITDATA.indication primitive, it checks the *Ethertype* in the SNAP header and extracts the packet. If the *Ethertype* value is 0x86DD, the LLC delivers the packet to IPv6. If the *Ethertype* value is 0x88DC, the LLC delivers the packet to WSMP. In either case, the packet is delivered to the higher layer via the IEEE 802.2 DL-UNITDATA.indication primitive.

When IPv6 uses an IEEE 802.2 DL-UNITDATA.request primitive to request transmission of a packet, the LLC protocol shall set the *Ethertype* to 0x86DD. It will encapsulate the packet in LLC and SNAP headers and pass the data to the lower layers using an IEEE 802.11 MA-UNITDATA.request primitive. Similarly, when WSMP uses a DL-UNITDATAX.request primitive (see 7.5) to request transmission of a packet, the LLC protocol shall set the *Ethertype* to 0x88DC. It will encapsulate the packet in LLC and SNAP headers and pass the data to the lower layers using an IEEE 1609.4 MA-UNITDATAX.request primitive. An example of the LLC and SNAP headers is shown in Figure 5.

To allow higher layer control of WAVE Short Message transmit parameters, the LLC service access point has been extended for WAVE Short Messages as specified in 7.5.

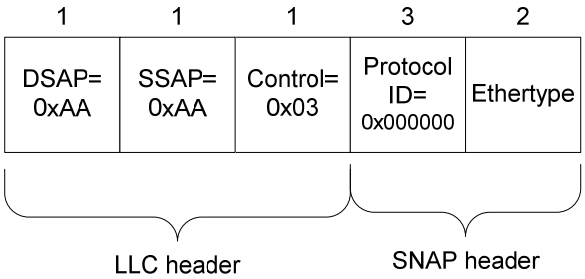


Figure 5—LLC and SNAP headers

5.3 Internet Protocol version 6

Support for IPv6 is conditionally mandatory as specified in 5.1. An IPv6 implementation shall comply with IETF RFC 2460. Related features defined in other RFCs are specified in 6.5. IP traffic is sent and received through the LLC sublayer as specified in 5.2.

5.4 Other IP-based protocols

WAVE devices may implement any Internet Engineering Task Force (IETF) protocol, but care should be taken not to compromise interoperability with other WAVE devices. Examples of IETF protocols are UDP (IETF RFC 768) and TCP (IETF RFC 793).

5.5 WAVE Short Message Protocol (WSMP)

5.5.1 General

Support for WSMP is conditionally mandatory as specified in 5.1. Acceptance of a received frame at the MAC layer is determined by the destination address; delivery of a received WAVE Short Message to a higher layer entity is determined by PSID. A WSM may be accepted by one or more destination devices, depending on the type of MAC-level addressing (e.g., individual or group) used. Note that MAC addresses may be changed for pseudonymity as specified in IEEE Std 1609.4.

Within a receiving device, the message will be delivered by WSMP to any higher layer entities with interest in the associated PSID. Interest is indicated via the WSM service request procedure specified in 6.2.2.1.4. Higher layer entities take responsibility for message signing and authentication (per IEEE P1609.2) and for providing the channel information for transmission. The WSMP header is specified in 8.3.

5.5.2 WSM transmission

On receipt of a WSM-WaveShortMessage.request from a higher layer, WSMP shall calculate the length of the WSMP header and verify that the length of that WSMP header combined with the accompanying *WSMData* is less than the value of the WME MIB parameter *WsmMaxLength*. Upon successful verification of *WsmMaxLength*, WSMP shall generate the WSMP header per the format specified in 8.3 for inclusion in an LLC DL-UNITDATAX.request (see 7.5.2). Upon failed verification of *WsmMaxLength*, the WSM is not passed and the failure is indicated in the resulting WSM-WaveShortMessage.confirm.

DL-UNITDATAX.request parameters are set as follows:

source_address	Set to the IEEE 802.11 *dot11StationID* for the WAVE interface.
destination_address	Set to the *Peer MAC address* from the WSM-WaveShortMessage.request.
data	Set as described in the subsequent discussion.
priority	Set to the *User Priority* from the WSM-WaveShortMessage.request.
Channel Identifier	Set to the *Channel Identifier* from the WSM-WaveShortMessage.request.
DataRate	Set to the *DataRate* from the WSM-WaveShortMessage.request.
TxPwr_Level	Set as described in the subsequent discussion.
WsmExpiryTime	Set to the *WsmExpiryTime* from the WSM-WaveShortMessage.request.

The *data* parameter consists of a WAVE Short Message, which is formed from the WSMP header described earlier in this subclause and the *Data* from the WSM-WaveShortMessage.request, as described in 7.3.2.

Transmit Power Level, received in the WSM-WaveShortMessage.request, is converted to *TxPwr_Level* in the DL-UNITDATAX.request as follows. *TxPwr_Level* is set to the index of the highest value of IEEE 802.11 MIB *Dot11PhyTxPowerEntry* that does not exceed *Transmit Power Level*. If there is no such value, then the request should be rejected.

5.5.3 WSM reception

On receipt of DL-UNITDATA.indication from LLC with the *WAVE Element ID* in the WSMP header indicating WAVE Short Message, WSMP shall pass the received information to the destination higher layer entity or entities, with the destination entity determined from the WSMP header *ProviderServiceIdentifier* and the MIB *WsmServiceRequestTable*. On receipt of DL-UNITDATA.indication from LLC with the *WAVE Element ID* in the WSMP Header indicating a WSMP WAVE element besides WAVE Short Message (e.g., WSMP safety supplement [WSMP-S]), WSMP shall pass the received information to the appropriate entity, as indicated by the reference listed in Annex E. In either case, the WSM-WaveShortMessage.indication primitive is used to deliver WSM information to higher layer entities. Annex E specifies WSMP WAVE *Element ID* values; Annex F specifies functionality for WSMP-S. A *WsmVersion* value higher than that specified in 8.2.5.8.3 may cause WSMP to fail to parse the message.

6. Management plane

6.1 General

The following subclauses specify the networking functions associated with the management plane. The primitives for exchanging information between the WME and other internal functional entities are described in Clause 7. Use of the WME-Notification.indication primitive (e.g., notifying a higher layer that a service request has been matched to an available service, or that an SCH has been assigned) is not explicitly specified, but it is left as choice for implementers.

Networking Services management functions provide the means to control transmit parameters (data rate and power) of data plane traffic; the data rates and power levels for transmitting management frames are specified in IEEE Std 1609.4.

6.2 Service requests and channel access assignment

6.2.1 General

The WME accepts service requests of several types from higher layer entities. To satisfy service requests, the WME may provide access to SCHs and may cause management messages to be periodically transmitted. Providing SCH access entails causing the device transceiver to be tuned to a specific channel during one or more channel intervals, via request to the MAC sublayer management entity (MLME).

For convenience, two WAVE device roles are defined. The provider role is assumed by a device transmitting WAVE Service Advertisements (WSAs) indicating its availability for data exchange on one or more SCHs. The user role is assumed by a device monitoring for received WSAs, with the potential to participate in the SCH data exchange. A WAVE device may assume one, both, or neither role.

The various service request types, the resulting channel access assignments, and the messages automatically generated are summarized in Table 1.

Table 1—Service request summary

Service request type	Reference	Primary purpose	Resulting channel/interval assignment	Message generation
Provider	7.4.2.2	Service advertisement/ SCH participation	SCH in SCH interval, or SCH in SCH and CCH intervals	VSA frame containing WSA
User	7.4.2.4	Service availability notification/ SCH participation	SCH in SCH interval, plus optionally SCH in CCH interval	none
WSM	7.4.2.6	Received message delivery	none	none
CCH	7.4.2.8	Control channel participation	CCH in CCH or SCH interval or both	none
Management data	7.4.2.10	Management data distribution	CCH or SCH; CCH or SCH interval or both	VSA frame
Timing advertisement	7.4.2.13	Time distribution	CCH or SCH; CCH or SCH interval or both	Timing Advertisement (TA) frame

To facilitate the specification, a distinction is made between single-physical layer (PHY) devices and multi-PHY devices (see IEEE Std 1609.4). A single-PHY device is a WAVE device not capable of simultaneous operation on multiple radio channels. A multi-PHY device is a WAVE device capable of simultaneous operation on multiple radio channels. A switching device has at least one PHY that is capable of switching between channels (e.g., SCH and CCH). Depending on the characteristics and usage of the WAVE device at a given time, it might support any number of channel usage models, including the following:

- Operation only on the CCH (e.g., by a single-PHY device). See Figure 6(a).
- Operation only on an SCH (e.g., by a single-PHY device).
- Alternating operation on CCH and SCH, in CCH intervals and SCH intervals, respectively (a switching device). See Figure 6(b).
- Simultaneous operation on CCH and one or more SCHs (by a multi-PHY device).
- The capabilities of the WAVE device should be considered by the WME in assigning channel access to satisfy requested services.

Several examples of channel access (applicable to a single-PHY WAVE device) are illustrated in Figure 6. The first option shows continuous operation on the CCH. The second option shows alternating operation on the CCH and an SCH during the CCH interval and the SCH interval, respectively, where SCH access is provided based on a request from a higher layer. The third option shows a variation where SCH access is provided on an immediate and extended basis at the request of the higher layer, with SCH operation during some CCH intervals. Networking Services may make channel access assignments based on the service requests described in this subclause, resulting in any of the following:

- Alternating operation with SCH access during SCH intervals, and CCH access during CCH intervals
- Immediate or extended SCH access during SCH and CCH intervals
- CCH access during CCH and/or SCH intervals (used for example to override immediate or extended SCH access in support of user service requests)

Examples of information flows associated with channel access under different scenarios are shown in Annex D.

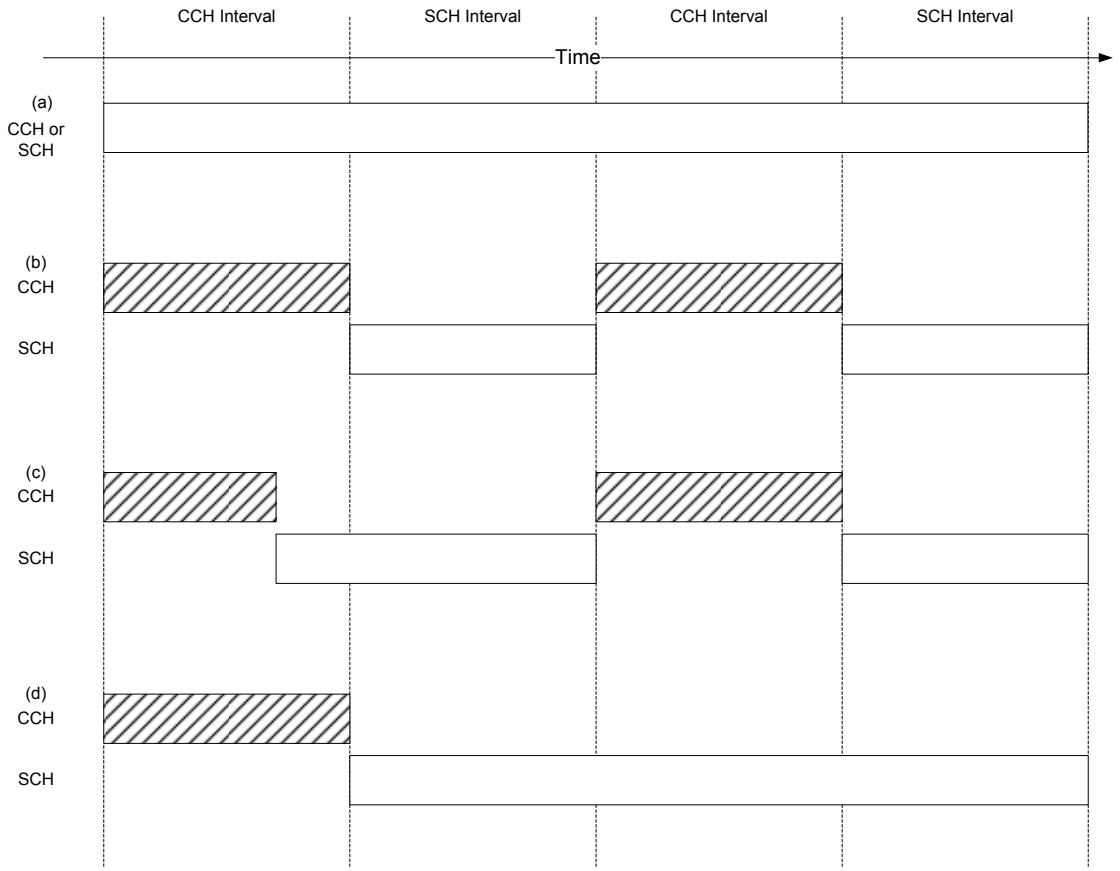


Figure 6 —Channel access options: (a) continuous, (b) alternating, and (c) immediate and extended

6.2.2 Service requests

This subclause generally describes internal device operation and is, therefore, optional. The WME maintains information about requested and available services for use in determining channel access assignments. Requested service information is maintained in the MIB *ProviderServiceRequestTable*, *UserServiceRequestTable, WsmServiceRequestTable*, and *CchServiceRequestTable*. Service requests may be made through the WME service access point (SAP) as described in 7.4, or via other mechanisms (e.g., preloaded information for static services).

Figure 7 illustrates actions available via service request primitives.

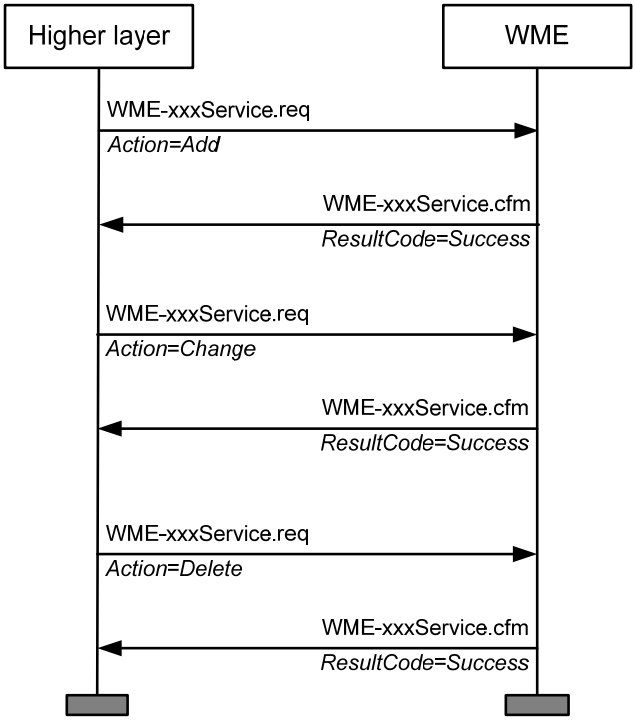


Figure 7—Service request information flow

6.2.2.1 Adding service requests

6.2.2.1.1 General

A service request may be of the form WME-ProviderService.request, WME-UserService.request, WME-WSMService.request, WME-CchService.request, WME-ManagementDataService.request, or WME-TimingAdvertisementService.request as described in 7.4.

6.2.2.1.2 Provider service request

A provider service request indicates to the WME that a higher layer wishes to have WSAs generated on its behalf and SCH access provided. Acceptance of a WME-ProviderService.request results in the population of a corresponding *ProviderServiceRequestTableEntry* in the MIB and consideration of the service request in determining channel access assignments. The device sending the WSA performs the provider role; a device accepting the WSA performs the user role.

In addition to advertising SCH-based services, the provider service request may be used to advertise local CCH channel configuration information (e.g., enhanced distributed channel access [EDCA] parameter set) in the WSA. Although not explicitly described in this document, the WSA may also be used to distribute system information (e.g., *Country String* in the WSA header extension). In these cases, there may or may not be service information in the WSA.

6.2.2.1.3 User service request

A user service request indicates to the WME that a higher layer entity has an interest in available services meeting specified criteria. The request indicates the action to be taken when the WME recognizes such available services, including assigning SCH access. A user service request may be employed by a higher layer entity desiring unilateral service channel access (either alternating or extended) without regard to available services, through the *UserRequestType* of Auto-access Unconditionally. Acceptance of a WME-UserService.request results in the population of a corresponding *UserServiceRequestTableEntry* in the MIB and in consideration of the service request in determining channel access assignments.

6.2.2.1.4 WSM service request

A WSM service request indicates to the WME that a higher layer entity wishes to receive WSMs addressed to a particular PSID. Acceptance of a WME-WSMService.request results in the population of a corresponding *WsmServiceRequestTableEntry* in the MIB and causes Networking Services to deliver any received WSMs with matching *ProviderServiceIdentifier* to the requesting higher layer entity.

6.2.2.1.5 CCH service request

A CCH request indicates that a higher layer entity requires ongoing CCH access during particular channel intervals (see IEEE Std 1609.4), e.g., for WSM activity or for WSA reception. Acceptance of a WME-CchService.request results in the population of a corresponding *CchServiceRequestTableEntry* in the MIB and in consideration of the service request in determining channel access assignments.

The CCH request indicates whether it applies to the CCH interval, the SCH interval, or both. A CCH service request indicating “CCH interval” does not affect decisions to assign SCH access during the SCH interval, but it can be used to prevent assigning SCH *ImmediateAccess* or *ExtendedAccess* (see 6.2.3) during the CCH interval based on a user service request with lower service priority.

6.2.2.1.6 Management data service request

A management data service request indicates to the WME that another management entity wishes to have Vendor Specific Action frames transmitted on its behalf and, possibly, SCH access provided. Acceptance of a WME-ManagementDataService.request results in the consideration of the service request in determining channel access assignments and in generating VSA requests to the MLME.

6.2.2.1.7 Timing advertisement service request

A timing advertisement service request indicates to the WME that another management entity wishes to have Timing Advertisement frames transmitted on its behalf and, possibly, channel access provided. Acceptance of a WME-TimingAdvertisementService.request results in the consideration of the service request in determining channel access assignments and in generating TA requests to the MLME.

6.2.2.2 Changing service requests

Upon receipt of any service request with *Action* equal to Change, the corresponding service information is updated in the MIB and in the WME channel access assignment function. Changing of an ongoing WAVE Service Advertisement is covered in more detail in 6.2.4.2.2.

6.2.2.3 Deleting service requests

Upon receipt of any service request with *Action* equal to Delete, the corresponding service information is removed from the MIB and from the WME channel access assignment function. Additional processing is specified in 6.2.3.7 and 6.2.4.5.

6.2.3 Channel access assignment

6.2.3.1 General

Each of the features specified in the subclauses of 6.2.3 are optional. Networking Services assigns local radio channel resources in support of service requests. This includes assigning radio resources to particular service channels to provide data communication opportunities for the requesting entities. The primitive exchanges related to channel access assignment for provider and user service requests are illustrated in Figure 8 and described here and in more detail in the following subclauses. Additional flows are depicted in Annex D.

Figure 8 illustrates the provider WME initiating WSA broadcast and SCH access, followed by the user joining the provider on the SCH. The first step is signing the WSA information via the WaveSecurityServices-SignedWsa.request. The provider WME then initiates WSA broadcasts on the CCH via the MLMEX-VSA.request and alternating SCH access via the MLMEX-SCHSTART.request. The user, monitoring the CCH, receives the WSA, indicated by the MLMEX-VSA.indication, recognizes a service of interest, and begins alternating SCH access via the MLMEX-SCHSTART.request. In each case, if the service employs IP, the MLMEX-REGISTERTXPROFILE.request is also used. Subsequently, higher layer traffic may be exchanged between the devices on the SCH.

The status of each service (e.g., satisfied or not) is maintained in a status field in its request table in the MIB.

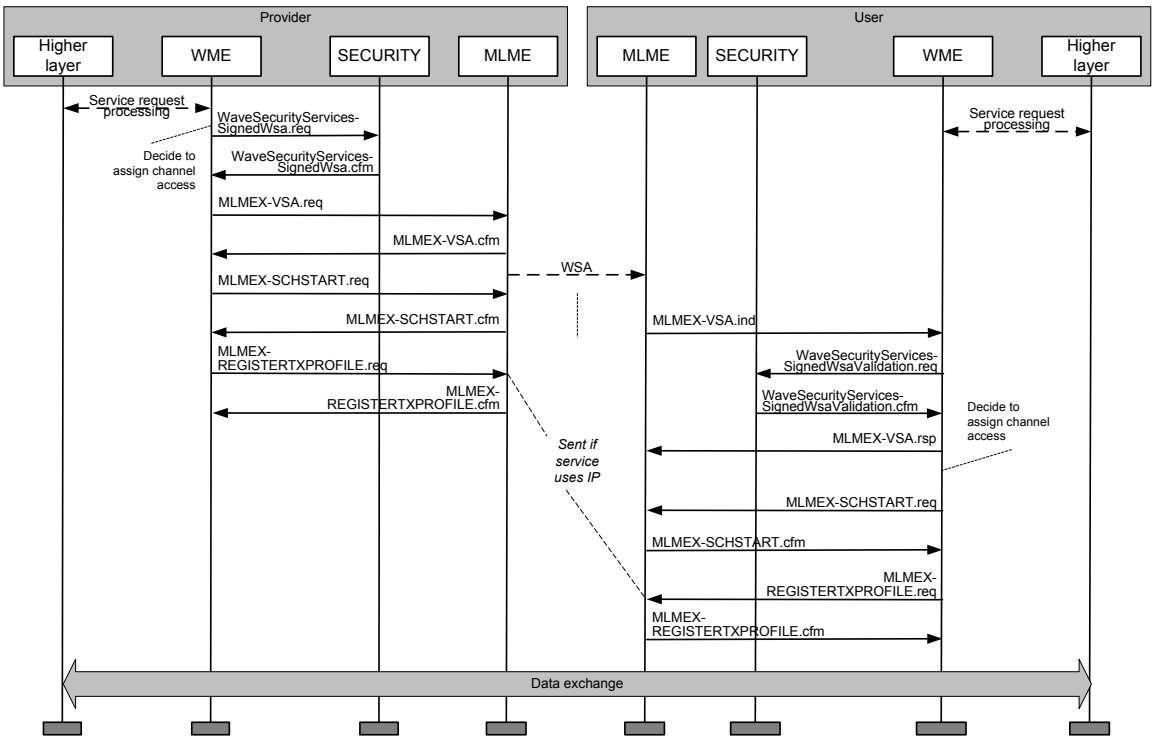


Figure 8—Example information flow for channel access assignment

6.2.3.2 Channel access assignment algorithm

By default, WAVE devices operate on the control channel. Access to service channels is determined by the WME based on service requests and other criteria. The channel access assignment algorithm is not specified, but the following criteria should be considered by algorithm developers:

- Number of channels that can be simultaneously accessed by the device's radio equipment
- Service requests (6.2.2)
- Available services (6.4) matching user service requests
- The relative priorities of the requested services
- The *Channel Access* of the service, i.e., the channel intervals in which it is active

In general, the access assignment algorithm should give precedence to requests with higher *ServicePriority*. CCH service requests with a higher *ServicePriority* should take precedence over lower priority assignments that would interfere with ongoing CCH operation. In this case, SCH access in support of a user service request may be assigned but without the requested *ImmediateAccess* or *ExtendedAccess*.

Other factors may be considered, such as the following:

- The geographic proximity to the transmitter of the WSA
- Link quality associated with the transmitter of the WSA
- Service duration

After channel access has been assigned, certain events such as the following should cause the assignment to be ended. Refer to 6.2.3.7:

— Service has been deleted (6.2.2.3)

— User service request no longer matching an available service (6.4)

— Need to satisfy a service request with higher service priority

— Loss of lower layer synchronization indicated by receipt of MLMEX-SCHEND.indication per IEEE Std 1609.4

— Other factors, such as poor *Link Quality* (6.4.3)

6.2.3.3 Assignments in support of provider service requests

When service channels are assigned to fulfill provider service requests, in addition to causing MAC sublayer generation of Vendor Specific Action frames by sending an MLMEX-VSA.request as specified in 6.2.4.2, the WME triggers MAC sublayer processing by sending an MLMEX-SCHSTART.request to start service channel access.

If the WME-ProviderService.request includes *IPv6Address*, indicating a need for IPv6 communications, the WME also sends a MLMEX-REGISTERTXPROFILE.request. Refer to Figure 8.

MLMEX-SCHSTART.request parameters are set as follows:

Channel Identifier	This defines the channel assigned. Set to the channel indicated by the MIB *ProviderChannelInfoTableEntry*.
OperationalRateSet	Not used.
EDCA Parameter Set	If present, insert values from MIB *ProviderChannelInfoTableEntry* associated with the channel.
ImmediateAccess	Set to FALSE.
ExtendedAccess	Set to 0.

MLMEX-REGISTERTXPROFILE.request parameters are set as follows:

Channel Identifier	Set equal to *Channel Identifier* in the MLMEX-SCHSTART.request.
Adaptable	From the MIB *ProviderChannelInfoTableEntry* associated with the channel.
TxPwr_Level	Set to the index of the highest value of IEEE 802.11 MIB *Dot11PhyTxPowerEntry* that does not exceed the MIB *ProviderChannelInfoTransmitPowerLevel*. If there is no such value, then the service request should be rejected.
DataRate	From the MIB *ProviderChannelInfoTableEntry*.

6.2.3.4 Assignments in support of user service requests

Channel access is assigned by the WME, e.g., to satisfy a user service request when the WME detects a suitable match between a MIB *AvailableServiceEntry* and a *UserServiceEntry*. When channel access is assigned to satisfy a user service request, the WME sends an MLMEX-SCHSTART.request, with parameters set as follows. Refer to Figure 8:

Channel Identifier	From the MIB *UserAvailableChannelNumber* and *UserAvailable OperatingClass*.
OperationalRateSet	From the MIB *UserAvailableServiceTableEntry ChannelInfo*.
EDCA Parameter Set	If present, insert values from MIB *UserAvailableServiceTableEntry ChannelInfo*.
ImmediateAccess	From the MIB *UserServiceRequestTableEntry*, as modified in 6.2.3.2.
ExtendedAccess	From the MIB *UserServiceRequestTableEntry*, as modified in 6.2.3.2.

If there exists an *IPv6Address* in the *UserAvailableServiceTableEntry*, the WME also sends a MLMEX-REGISTERTXPROFILE.request, with parameters set as follows:

Channel Identifier	From the MIB *UserAvailableChannelNumber* and *UserAvailable OperatingClass*.
Adaptable	From the MIB UserAvailableAdaptable.
TxPwr_Level	Set to the index of the highest value of IEEE 802.11 MIB *Dot11PhyTxPowerEntry* that does not exceed *UserAvailable TransmitPowerLevel*. If there is no such value, then the service request should be rejected.
DataRate	From the MIB *UserAvailableDataRate*.

6.2.3.5 Assignments in support of management data requests

When management data service requests are fulfilled, the WME triggers MAC sublayer processing and generation of Vendor Specific Action frames by sending an MLMEX-VSA.request as specified in 6.2.4.4 to start WSA transmission and if needed an MLMEX-SCHSTART.request to start service channel as follows:

Channel Identifier	Set to the *Channel Identifier* of the triggering WME-Management DataService.request.
OperationalRateSet	Not used.
EDCA Parameter Set	Not used.
ImmediateAccess	If *Channel Interval* of the related MLMEX-VSA.request indicates both CCH and SCH interval, set to TRUE; otherwise set to FALSE.
ExtendedAccess	If *Channel Interval* of the related MLMEX-VSA.request indicates both CCH and SCH interval, set to Indefinite; otherwise set to 0.

6.2.3.6 Assignments in support of timing advertisement requests

When timing advertisement service requests are fulfilled, the WME triggers MAC sublayer processing and generation of Timing Advertisement frames by sending an MLMEX-TA.request as specified in 6.2.4.2.2 to start TA transmission and if needed an MLMEX-SCHSTART.request to start service channel access as follows:

Channel Identifier	Set to the *Channel Identifier* of the triggering WME-Timing AdvertisementService.request.
OperationalRateSet	Not used.
EDCA Parameter Set	Not used.
ImmediateAccess	If immediate access can be scheduled as described in 6.2.3.2, and if *Channel Interval* of the related MLMEX-TA.request indicates both CCH and SCH interval, set to TRUE; otherwise set to FALSE.
ExtendedAccess	If extended access can be scheduled as described in 6.2.3.2, and if *Channel Interval* of the related MLMEX-TA.request indicates both CCH and SCH interval, set to Indefinite Access; otherwise set to 0.

6.2.3.7 Ending service channel access assignments

When channel access is no longer required as described throughout 6.2.3, the WME causes MAC channel assignment processing to terminate by sending the following primitives.

If the channel access has an associated automatic message generation, the message generation is ended as specified in 6.2.4.5.

The WME sends an MLMEX-SCHEND.request, with *Channel Identifier* set equal to *Channel Identifier* in the associated MLMEX-SCHSTART.request.

If the service is an IP service (indicated by the sending of MLMEX-REGISTERTXPROFILE.request), the WME also sends a MLMEX-DELETETXPROFILE.request, with *Channel Identifier* set equal to *Channel Identifier* in the associated MLMEX-REGISTERTXPROFILE.request.

Information flow associated with ending channel access assignments is illustrated in Figure 9.

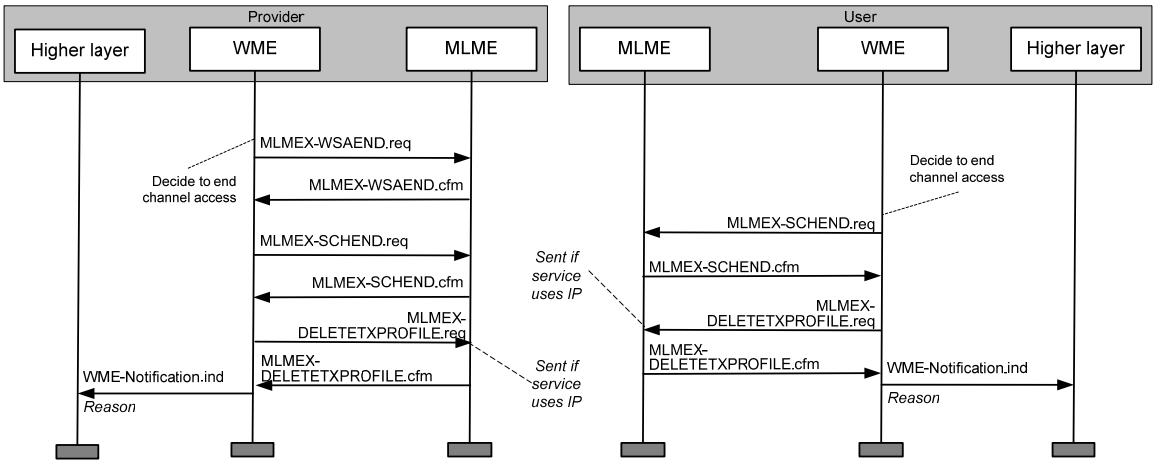


Figure 9—Ending channel access assignment example

6.2.4 Automatic message generation

6.2.4.1 General

Each of the features specified in the subclauses of 6.2.4 is optional. Satisfying certain service requests results in the WME triggering automatic management message generation by the MLME as specified in 6.2.4.2 through 6.2.4.5.

6.2.4.2 VSA generation for service announcement

6.2.4.2.1 Initiating advertisements

Figure 8 shows the primitives supporting the flow of service advertisements between WAVE devices. The content and format of the advertisements are specified in 8.2.

The WME constructs the *WaveServiceAdvertisement*, which may include *ServiceInfo*, and *ChannelInfo* derived from the MIB *ChannelInfoEntry* corresponding to any *Channel Number*(s) contained in the *ServiceInfo*. If the *ServiceInfo* indicates IP Service and a *WaveRoutingAdvertisement* exists in the WME MIB, the WRA may be included in the *WaveServiceAdvertisement* to announce the availability of internetwork connectivity. See Figure 17.

The WME constructs the *1609Dot2Message*, consisting of either an *Unsecured WSA* or a *Secured WSA*, from the *WaveServiceAdvertisement* as specified in the subsequent discussion and in Figure 16. In either case, *protocol_version* and *type* values are specified in 8.2.1 and IEEE P1609.2.

If any service requires security, as indicated by its entry in the MIB *ProviderServiceRequestTable* with WSA Type set equal to Secured WSA, the *WaveServiceAdvertisement* is digitally signed to generate a

Secured WSA via a WaveSecurityServices-SignedWsa.request. *Lifetime* indicates when the signature becomes invalid. It is encoded per IEEE P1609.2 and is set to the current time plus the smallest of any *SignatureLifetime* (from the WME-ProviderService.request) associated with the services being advertised.

WaveSecurityServices-SignedWsa.request parameters are set as follows:

WSA Data	Set to the *WaveServiceAdvertisement*.
Permissions	An array built from contents of *ProviderServiceRequestTableEntry* as specified in IEEE P1609.2.
Lifetime	As specified in the preceding discussion.

The WME generates a MLMEX-VSA.request to trigger VSA transmission. It constructs the *Vendor specific content* by concatenating a *ContentDescriptor* and the *1609Dot2Message*. The *ContentDescriptor* is set to the value WAVE Service Advertisement.

MLMEX-VSA.request parameters are set as follows:

Destination MAC address	Set to the *Destination MAC Address* of the triggering WME-Provider Service.request if present; otherwise set to the broadcast MAC address.
Management ID	Set to IEEE P1609.3.
Vendor specific content	*ContentDescriptor* and *1609Dot2Message* as described previously.
Repeat Rate	Set to the *Repeat Rate* of the triggering WME-ProviderService.request.
Channel Identifier	*Channel Identifier* of the CCH.
Channel Interval	Set to CCH Interval. This standard assumes, but does not require, that the WSA is sent only on the CCH during the CCH Interval.

6.2.4.2.2 Changing advertisements

The WME may change the contents of WAVE Service Advertisements while the advertisements are ongoing. This is useful, for example, when a new provider service request is received compatible with *ChannelInfo* in the existing WSA, when one of multiple advertised services is deleted via WME-ProviderService.request with *Action* equal to Delete, or if a higher layer entity changes the PSC in the advertised service via a WME-ProviderService.request with *Action* equal to Change.

If changes are made to the signed portions of a *Secured WSA*, the advertisement is signed and secured using the WaveSecurityServices-SignedWsa.request. Before the *Lifetime* of an ongoing WSA is reached, the WME should update the signature via a WaveSecurityServices-SignedWsa.request.

Changes to ongoing advertisements are made by accessing the WSA in the MLME MIB via MLME-Set.request. An information flow illustrating WSA updates is illustrated in Figure 10.

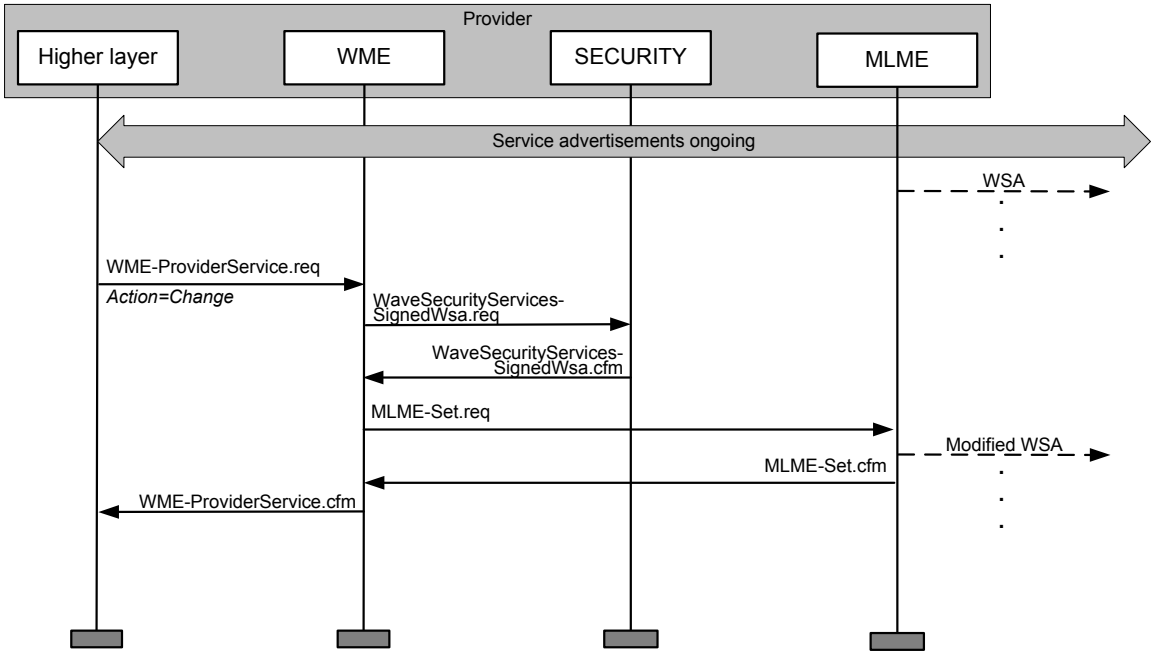


Figure 10—Example information flow for WSA update

6.2.4.3 TA generation for time distribution

Figure 11 shows the primitives supporting the flow of timing information between WAVE devices. Note that the WME-TimingAdvertisement.confirm primitive indicates that the request was successfully accepted by the WME and will be considered for scheduling. The WME-Notification.indication primitive indicates that the timing advertisement message has been scheduled.

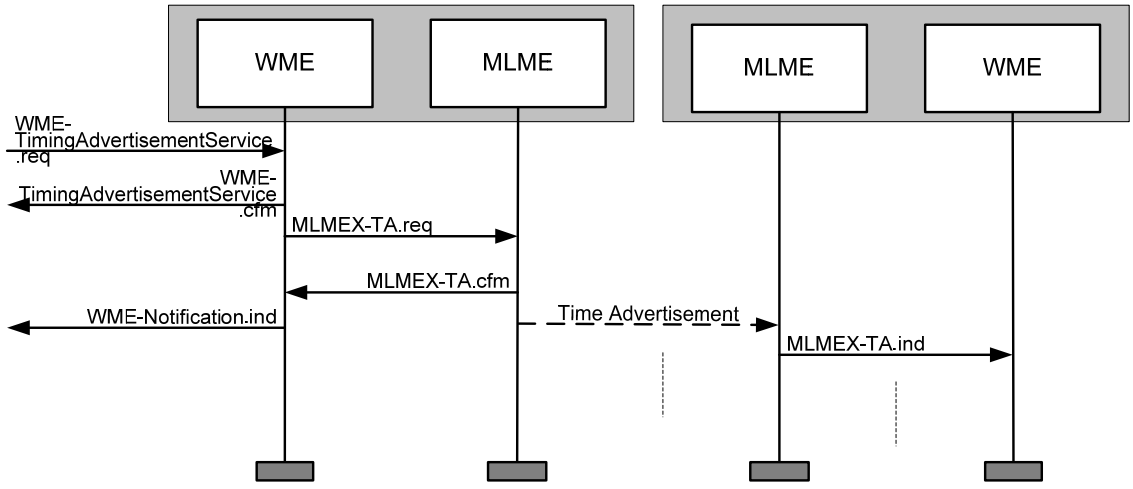


Figure 11—Information flow for timing advertisement

MLMEX-TA.request parameters are set as follows:

Destination MAC address	Set to the *Destination MAC address* of the triggering WME-TimingAdvertisementService.request.
Repeat Rate	Set to the *Repeat Rate* of the triggering WME-TimingAdvertisementService.request.
Channel Identifier	Set to the *Channel Identifier* of the triggering WME-TimingAdvertisementService.request.
Channel Interval	Set to the *Channel Interval* of the triggering WME-TimingAdvertisementService.request.
TimingAdvertisementContents	Set to the *TimingAdvertisementContents* of the triggering WME-TimingAdvertisementService.request.

6.2.4.4 VSA generation for management data delivery

Figure 12 shows the primitives supporting the flow of management data between WAVE devices. Note that the WME-ManagementDataService.confirm primitive indicates that the request was successfully accepted by the WME and will be considered for scheduling. The WME-Notification.indication primitive indicates that the vendor specific action message has been scheduled.

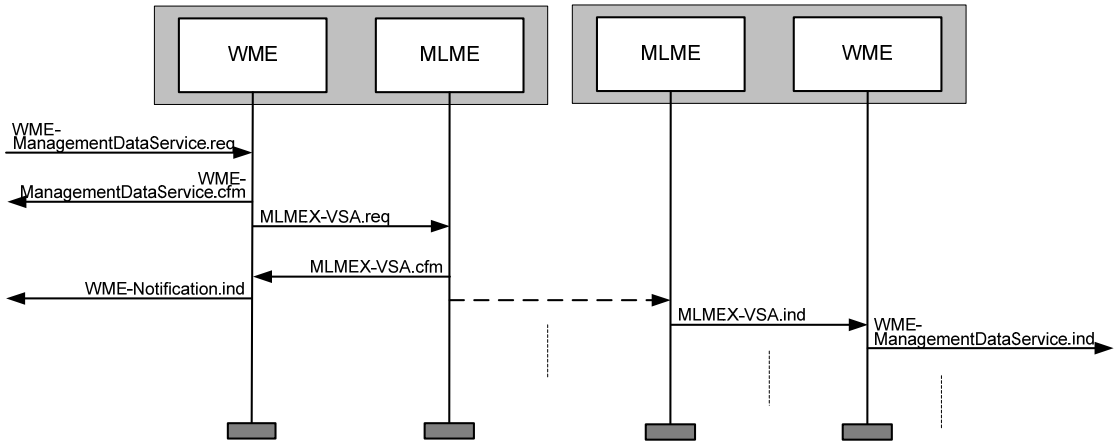


Figure 12—Information flow for management data distribution

MLMEX-VSA.request parameters are set as follows. Only one of *Management ID* and *Organization Identifier* should be present in a given request:

Destination MAC address	Set to the *Destination MAC address* of the triggering WME-Management DataService.request.
Management ID	Set to the *Management ID* of the triggering WME-Management DataService.request.
Organization Identifier	Set to the *Organization Identifier* of the triggering WME-Management DataService.request.
Vendor specific content	Set to the *Data* of the triggering WME-ManagementDataService.request.
Repeat Rate	Set to the *Repeat Rate* of the triggering WME-Management DataService.request.
Channel Identifier	Set to the *Channel Identifier* of the triggering WME-Management DataService.request.
Channel Interval	Set to the *Channel Interval* of the triggering WME-Management DataService.request.

6.2.4.5 Ending automatic message generation

6.2.4.5.1 Ending generation of VSA frames

When ending VSA generation, the WME sends an MLMEX-VSAEND.request.

6.2.4.5.2 Ending generation of TA frames

When ending TA generation, the WME sends an MLMEX-TAEND.request.

6.3 Management data delivery on receipt

Management data delivery is optional. The WME accepts management data received over the air and either processes it or passes it to a designated management entity. Delivery is performed based on the *Management ID* associated with the data. Refer to 8.1.2.

Figure 12 shows the primitives supporting the flow of management data between WAVE devices. On receipt of a MLMEX-VSA.indication with *Management ID* indicating IEEE P1609.3 (per IEEE P1609.0), the WME interprets the octet following *Management ID* as *Content Descriptor* per 8.1.2. When the *Content Descriptor* indicates a WAVE Service Advertisement, the WME performs the processing specified in 6.4.

On receipt of a MLME-VSA.indication with *Management ID* indicating another management entity (per IEEE P1609.0), the WME delivers the received data to that entity in the form of WME-ManagementDataService.indication.

6.4 WAVE Service Advertisement monitoring

6.4.1 General

WAVE Service Advertisement monitoring is optional. WAVE devices transmit announcements containing information about available services, in the form of WSAs as described in 6.2.4.2. A WAVE device receiving a WSA may maintain a collection of currently available services as specified in 6.4.2. The available services are maintained in the MIB for use by higher layer entities and the channel access assignment function. Receipt of the WSA is indicated to the WME via the MLMEX-VSA.indication with *Management ID* indicating 1609.3 and *Content Descriptor* indicating WAVE Service Advertisement as specified in 8.1.2.

6.4.2 Available services

A received WSA may contain either a *Secured WSA* or an *Unsecured WSA* as specified in 8.2.1. If a *Secured WSA* is present, the WME may invoke security verification via WaveSecurityServices-SignedWsaValidation.request. Note that receipt of multiple advertisements may result in multiple MLMEX-VSA.indications to the WME, and that the contents of the advertisement from a provider may change over time. Changes to a repeated WSA are indicated by an incremented *Change Count* as specified in 8.2.2.3.

The following *UserAvailableServiceTable* parameters are set as follows:

- *UserAvailableLinkQuality* set per 6.4.3.
- *UserAvailableServiceStatus* set per 6.2.3.1.
- *UserAvailableServiceSpecificPermissions* per the appropriate entry, if present, in the *ServiceSpecificPermissions* array returned by the WaveSecurityServices-Signed WsaValidation.confirm.
- *UserAvailableResultCode*, *UserAvailableTxLatitude*, *UserAvailableTxLongitude*, *UserAvailableTxElevation*,*UserAvailableTxPositionConfidence*, *UserAvailableTxElevation Confidence* set, if present, per the WaveSecurityServices-SignedWsaValidation.confirm.
- *UserAvailableSourceMacAddress* per the received *Source MAC Address* in the MLMEX-VSA.indication
- *UserAvailableRcpi* per the *RCPI* in the MLMEX-VSA.indication, where RCPI represents the Received Channel Power Indicator.

Other parameters shall be set per the appropriate parameters in the WSA in the received MLMEX-VSA.indication.

6.4.3 Available service link quality

When the *UserAvailableLinkQuality* associated with an available service falls below an implementation-specific threshold, the *UserAvailableServiceTableEntry* is removed from the MIB.

The link quality calculation may consider the following information received over time in MLMEX-VSA.indication primitives:

- *RCPI* represents the signal strength of WSA frames associated with the available service, from which the quality of the communications path can be inferred.
- *Repeat Rate* can be used to calculate the proportion of WSAs that were received locally. This is also an indication of communications path quality.
- Other mechanisms may be used as available.

Link Quality should be kept over multiple time intervals (e.g., 200 ms, 1 s, and 5 s) so as to provide information on link trends and stability.

6.5 IPv6 configuration

6.5.1 General

The requirements in 6.5 are mandatory for WAVE devices supporting IPv6. The IPv6 information used by stationary WAVE devices may be provided by a network administrator. A mobile WAVE device may derive the information that allows it to operate on an infrastructure network from a WAVE device connected to that network. Device link-local addresses are derived locally by any WAVE device and may be used without any external configuration information.

Each WAVE IP device shall support link-local, global, and multicast IPv6 addresses on its WAVE interface per IETF RFC 2373.

The link-local address is derived from the device MAC address per IETF RFC 2462.

The derivation of a static global address may occur through any means, such as administrative entry, and is not specified here.

For WAVE devices supporting dynamic IP addressing, the WME shall calculate the global IPv6 address via a stateless configuration procedure. This is the procedure described in IETF RFC 3513, except the WAVE device uses its MAC address in conjunction with the *IpPrefix* received in the *WaveRoutingAdvertisement* of a WAVE Service Advertisement, rather than that received in an IETF RFC 4861 Router Advertisement message. WAVE provides this IP configuration information in the WAVE Service Advertisement to minimize the need for neighbor and router discovery with their associated overhead and latency. Note that IP transmissions are prohibited on the control channel.

WAVE IP devices shall accept IPv6 packets addressed to predefined Host or Router multicast addresses per IETF RFC 2462.

6.5.2 Readdressing

For devices that support pseudonymity, it is desirable to change device addresses from time to time. On receipt of a WME-AddressChange.request, the WME resets the link-local IPv6 address to a new value. If WME-AddressChange.request includes an address value, it is used; otherwise, a randomly chosen address is used. Note that an address change may disrupt any ongoing communications. MAC address changes may be accomplished via the methods specified in IEEE Std 1609.4.

6.6 MIB maintenance

In a managed WAVE device, the WME shall maintain a MIB containing the configuration and status information identified in Annex A and specified in Annex B. The MIB is accessible to higher layers via the WME-Get and WME-Set primitives. Additionally, the device may support other MIBs related to Networking Services (e.g., MIBs relevant to IPv6 or containing vendor-specific information). MIB access is via generic primitives illustrated in Figure 13 and described in 7.4.4.

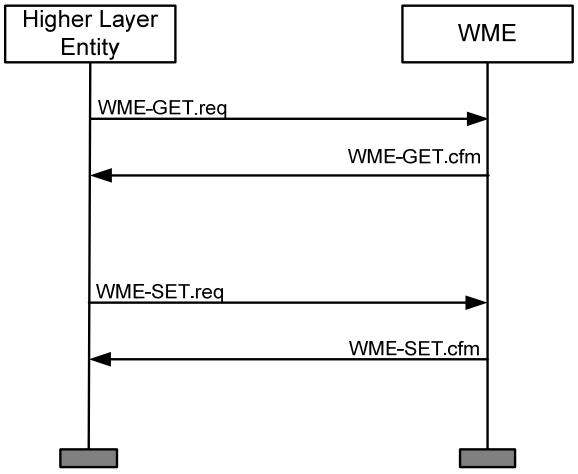


Figure 13—Information flow for MIB access

7. Service primitives

7.1 General

Service access points (SAPs) are shown in Figure 14 to support communications between WAVE Networking Services entities and other entities in the same device. The WSM SAP, WME SAP, and extensions to the link service access point (LSAP) are specified in this standard. Where there are no unique requirements to this specification (e.g., in the case of the Transport Service Access Point [TSAP]), the SAP is specified by reference to the defining standard. Individual management SAPs are not defined for the data protocols; the WME SAP gives higher layer entities access to layer management services.

The primitives used on each interface are summarized in Table 2 and described subsequently. The implementation of the primitives and their exchange protocols are not specified but are left as design decisions.

Where error indications or failure responses are considered critical to the Networking Service operation, they are specified. Otherwise, they are left undefined, as a choice for the implementer.

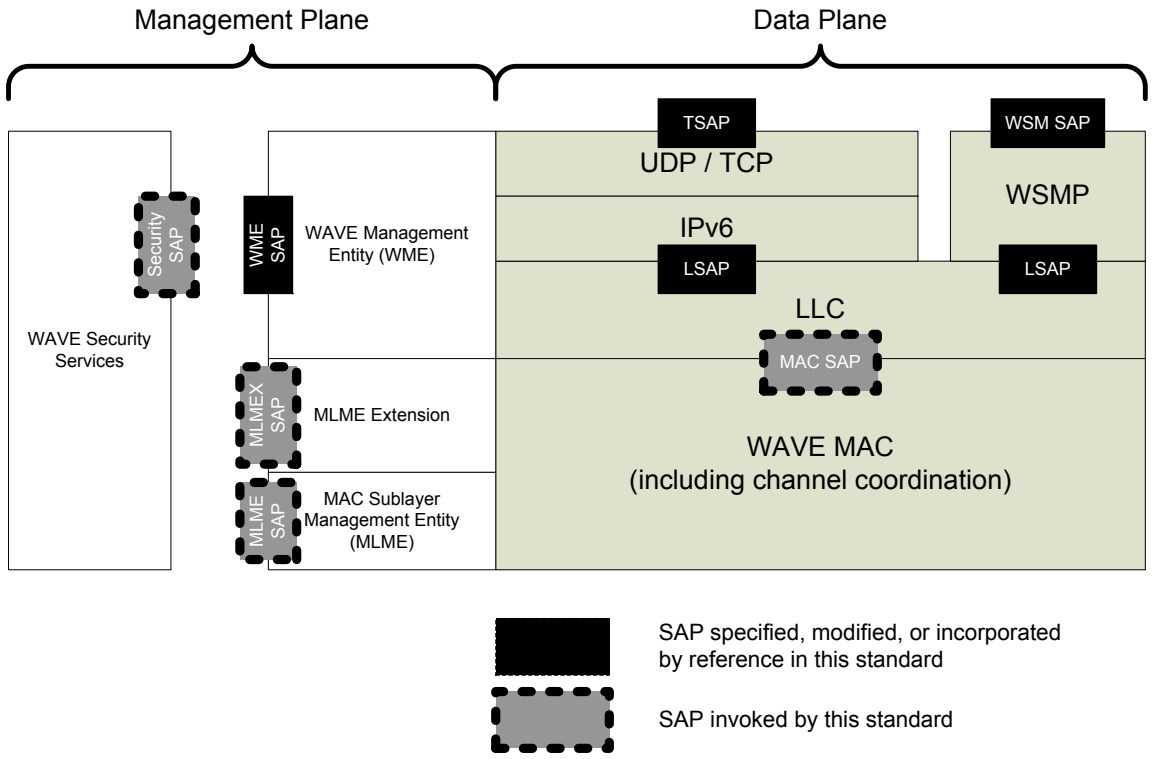


Figure 14—Service access points

Table 2—Summary of primitives

SAP	Primitive	Specified in
WSMP	WSM-WaveShortMessage.request	7.3.2
	WSM-WaveShortMessage.confirm	7.3.3
	WSM-WaveShortMessage.indication	7.3.4
WME	WME-ProviderService.request	7.4.2.2
	WME-ProviderService.confirm	7.4.2.3
	WME-UserService.request	7.4.2.4
	WME-UserService.confirm	7.4.2.5
	WME-WSMService.request	7.4.2.6
	WME-WSMService.confirm	7.4.2.7
	WME-CchService.request	7.4.2.8
	WME-CchService.confirm	7.4.2.9
	WME-ManagementDataService.request	7.4.2.10
	WME-ManagementDataService.confirm	7.4.2.11
	WME-ManagementDataService.indication	7.4.2.12
	WME-TimingAdvertisementService.request	7.4.2.13
	WME-TimingAdvertisementService.confirm	7.4.2.14
	WME-Notification.indication	7.4.3.1
	WME-Get.request	7.4.4.1
	WME-Get.confirm	7.4.4.2
	WME-Set.request	7.4.4.3
	WME-Set.confirm	7.4.4.4
	WME-AddressChange.request	7.4.5.1
	WME-AddressChange.confirm	7.4.5.2
LSAP	DL-UNITDATAX	7.5.2
	DL-UNITDATA	ISO/IEC 8802-2 [IEEE Std 802.2]
MLME	MLMEX-DELETETXPROFILE	IEEE Std 1609.4
	MLMEX-REGISTERTXPROFILE	IEEE Std 1609.4
	MLME-TA	IEEE Std 1609.4
	MLME-TAEND	IEEE Std 1609.4
	MLMEX-VSA	IEEE Std 1609.4
	MLMEX-VSAEND	IEEE Std 1609.4
	MLMEX-SCHSTART	IEEE Std 1609.4
	MLMEX-SCHEND	IEEE Std 1609.4
	MLME-GET	IEEE Std 802.11
	MLME-SET	IEEE Std 802.11
	MLME-MREPORT	IEEE Std 802.11
	MLME-MREQUEST	IEEE Std 802.11
MAC	MA-UNITDATA	IEEE Std 802.11
	MA-UNITDATAX	IEEE Std 1609.4
Security	WaveSecurityServices-SignedWsa	IEEE P1609.2
	WaveSecurityServices-SignedWsaValidation	IEEE P1609.2

7.2 Channel identification

IEEE Std 802.11 uses *Channel Number* (range, 0–200) to identify a specific radio channel within the context of a group of channels identified by *Operating Class* (see IEEE P802.11REVmb™ and Annex J of IEEE Std 802.11p) applicable to a geographic region (country) defined by *Country String*. 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 WSMP SAP

7.3.1 General

The WSM primitives allow higher layer entities to send and receive WSMs.

7.3.2 WSM-WaveShortMessage.request

7.3.2.1 Function

The WSM-WaveShortMessage.request primitive is used by a higher layer entity to request sending a WAVE Short Message.

7.3.2.2 Semantics of the service primitive

The parameters of the WSM-WaveShortMessage.request primitive are as follows:

WSM-WaveShortMessage.request (
Channel Identifier,
DataRate,
Transmit Power Level,
ProviderServiceIdentifier,
User Priority,
WsmExpiryTime,
Length,
Data,
Peer MAC address,
WSMP header extensions,
WAVE Element ID
)

Name	Type	Valid range	Description
Channel Identifier	See 7.2	See 7.2	The *Channel Number* component of *Channel Identifier* is inserted into the DL-UNITDATA.request. It indicates the channel number to be used for this transmission and may be included in the transmitted WSM per 8.3.
DataRate	Integer	2–127	Inserted into the DL-UNITDATA.request. Indicates the data rate to be used for this transmission. May be included in the transmitted WSM per 8.3. Parameter specified in IEEE Std 802.11.
Transmit Power Level	Signed integer	–127 to 127	Indicates the power level in dBm to be used for this transmission. May be included in the transmitted WSM per 8.3.
ProviderServiceIdentifier	Octet string	See 8.1.3	Used to construct the WSM per 8.3.
User Priority	Integer	0–7	Inserted into the DL-UNITDATA.request *priority*. Parameter specified in IEEE Std 802.11.
WsmExpiryTime	Integer	$0–(2^{64} – 1)$	This field is optionally generated by the higher layer originator of the message content

Name	Type	Valid range	Description
			and indicates the time at which the message is no longer valid. The lower layers may purge messages that have exceeded their *WsmExpiryTime* before transmission. *WsmExpiryTime* is compared to the value of the timing synchronization function (TSF) timer to determine the message's validity. It is not sent over the air.
Length	Integer	1–65 535	Used to construct the WSM per 8.3.
Data	Octet string	Not specified	Used to construct the WSM per 8.3.
Peer MAC Address	MACAddress	Any valid individual or broadcast MAC address	Inserted into the DL-UNITDATA.request *destination_address*. Parameter specified in IEEE Std 802.11.
WSMP header extensions	Bit string	Not specified	Indicates which of the WSMP header extension fields specified in 8.3.4 should be included in the transmitted WSM.
WAVE Element ID	Integer	128–255	Indicates the type of WSM data contained, to assist receive-side processing. Default 128 = WAVE Short Message per Annex E.

7.3.2.3 When generated

The WSM-WaveShortMessage.request primitive is generated by a higher layer entity to request sending a WAVE Short Message.

7.3.2.4 Effect of receipt

Upon receipt of the WSM-WaveShortMessage.request primitive, the WSMP delivers the WAVE Short Message to LLC.

7.3.3 WSM-WaveShortMessage.confirm

7.3.3.1 Function

This primitive is used to confirm the acceptance of a WAVE Short Message.

7.3.3.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WSM-WaveShortMessage.confirm (
ResultCode
)

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

7.3.3.3 When generated

This primitive is generated in response to WSM-WaveShortMessage.request.

7.3.3.4 Effect of receipt

No behavior is specified.

7.3.4 WSM-WaveShortMessage.indication

7.3.4.1 Function

The WSM-WaveShortMessage.indication primitive indicates to a higher layer entity that a WAVE Short Message has been received.

7.3.4.2 Semantics of the service primitive

The parameters of the WSM-WaveShortMessage.indication primitive are as follows:

WSM-WaveShortMessage.indication (
WsmpVersion
Channel Number,
DataRate,
Transmit Power Used,
ProviderServiceIdentifier,
User Priority,
Length,
Data,
Peer MAC address
)

Name	Type	Valid range	Description
WsmpVersion	Integer	0–15	Extracted from the WSMP header per 8.3.
Channel Number	Integer	0–200	If present, extracted from the WSMP header per 8.3. Parameter specified in IEEE Std 802.11.
DataRate	Integer	2–127	If present, extracted from the WSMP header per 8.3. Parameter specified in IEEE Std 802.11.
Transmit Power Used	Integer	–127 to 127	If present, extracted from the WSMP header per 8.3.
ProviderServiceIdentifier	Octet string	See 8.1.3	Extracted from the WSMP header per 8.3.
User Priority	Integer	0–7	Extracted from the DL-UNITDATA.indication *priority*.
Length	Integer	1–65 535	Extracted from the WSMP header per 8.3.
Data	Octet string	Any	Extracted from the WSM per 8.3.
Peer MAC Address	MACAddress	Any valid unicast MAC Address	Extracted from the DL-UNITDATA.indication *source_address*. Parameter specified in IEEE Std 802.11.

7.3.4.3 When generated

The WSM-WaveShortMessage.indication primitive is generated by the WSMP to deliver WSM information to a higher layer entity indicated in the MIB *WsmServiceRequestTable*.

7.3.4.4 Effect of receipt

The received information is processed as determined by the receiving higher layer entity.

7.4 WME SAP

7.4.1 General

The WME primitives allow higher layer entities to access WME functions (e.g., service requests and MIB access) and also allow the WME to notify higher layer entities.

7.4.2 Service requests

7.4.2.1 General

Service requests defined in this subclause provide access to management functions of Networking Services. In some cases, service request information is stored in the WME MIB for future access. In this case, a *Local Service Index* is used to distinguish one collection of request data from another. Conceptually, the *Local Service Index* is the index to the associated MIB table or internal data structure, but its form and use is implementation dependent.

7.4.2.2 WME-ProviderService.request

7.4.2.2.1 Function

This primitive indicates that a higher layer entity requests a WSA be transmitted.

7.4.2.2.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-ProviderService.request (
Local Service Index,
Action,
Destination MAC address (optional),
WSA Type,
ProviderServiceIdentifier,
ProviderServiceContext,
ServicePriority,
ServiceSpecificPermissions,
SecurityServiceIdentifier,

Channel Identifier (optional),
Channel Access,
Repeat Rate,
IP Service,
IPv6 Address (optional),
Service Port (optional),
Provider MAC address (optional),
RCPI Threshold (optional),
WSA Count Threshold (optional),
WSA Count Threshold Interval (optional),
WSA header extensions,
SignatureLifetime
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	This is an internal identifier of the information in the WME MIB.
Action	Enumerated	Add, Delete, Change	Indicates the requested action.
Destination MAC Address	MACAddress	Any valid MAC address	Destination of the WSA; default is the broadcast address.
WSA Type	Enumerated	Send unsecured WSA, Send secured WSA	Indicates requested security processing.
ProviderServiceIdentifier	Octet string	See 8.1.3	For insertion in WSA.
ProviderServiceContext	Octet string	0–31 octets	For insertion in WSA.
ServicePriority	Integer	0–63	Used in channel assignment determination.
ServiceSpecificPermissions	Octet string	0–255 octets	Used in security processing (see IEEE P1609.2).
SecurityServiceIdentifier	Octet string	16 octets	Used in security processing (see IEEE P1609.2).
Channel Identifier	See 7.2	See 7.2	Used in channel assignment. If no *Channel Identifier* is present, the WME may choose a suitable channel. In choosing a suitable channel, any relevant regulatory constraints should be considered.
Channel Access	Enumerated	Both SCH Interval and CCH Interval (continuous access), SCH Interval only (alternating access)	Defaults to SCH Interval only. Indicates the desired service channel access, for use by the scheduling algorithm.
Repeat Rate	Integer	0–255	The number of WSAs to be transmitted per 5 s. If *Destination MAC Address* is an individual address, *Repeat Rate* is ignored.
IP Service	Boolean	True, false	Indicates whether the advertised service is IP based, and a *WAVE Routing Advertisement* is needed to support this service.
IPv6 Address	Ipv6Address	Any valid unicast IPv6 address	For insertion in WSA, if present.
Service Port	Integer	$0–(2^{16} – 1)$	For insertion in WSA, if present.
Provider MAC address	MACAddress	Any valid individual MAC Address	For insertion in WSA, if present.
RCPI Threshold	Integer	As specified in Clause 8.	For insertion in WSA, if present.
WSA Count Threshold	Integer	As specified in Clause 8.	For insertion in WSA, if present.

WSA Count Threshold Interval	Integer	As specified in Clause 8.	For insertion in WSA, if present.
WSA header extensions	Bit string	Not specified	Indicates which of the WSA header extension fields specified in 8.2.2.4 should be included in the transmitted WSA.
SignatureLifetime	Integer	10–30 000 ms	The number of milliseconds over which the WSA signature should be valid (for secured WSAs)

7.4.2.2.3 When generated

The primitive is generated as needed by higher layer entities.

7.4.2.2.4 Effect of receipt

On receipt, the WME generates a WME-ProviderService.confirm indicating whether the request is accepted. On acceptance, the provider service request is considered for channel access assignment.

7.4.2.3 WME-ProviderService.confirm

7.4.2.3.1 Function

This primitive confirms the acceptance of the corresponding request.

7.4.2.3.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-ProviderService.confirm (
Local Service Index,
ResultCode
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	This is an internal identifier of the information in the WME MIB.
ResultCode	Enumerated	Accepted, Rejected invalid parameter, Rejected unspecified	Indicates the result of the associated request.

7.4.2.3.3 When generated

The primitive is generated in response to WME-ProviderService.request.

7.4.2.3.4 Effect of receipt

The receiving higher layer entity may take action based on the confirmation.

7.4.2.4 WME-UserService.request

7.4.2.4.1 Function

This primitive indicates that a higher layer entity requests a communication services.

7.4.2.4.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-UserService.request (
Local Service Index,
Action,
UserRequestType,
**ProviderServiceIdentifier,*
**ServicePriority,*
**WsaTypes,*
**ProviderServiceContext* (optional),
**Channel Identifier* (optional),
**Source MAC address* (optional),
**AdvertiserIdentifier* (optional),
**Link Quality* (optional),
ImmediateAccess (optional),
ExtendedAccess (optional)
)

* Indicates parameters that may be matched to available services.

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	This is an internal identifier of the information in the WME MIB.
Action	Enumeration	Add, Delete	Indicates whether the associated user service info should be added or removed from the MIB.
UserRequestType	Enumerated	Auto-access on service match, Auto-access unconditionally, No SCH access	Indicates the requested scheduler behavior. Auto-access on service match indicates the scheduler should provide SCH access when detecting an available service matching all the criteria indicated in the request. Auto-access unconditionally indicates the scheduler should provide SCH access regardless of available services. No SCH access indicates the scheduler should attempt to match the request with available services, but not automatically provide SCH access.
ProviderServiceIdentifier	Octet string	See 8.1.3	Identifies the PSID of the desired service.
ServicePriority	Integer	0–63	Used in channel assignment determination.
WsaTypes	Enumerated	Unsecured WSA, Secured WSA, Secured or Unsecured WSA, Any	Indicates what types of WSAs are acceptable to the requester. Unsecured WSA refers to a WSA received with no security credentials. Secured WSA refers to a WSA whose security credentials have been successfully verified. Any encompasses Secured WSA, Unsecured WSA, and also a WSA whose security verification has failed.

Name	Type	Valid range	Description
ProviderServiceContext	Octet string	0–31 octets	Identifies the PSC of the desired service.
Channel Identifier	See 7.2	See 7.2	Identifies the radio channel of the desired service.
Source MAC address	MACAddress	Any valid individual MAC Address	The address of the sender of the WSA.
Advertiser Identifier	Octet string	1–32 octets	The *Advertiser Identifier* string optionally sent in the WSA.
Link Quality	Variable	Unspecified	The quality of the RF link, below which the service will be ignored.
ImmediateAccess	Boolean	True, false	This indicates that the device should immediately visit the SCH, rather than waiting for the next SCH Interval.
ExtendedAccess	Integer	0–65 535	This indicates a request for continuous operation on the SCH for *ExtendedAccess* number of control channel intervals. A value of 0 indicates alternating access. A value of 65 535 indicates indefinite access.

7.4.2.4.3 When generated

The primitive is generated as needed by higher layer entities.

7.4.2.4.4 Effect of receipt

On receipt, the WME generates a WME-UserService.confirm indicating whether the request is accepted. On acceptance, the user service request is considered for channel access assignment.

7.4.2.5 WME-UserService.confirm

7.4.2.5.1 Function

This primitive confirms the acceptance of the corresponding request.

7.4.2.5.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-UserService.confirm (
Local Service Index,
ResultCode
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	This is an internal identifier of the information in the WME MIB.
ResultCode	Enumerated	Accepted, Rejected invalid parameter, Rejected unspecified	Indicates the result of the associated request.

7.4.2.5.3 When generated

The primitive is generated in response to WME-UserService.request.

7.4.2.5.4 Effect of receipt

The receiving higher layer entity may take action based on the confirmation.

7.4.2.6 WME-WSMService.request

7.4.2.6.1 Function

This primitive indicates that a higher layer entity requests a short message service.

7.4.2.6.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-WSMService.request (
Local Service Index,
Action,
ProviderServiceIdentifier,
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	This is an internal identifier of the information in the WME MIB.
Action	Enumeration	Add, Delete	Indicates the requested action.
ProviderServiceIdentifier	Octet string	See 8.1.3	Received WSMs with this PSID will be sent to the requesting higher layer entity.

7.4.2.6.3 When generated

The primitive is generated as needed by higher layer entities.

7.4.2.6.4 Effect of receipt

On receipt, the WME generates a WME-WSMService.confirm indicating whether the request is accepted. On acceptance, the received WAVE Short Messages will be sent to the requesting higher layer entity.

7.4.2.7 WME-WSMService.confirm

7.4.2.7.1 Function

This primitive confirms the acceptance of the corresponding request.

7.4.2.7.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-WSMService.confirm (
Local Service Index,
ResultCode
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	This is an internal identifier of the information in the WME MIB.
ResultCode	Enumeration	Accepted, Rejected invalid parameter, Rejected unspecified	Indicates the result of the associated request.

7.4.2.7.3 When generated

The primitive is generated in response to WME-WSMService.request.

7.4.2.7.4 Effect of receipt

The receiving higher layer entity may take action based on the confirmation.

7.4.2.8 WME-CchService.request

7.4.2.8.1 Function

This primitive indicates that a higher layer entity requests uninterrupted access to the control channel.

7.4.2.8.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-CchService.request (
Local Service Index,
Action,
ChannelInterval,
ServicePriority
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	This is an internal identifier of the information in the WME MIB.
Action	Enumerated	Add, Delete	Indicates the requested action.
Channel Interval	Enumeration	CCH Interval, SCH Interval, both	The channel interval to which the request applies.
ServicePriority	Integer	0–63	Used in channel assignment determination.

7.4.2.8.3 When generated

The primitive is generated as needed by higher layer entities.

7.4.2.8.4 Effect of receipt

On receipt, the WME generates a WME-CchService.confirm indicating whether the request is accepted. On acceptance, the CCH service request is considered for channel access assignment.

7.4.2.9 WME-CchService.confirm

7.4.2.9.1 Function

This primitive confirms the acceptance of the corresponding request.

7.4.2.9.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-CchService.confirm (
Local Service Index,
ResultCode
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	This is an internal identifier of the information in the WME MIB.
ResultCode	Enumeration	Accepted, Rejected invalid parameter, Rejected unspecified	Indicates the result of the associated request.

7.4.2.9.3 When generated

The primitive is generated in response to WME-CchService.request.

7.4.2.9.4 Effect of receipt

The receiving higher layer entity may take action based on the confirmation.

7.4.2.10 WME-ManagementDataService.request

7.4.2.10.1 Function

This primitive indicates that an external management entity requests a Vendor Specific Action frame be transmitted.

7.4.2.10.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-ManagementDataService.request (
Local Service Index,
Action,
Destination MAC address,
Management ID,
Organization Identifier,
Data,
Repeat Rate,
Channel Identifier,
Channel Interval,
ServicePriority
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	An internal identifier of the information related to the request.
Action	Enumerated	Add, Delete, Change	Indicates the requested action.
Destination MAC address	MACAddress	Any valid address	The intended destination of the VSA.
Management ID	Enumerated	0–15	Identifies the source of the data when the source is a 1609 entity. See 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 a 1609 entity. See IEEE Std 802.11p.
Data	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 seconds. 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.
ServicePriority	Integer	0–63	Used in channel assignment determination.

7.4.2.10.3 When generated

The primitive is generated as needed by higher layer entities.

7.4.2.10.4 Effect of receipt

On receipt, the WME generates a WME-ManagementDataService.confirm indicating whether the request is accepted. On acceptance, the VSA request is considered for channel access assignment.

7.4.2.11 WME-ManagementDataService.confirm

7.4.2.11.1 Function

This primitive confirms the acceptance of the corresponding request.

7.4.2.11.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-ManagementDataService.confirm (
Local Service Index,
ResultCode
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	An internal identifier of the information related to the request.
ResultCode	Enumerated	Accepted, Rejected invalid parameter, Rejected unspecified	Indicates the result of the associated request.

7.4.2.11.3 When generated

The primitive is generated in response to WME-ManagementDataService.request.

7.4.2.11.4 Effect of receipt

The receiving higher layer entity may take action based on the confirmation.

7.4.2.12 WME-ManagementDataService.indication

7.4.2.12.1 Function

The primitive indicates that a Vendor Specific Action frame has been received.

7.4.2.12.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-ManagementDataService.indication (
Source MAC Address,
Management ID,
Data,
Channel Identifier,
RCPI
)

Name	Type	Valid range	Description
Source MAC Address	MACAddress	Any valid unicast MAC Address	Indicates the source of the VSA frame
Management ID	Integer	0–15	An identifier of the source of the data. See IEEE P1609.0.
Data	Octet string	Unspecified	Information received as vendor specific content.
Channel Identifier	See 7.2	See 7.2	Identifies the channel on which the data was received.
RCPI	As specified in IEEE Std 802.11	As specified in IEEE Std 802.11	A measure of the received channel power for the VSA.

7.4.2.12.3 When generated

The primitive is generated by the WME to deliver data received in a VSA.

7.4.2.12.4 Effect of receipt

The data is processed as determined by the recipient.

7.4.2.13 WME-TimingAdvertisementService.request

7.4.2.13.1 Function

This primitive indicates that a higher layer entity requests a Timing Advertisement frame be transmitted.

7.4.2.13.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-TimingAdvertisementService.request (
Local Service Index,
Action,
Destination MAC address,
Repeat Rate,
Channel Identifier,
Channel Interval,
ServicePriority,
TimingAdvertisementContents
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	An internal identifier of the information related to the request.
Action	Enumerated	Add, Delete, Change	Indicates the requested action.
Destination MAC address	MAC Address	Any valid address	The intended destination of the TA.
Repeat Rate	Integer	0–255	The number of Timing Advertisement frames to be transmitted per 5 s. If *Destination MAC Address* is an individual address, *Repeat Rate* is ignored.

Name	Type	Valid range	Description
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.
ServicePriority	Integer	0–63	Used in channel assignment determination.
TimingAdvertisementContents	As described in the subsequent discussion	As described in the subsequent discussion	As described in the subsequent discussion.

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.*

7.4.2.13.3 When generated

The primitive is generated as needed by management entities.

7.4.2.13.4 Effect of receipt

On receipt, the WME generates a WME-TimingAdvertisementService.confirm indicating whether the request is accepted. On acceptance, the TA request is considered for channel access assignment.

7.4.2.14 WME-TimingAdvertisementService.confirm

7.4.2.14.1 Function

This primitive confirms the acceptance of the corresponding request.

7.4.2.14.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WME-TimingAdvertisementService.confirm (
Local Service Index,
ResultCode
)

Name	Type	Valid range	Description
Local Service Index	Integer	0–65 535	An internal identifier of the information related to the request.
ResultCode	Enumerated	Accepted, Rejected invalid parameter, Rejected unspecified	Indicates the result of the associated request.

7.4.2.14.3 When generated

The primitive is generated in response to WME-TimingAdvertisementService.request.

7.4.2.14.4 Effect of receipt

The receiving management entity may take action based on the confirmation.

7.4.3 Notification

7.4.3.1 WME-Notification.indication

7.4.3.1.1 Function

The WME-Notification.indication primitive indicates that an event has occurred. Implementers may expand the information contents as appropriate for a particular design.

7.4.3.1.2 Semantics of the service primitive

The parameters of the WME-Notification.indication primitive are as follows:

WME-Notification.indication (
Event,
Local Service Index,
Reason
)

Name	Type	Valid range	Description
Event	Enumeration	Channel assigned, No channel assigned, Request matched with available service	Identifies the type of event that triggered the notification.
Local Service Index	Integer	0–65 535	An internal identifier of the information related to a request.
Reason	Enumeration	Unspecified, Requested, ChannelUnavailable, ServiceComplete, Requested frame scheduled, PriorityPreemption, SecurityCredentialFailure	Provides further information about the event that triggered the notification.

7.4.3.1.3 When generated

The WME-Notification.indication primitive is sent when one of the designated events occurs.

7.4.3.1.4 Effect of receipt

The notification is processed as determined by the receiving higher layer entity.

7.4.4 MIB access

7.4.4.1 WME-Get.request

7.4.4.1.1 Function

The WME-Get.request primitive is generated by a higher layer entity to retrieve the value of a specific WME MIB attribute.

7.4.4.1.2 Semantics of the service primitive

The parameters of the WME-Get.request primitive are as follows:

```
WME-Get.request (
                    MIBattribute
                    )
```

7.4.4.1.3 When generated

The WME-Get.request primitive is generated by a higher layer entity to request the value of the specified *MIBattribute*.

7.4.4.1.4 Effect of receipt

Upon receipt of the WME-Get.request primitive, the WME returns the MIBattribute value from the WME MIB.

7.4.4.2 WME-Get.confirm

7.4.4.2.1 Function

The WME-Get.confirm primitive returns the result of the WME-Get.request.

7.4.4.2.2 Semantics of the service primitive

The parameters of the WME-Get.confirm primitive are as follows:

```
WME-Get.confirm (
                    Status,
                    MIBattribute,
                    MIBattributevalue
                    )
```

7.4.4.2.3 When generated

The WME-Get.confirm primitive is used to confirm the result of the WME-Get.request.

7.4.4.2.4 Effect of receipt

The WME-Get.confirm primitive returns the appropriate WME MIB attribute value if Status equals success; otherwise, it returns an error indication in the Status field. Possible error status values include "invalid MIB attribute" and "attempt to get write-only MIB attribute."

7.4.4.3 WME-Set.request

7.4.4.3.1 Function

The WME-Set.request primitive is generated by higher layer entities to set the value of a specific WME MIB attribute.

7.4.4.3.2 Semantics of the service primitive

The parameters of the WME-Set.request primitive are as follows:

WME-Set.request (
MIBattribute,
MIBattributevalue
)

7.4.4.3.3 When generated

The WME-Set.request primitive is generated by higher layer entities to request that the value of the specified MIBattribute be set to the indicated value. The MIB contains sensitive data; implementers are responsible for protecting sensitive data, although this standard does not specify the means for doing so.

7.4.4.3.4 Effect of receipt

Upon receipt of the WME-Set.request primitive, the WME returns WME-Set.confirm, and, if it is able, sets the MIB attribute value in the WME MIB.

7.4.4.4 WME-Set.confirm

7.4.4.4.1 Function

The WME-Set.confirm primitive returns the result of the WME-Set.request.

7.4.4.4.2 Semantics of the service primitive

The parameters of the WME-Set.confirm primitive are as follows:

WME-Set.confirm (
Status,
MIBattribute
)

7.4.4.4.3 When generated

The WME-Set.confirm primitive is used to confirm the result of the WME-Set.request.

7.4.4.4.4 Effect of receipt

If *Status* equals Success, this confirms that the indicated MIB attribute was set to the requested value; otherwise, it returns an error condition in *Status* parameter. If this *MIBattribute* implies a specific action, then this confirms that the action was performed. Possible error *Status* values include "invalid MIB attribute" and "attempt to set read-only MIB attribute."

7.4.5 Address change

7.4.5.1 WME-AddressChange.request

7.4.5.1.1 Function

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

7.4.5.1.2 Semantics of the service primitive

The parameters of the primitive are as follows. If no address is included the device will generate a new link local address.

WME-AddressChange.request (
Link Local Address (optional)
)

Name	Type	Valid range	Description
Link Local Address	IPv6 Address	Any	If present, the new link local IP address to be adopted by the device.

7.4.5.1.3 When generated

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

7.4.5.1.4 Effect of receipt

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

7.4.5.2 WME-AddressChange.confirm

7.4.5.2.1 Function

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

7.4.5.2.2 Semantics of the service primitive

The parameters of the primitive are as follows:

```
WME-AddressChange.confirm (
                    ResultCode
                    )
```

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

7.4.5.2.3 When generated

The primitive is generated in response to WME-AddressChange.request.

7.4.5.2.4 Effect of receipt

No effect is specified.

7.5 WAVE LSAP

7.5.1 General

LLC primitives, identified by the "DL-" prefix, pass data packets into and out of the LLC layer and are specified in 7.5.2 and in ISO/IEC 8802-2 [IEEE Std 802.2].

The WAVE LLC SAP is identical to the IEEE 802.2 LLC SAP with the exception of parameters added to the DL-UNITDATA.request. These parameters are included in the DL-UNITDATAX.request specified in 7.5.2 and allow higher layers to control the transmission characteristics of WSMP data.

7.5.2 DL-UNITDATAX.request

7.5.2.1 Function

This primitive is the service request primitive for the unacknowledged connectionless-mode data transfer service. The DL-UNITDATA.request specified in ISO/IEC 8802-2 [IEEE Std 802.2] is extended here by the addition of *Channel Identifier*, *DataRate*, *TxPwr_Level*, and *WsmExpiryTime*, which are used in the case of WAVE Short Message transmissions.

7.5.2.2 Semantics of the service primitive

The parameters of the primitive are as follows:

```
DL-UNITDATAX.request (
                    source_address,
                    destination_address,
                    data,
```

priority,
Channel Identifier,
DataRate,
TxPwr_Level,
WsmExpiryTime
)

Name	Description
source_address	As specified in ISO/IEC 8802-2 [IEEE Std 802.2]
destination_address	As specified in ISO/IEC 8802-2 [IEEE Std 802.2]
Data	As specified in ISO/IEC 8802-2 [IEEE Std 802.2]
Priority	*User priority*, as specified in ISO/IEC 8802-2 [IEEE Std 802.2]
Channel Identifier	As specified in 7.2 and 7.3.2. Used for WSMP traffic.
DataRate	As specified in 7.3.2. Used for WSMP traffic.
TxPwr_Level	As specified in 5.5.2. Used for WSMP traffic.
WsmExpiryTime	As specified in 7.3.2. Optionally used for WSMP traffic.

Channel Identifier, *DataRate*, *TxPwr_Level*, and *WsmExpiryTime* are passed through the WAVE LLC to the MAC layer. Otherwise the DL-UNITDATAX.request is processed as the DL-UNITDATA.request specified in ISO/IEC 8802-2 [IEEE Std 802.2].

7.6 MLME and MLMEX SAPs

The MLME and MLMEX primitives allow communication between the WME and the MAC sublayer management entity as specified in the documents referenced in Table 2. The WSA used in certain primitives is specified in Clause 8.

7.7 Security SAP

The security primitives allow communication between the WME and the security entity as specified in IEEE P1609.2.

8. WAVE information formats

8.1 General

This clause specifies the formats of information sent over the air and originating within Networking Services. Subclause 8.2 specifies the format of the WAVE Service Advertisement; 8.3 specifies the format of the WAVE Short Message. When a message is supported by an implementation, the formats specified in this clause are mandatory. Example message contents are found in Annex G.

Over-the-air formats shall adhere to the following conventions, compatible with IEEE Std 802.11 and IEEE Std 802.11p.

The information is described as a sequence of fields in specific order. The figures depict the fields/subfields as they appear in the MAC frame, from left to right.

Using the convention from IEEE Std 802.11, bits within fields are numbered from 0 to *k*, where the length of the field is *k* + 1 bits. Whenever a multioctet field represents a numeric quantity, the highest numbered

bit of the whole field is the most significant bit. When a multioctet quantity is transmitted, the most significant octet (i.e., the octet containing the most significant bit) is transmitted first.

An example is shown as follows, where the decimal number 1 is represented by a 16-bit number. This would appear in the WSA or WSM as hexadecimal value 0x0001.

	More significant octet								**Less significant octet**							
Bit number	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Bit value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

MAC addresses are assigned as ordered sequences of bits. The *Individual/Group* bit is always transferred first and is bit 0 of the first octet. A field containing a PSID is transmitted as specified in 8.1.3.

Values specified in decimal are coded in natural binary unless otherwise stated.

Reserved fields and subfields are set to 0 upon transmission and are ignored upon reception.

8.1.1 Extension fields and WAVE Element ID

The WSM header and several segments of the WSA allow for the presence of extension fields. Subclauses 8.1.2 and 8.1.3 specify the content of these extension fields. Extension fields may be sent in any order relative to each other. Each extension field shall have the structure shown in Figure 15, where field lengths are shown in octets. The *Length* field indicates the length in octets of the *Contents*. The *Contents* of extension fields are defined in 8.1.2 and 8.1.3; *WAVE Element ID* values are specified in Annex E. Extension fields are only allowed in the message components specified in Annex E (e.g., *Transmit Power Used* is only permitted in WSA header and WSMP header). Identifiers used to identify message components that are not extension fields (i.e., WSA WAVE elements and WSMP WAVE elements), are stand alone, and do not have the structure shown in Figure 15. See 8.2.1 and 8.3.1.

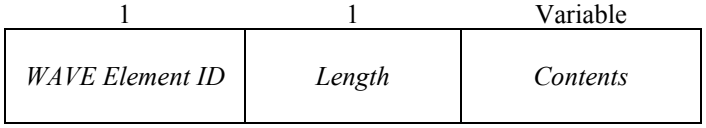


Figure 15—Extension fields

A WAVE device shall be capable of accepting and processing messages with unrecognized *WAVE Element IDs*. Unless otherwise specified, extension fields are optionally included in the transmitted message and optionally processed on receipt.

8.1.2 Management ID and Content Descriptor

The *Management ID* is used by WAVE in the IEEE 802.11 Vendor Specific Action frame and *Vendor Specific Information Element* to indicate the management entity sourcing the management information. *Management ID* is passed across the MLMEX SAP via primitives specified in IEEE Std 1609.4. See IEEE P1609.0 for specification of *Management ID* values. For the case of WSAs, *Management ID* has a value of 3 indicating "1609.3." In this case (i.e., for 1609.3 management messages), the first octet of the associated vendor-specific content is an 8-bit *Content Descriptor,* which is coded as shown in Table 3.

Table 3—*Content Descriptor*

Value	Meaning
1	WAVE Service Advertisement
	Other values reserved

8.1.3 Provider Service Identifier

PSID is specified as a variable-length ordered sequence of octets and may be expressed in hexadecimal representation (see 3.1). The octets of the PSID shall always be transmitted in sequential order, starting from the leftmost octet in hexadecimal representation. Octets are numbered starting at zero (Octet 0) and numbering increments in order of transmission. The length of the PSID is deterministic based on Octet 0, where the position of the first zero-value bit in descending order of bit significance in the octet shall specify the length in octets of the PSID as defined in Table 4. PSID examples are shown in Annex H.

Table 4—PSID

Most significant bits of Octet 0 (x indicates "don't care")				PSID length (octets)	PSID range (hexadecimal representation)
b7	b6	b5	b4		
0	x	x	x	1	00 to 7F
1	0	x	x	2	80-00 to BF-FF
1	1	0	x	3	C0-00-00 to DF-FF-FF
1	1	1	0	4	E0-00-00-00 to EF-FF-FF-FF
1	1	1	1	Reserved for lengths ≥ 5	Reserved

NOTE—The PSID format specified in this subclause is compliant with and can be expressed using Abstract Syntax Notation One (ASN.1) notation and unaligned packed encoding rules, as used by other ITS standards. The PSID is specified explicitly to be consistent with the conventions used in this document.[8]

8.2 WSA format

8.2.1 General

WSA information passed over the air shall be formatted as a *1609Dot2Message* specified in IEEE P1609.2 and illustrated in Figure 16 and Figure 17, and it is described in 8.2. A WSA may be sent within either an *Unsecured WSA* or a *Secured WSA*. An *Unsecured WSA* consists of a *WaveServiceAdvertisement* with *protocol_version* and *type* fields as specified in IEEE P1609.2. *Type* equals unsecured.

A *Secured WSA* consists of the *WaveServiceAdvertisement*, plus *protocol_version* and *type* fields and security header and trailer as shown in Figure 16. The security header and security trailer are specified in IEEE P1609.2 and generated via the WaveSecurityServices-SignedWsa.request primitive. *Type* equals signed_wsa.

[8] Notes in text, tables, and figures of a standard are given for information only and do not contain requirements needed to implement this standard.

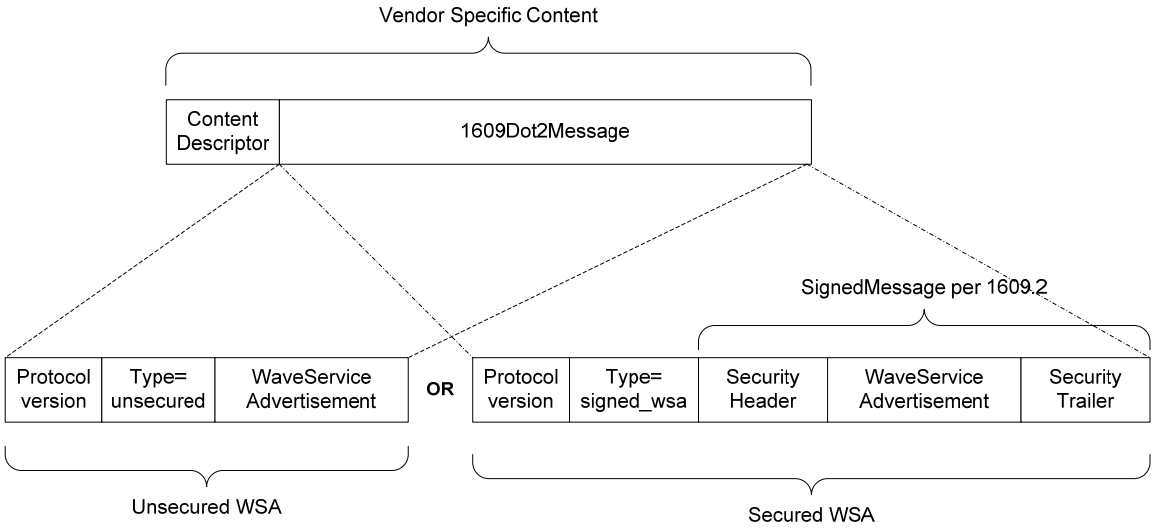


Figure 16—WSA format

As depicted in Figure 17 and described in the following discussion, the *WaveServiceAdvertisement* includes header information and may include a series of variable-length *Service Info* segments, a series of variable-length *Channel Info* segments, and a *WAVE Routing Advertisement*. *Service Info*, *Channel Info*, and *WAVE Routing Advertisement* are optionally included in any given WSA. Networking Services implementing the user role shall recognize as valid any received WSA that conforms to this standard.

In Figure 17, optional fields are indicated by stripes. Parameter lengths shown in the illustrations are in octets unless otherwise noted. Field ordering shall not be changed.

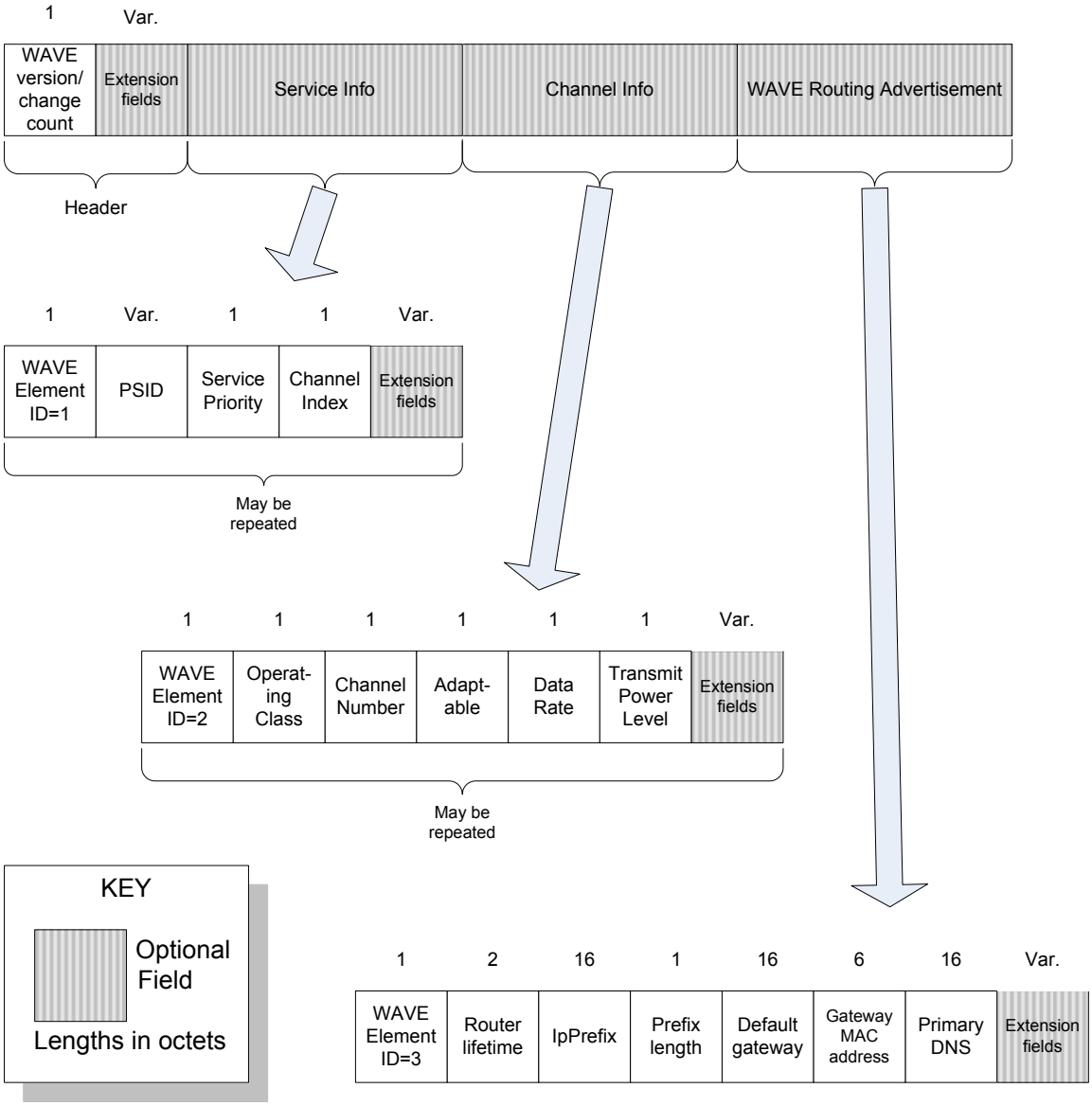


Figure 17—WaveServiceAdvertisement format

8.2.2 Header

8.2.2.1 General

The WSA header segment contains general information about the transmitting WAVE device. Its contents are specified in 8.2.2.2 through 8.2.2.4.

The mandatory portion of the WSA Header is composed of two fields sharing one octet as shown in Figure 18, with the fields specified in the subsequent discussion.

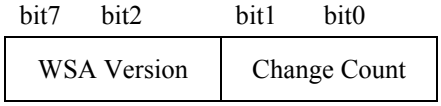


Figure 18—WSA Header mandatory fields

8.2.2.2 WAVE Version

For this standard, the value of the *WAVE Version* is 1. All other values are reserved. The revision level will be incremented only when a fundamental incompatibility exists between a new revision and the prior edition of the standard. A device that receives a WSA with an unsupported revision level shall discard the frame.

8.2.2.3 Change Count

The *Change Count* may be used by the recipient to determine whether a WSA is a repeat of the previous one from the same source. The sender shall increment its value modulo-4 each time the WSA (including the *WAVEServiceAdvertisement* contents or security information) changes.

8.2.2.4 WSA header extension fields

8.2.2.4.1 General

Extension fields shall not cause the WSA header to exceed 255 octets.

8.2.2.4.2 Repeat Rate

This 8-bit field indicates the number of times the advertisement is transmitted per 5 s. It may be used by recipients in evaluating link quality. *Repeat Rate* is not meant to be used with unicast announcements.

8.2.2.4.3 Transmit Power Used

This 8-bit field indicates the power at which the WSA was transmitted, as specified in IEEE Std 802.11k-2008.

8.2.2.4.4 2DLocation

This field provides the location of the WSA transmit antenna and is encoded as specified in Figure 19 and the text that follows as derived from SAE J2735.

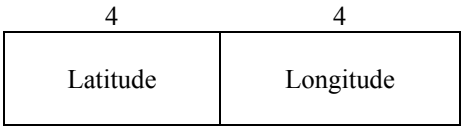


Figure 19—2DLocation

Latitude has an LSB representing 1/10 micro degree, representing a range from –90° to +90°, with value 900000001 indicating unavailable.

Longitude has an LSB representing 1/10 micro degree, representing a range from –180° to +180°, with value 1800000001 indicating unavailable.

8.2.2.4.5 3DLocationAndConfidence

This field provides the location of the WSA transmit antenna and is encoded as specified in Figure 20 and the text that follows as derived from SAE J2735.

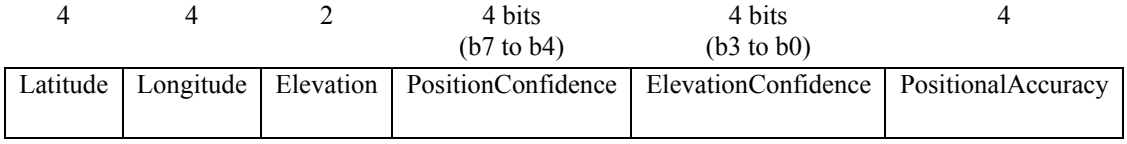


Figure 20—3DLocationAndConfidence

Latitude is encoded as in *2DLocation*.

Longitude is encoded as in *2DLocation*.

Elevation represents the geographic position above or below the reference ellipsoid (typically WGS-84). The 16-bit number has a resolution of 1 m and represents an asymmetric range of positive and negative values. The encoding is as follows: the range 0x0000 to 0xEFFF (0 to 61439 decimal) are positive numbers representing elevations from 0 to +6143.9 m (i.e., above the reference ellipsoid). The range 0xF001 to 0xFFFF includes negative numbers representing elevations from –409.5 m to –0.1 m (i.e., below the reference ellipsoid). An elevation higher than +6143.9 m is represented 0xEFFF. An elevation lower than –409.5 m is represented 0xF001. If the sending device does not know its elevation, it shall encode the *Elevation* data element with 0xF000.

Examples of this encoding are as follows: The elevation 0 m is encoded as 0x0000, the elevation –0.1 m is encoded as 0xFFFF, and the elevation +100.0 m is encoded as 0x03E8.

PositionConfidence is used to provide the confidence interval of the 95% confidence level for the currently reported location, taking into account the current calibration and precision of the sensor(s) used to measure and/or calculate the value. It is used in the horizontal plane. See SAE J2735 for details.

ElevationConfidence is used to provide the confidence interval of the 95% confidence level for the currently reported elevation, taking into account the current calibration and precision of the sensor(s) used to measure and/or calculate the value. See SAE J2735 for details.

PositionalAccuracy consists of three parameters of quality used to model the accuracy of the positional determination with respect to each given axis: semimajor accuracy (1 octet), semiminor accuracy (1 octet), and orientation of the semimajor axis (2 octets). See SAE J2735 for details.

8.2.2.4.6 Advertiser Identifier

The *Advertiser Identifier* has a length from 1 to 32 octets and carries a text string associated with the advertising device.

8.2.2.4.7 Country String

If sent, this 3-octet field is set to the value of *dot11CountryString*, as specified in IEEE Std 802.11. It indicates the regulatory domain in which the system is operating.

8.2.3 Service Info

8.2.3.1 General

There shall be 0 through 32 instances of *Service Info* segments in the WSA. Each provides a definition of one service and contains the fields listed in 8.2.3.2 through 8.2.3.6.

8.2.3.2 Service Info WSA WAVE Element ID

This indicates the beginning of an instance of *Service Info*. The *WAVE Element ID* value for *Service Info* is specified in Annex E.

8.2.3.3 Provider Service Identifier

Provider Service Identifier is specified in 8.1.3 and identifies the service being offered.

8.2.3.4 ServicePriority

ServicePriority is an 8-bit field ranging in value from 0 to 63, where 0 indicates the lowest priority and 63 is the highest. It indicates the priority as determined by the higher layer entity advertising the service and is used by the provider Networking Services in prioritizing channel access. It may also be used by the recipient in evaluating the service or in prioritizing channel access.

8.2.3.5 Channel Index

Channel Index provides a pointer to the *n*th set of channel parameters within *Channel Info* in the WSA and indicates the service channel where the advertised service is being offered. *Channel Index* of 1 indicates the first set of parameters, *Channel Index* of 2 indicates the second set of parameters, etc., through *Channel Index* of 32. Other *Channel Index* values are reserved.

8.2.3.6 Service Info extension fields

Extension fields shall not cause the *Service Info* to exceed 255 octets.

8.2.3.6.1 Provider Service Context

The PSC provides supplementary information related to the service with which it is associated as defined by the higher layer. The PSC is an octet string of length 1 to 31.

8.2.3.6.2 IPv6 Address

The *IPv6 Address* is the 128-bit IPv6 address of the device hosting the service and is formatted per IETF RFC 3513. This is present when the service employs IP addressing.

8.2.3.6.3 Service Port

The *Service Port* is the 16-bit port number of the higher layer entity providing the service and takes values from 0 to 65 535. It is present when the service employs IP addressing.

8.2.3.6.4 Provider MAC Address

The *Provider MAC Address* is the 48-bit MAC address of the device hosting the service. This is present if different from the MAC address of the device transmitting the WSA.

8.2.3.6.5 RCPI Threshold

If present, the *RCPI Threshold* indicates the recommended minimum received WSA signal value in dBm, 0 to –110, below which the service should be ignored by a recipient. It has a length of 1 octet, coded as *RCPI* in IEEE Std 802.11k-2008.

8.2.3.6.6 WSA Count Threshold

If present, the *WSA Count Threshold* indicates the recommended minimum number of received WSAs, below which a service should be ignored by a recipient. It has a length of 1 octet, with range 0 to 255.

8.2.3.6.7 WSA Count Threshold Interval

Optionally used with the *WSA Count Threshold*, it indicates the time interval over which received WSAs are counted. It has a length of 1 octet, with range 1 to 255, in units of 100 ms. The default value used if no *WSA Count Threshold Interval* is included in the WSA is 1 s.

8.2.4 Channel Info

There shall be 0 to 32 instances of *Channel Info* segments in the WSA. Each indicates the characteristics of one channel associated with zero or more *Service Info* and contains the fields described within this subclause.

8.2.4.1 Channel Info WSA WAVE Element ID

This indicates the beginning of an instance of *Channel Info*. The *WAVE Element ID* value for *Channel Info* is specified in Annex E.

8.2.4.2 Operating Class

Operating Class is one octet as specified in IEEE Std 802.11, IEEE Std 802.11p, and IEEE P802.11REVmb, and it allows the following *Channel Number* to identify a specific channel uniquely in the context of a country.

8.2.4.3 Channel Number

Channel Number indicates the number of the channel to which the accompanying information pertains. An identical *Operating Class/Channel Number* pair shall not appear in two instances of *Channel Info* in the same WSA. The *Channel Number* is 8 bits in length and coded as specified in IEEE Std 802.11.

8.2.4.4 Adaptable

Adaptable is 8 bits long and indicates whether *DataRate* and *Transmit Power Level* are boundary values or fixed values. A value of 1 indicates *DataRate* is interpreted as the minimum rate allowed and *Transmit Power Level* as the maximum level allowed. A value 0 indicates that *DataRate* and *Transmit Power Level* are interpreted as fixed values.

8.2.4.5 DataRate

DataRate indicates the data rate used on the channel. Per IEEE Std 802.11, *DataRate* is represented by a count from 0x02 through 0x7F, corresponding to data rates in increments of 500 kbit/s from 1 Mb/s to 63.5 Mb/s subject to limitations of each PHY. If *Adaptable* is set, *DataRate* is interpreted as the minimum rate allowed, and any higher rate is also allowed.

8.2.4.6 Transmit Power Level

Transmit Power Level is a signed number and one octet in length. It indicates the actual power, in dBm, to be used for transmissions on the associated channel as measured at the output of the antenna connector. If *Adaptable* is set, *Transmit Power Level* is interpreted as the maximum power allowed, and any lower power is also allowed.

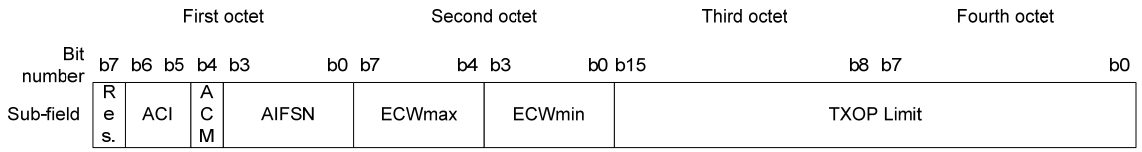
8.2.4.7 Channel Info extension fields

8.2.4.7.1 General

Extension fields shall not cause the *Channel Info* to exceed 255 octets.

8.2.4.7.2 EDCA Parameter Set

If present, this indicates the MAC-layer channel access parameters to be used by the various devices communicating on the channel. The format of the *EDCA Parameter Set* is shown in the subsequent figure, where each subfield is coded as specified in IEEE Std 802.11.



8.2.4.7.3 Channel Access

If present, this indicates the times, in terms of channel interval, during which the provider is on the associated SCH. Its length is one octet.

Table 5—Channel Access

Value	Meaning
0	Both SCH Interval and CCH Interval (continuous access)
1	SCH Interval only (alternating access)
	Other values reserved

8.2.5 WAVE Routing Advertisement

8.2.5.1 General

The *WAVE Routing Advertisement* segment of the WSA, if present, provides information about infrastructure internetwork connectivity, allowing receiving devices to be configured to participate on the advertised IPv6 network. As depicted in Figure 17, the *WAVE Routing Advertisement* contains the fields specified following. If the *WAVE Routing Advertisement* is present, all fields are mandatory unless indicated otherwise. Note that no more than one WRA is present in a WSA.

8.2.5.2 WRA WSA WAVE Element ID

This indicates the beginning of a *WAVE Routing Advertisement*. The *WAVE Element ID* value for WRA is as specified in Annex E. If there is no WRA in the WSA, neither the *WAVE Element ID* nor any of the following fields are present.

8.2.5.3 Router Lifetime

Router Lifetime is 16 bits and indicates the duration for which the *Default Gateway* and associated information is valid. It is coded and interpreted as specified in IETF RFC 2461.

8.2.5.4 IpPrefix

IpPrefix is 128 bits and indicates the IPv6 subnet prefix of the link, as described in IETF RFC 3513.

8.2.5.5 Prefix Length

Prefix Length is 8 bits and indicates how many of the higher order bits of *IpPrefix* are significant, as described in IETF RFC 3513.

8.2.5.6 Default Gateway

Default Gateway is the 128-bit IPv6 address of a router that provides internetwork connectivity to the subnet.

8.2.5.7 Primary domain name system (DNS)

Primary DNS is the 128-bit IPv6 address of a device that can provide DNS lookup for the subnet devices.

8.2.5.8 WRA extension fields

8.2.5.8.1 General

Extension fields shall not cause the WRA to exceed 255 octets.

8.2.5.8.2 Secondary DNS

Secondary DNS is the 128-bit IPv6 address of an alternative device that can provide DNS lookup for the subnet devices.

8.2.5.8.3 Gateway MAC Address

Gateway MAC Address is the 48-bit MAC address associated with the *Default Gateway*, if different than the MAC address of the device transmitting the WSA.

8.3 WSM format

8.3.1 General

Figure 21 illustrates how headers are constructed for a WSM as the data is passed from the source higher layer entity through Networking Services to the MAC. The fields with dotted outlines are passed as separate parameters in the primitive, not as part of the data package. The solid outlines indicate fields formatted for over-the-air transfer. The striped fields are optional. The received WSM information is passed up through the same set of layers, with each header stripped and processed by the appropriate protocol.

The WSMP header has the form as illustrated in Figure 21 and specified in 8.3.2 through 8.3.7. The field lengths are in octets unless otherwise indicated. The header extension fields may be included as specified in 8.1.1 and 8.3.4. A WAVE *Element ID* with a value indicating an Extension WAVE Element (Annex E) indicates an optional extension field (see 8.1.1). A WAVE Element ID value with a value indicating a WSMP WAVE Element (see Annex F) is followed by the *WSMLength* and the *WSM Data*. Additional WSM encoding conventions are specified in 8.1.

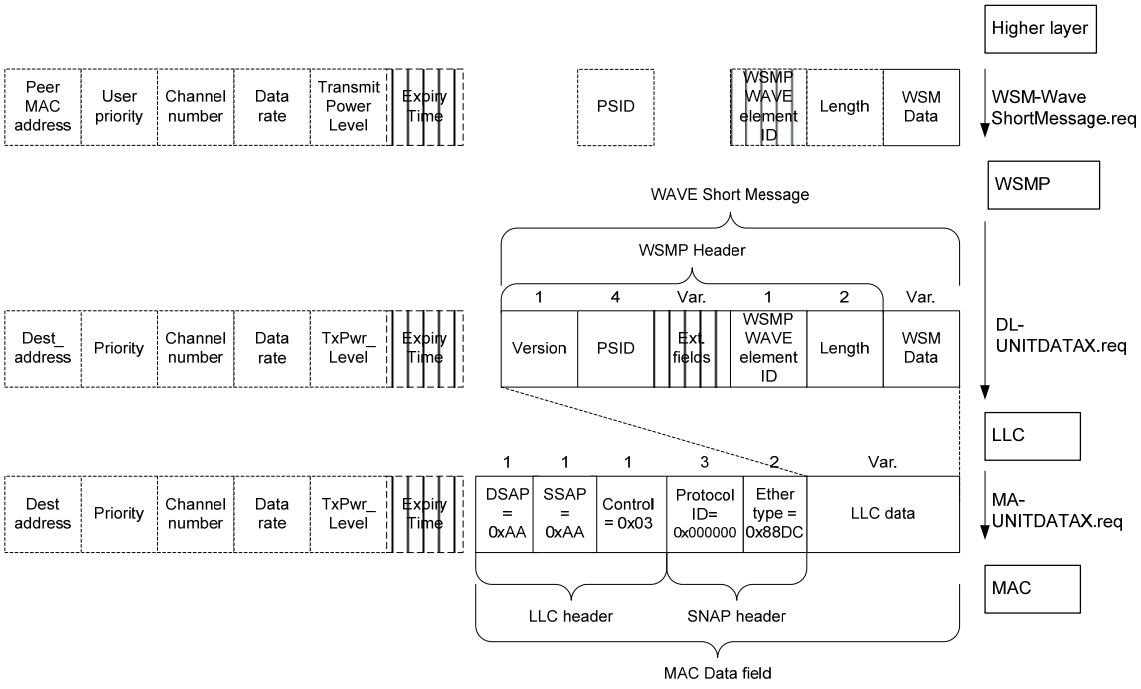


Figure 21—Building the WSM package

8.3.2 Version

The *Version* includes four bits reserved for future use in WSMP and a 4-bit *WsmpVersion* field as illustrated in Figure 22. The *WsmpVersion* indicates the version of the WSM protocol. For implementations conformant to this standard, the value of the *WsmpVersion* is 2. All other values are reserved. The revision level should be incremented only when a fundamental incompatibility exists between a new revision and the prior edition of the standard. A device that receives a WSMP packet with an unsupported revision level should discard the packet.

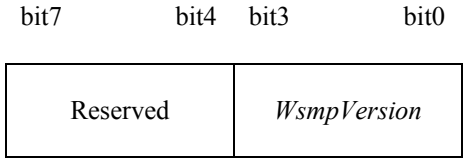


Figure 22—Version

8.3.3 Provider Service Identifier

The *Provider Service Identifier* value is used to determine the appropriate higher layer destination of the WSM. It is specified in 8.1.3.

8.3.4 WSMP header extension fields

8.3.4.1 General

Extension fields may be included in the WSMP header as specified in 8.3.4.2 through 8.3.4.4.

8.3.4.2 Channel Number

The *Channel Number* is optionally included in the WSMP header for use by the WSM recipient. The *Channel Number* is specified in IEEE Std 802.11.

8.3.4.3 DataRate

The *DataRate* is optionally included in the WSMP header for use by the WSM recipient. It is encoded as specified in 8.2.4.5.

8.3.4.4 Transmit Power Used

The *Transmit Power Used*, as specified in IEEE Std 802.11k-2008, is optionally included in the WSMP header for use by the WSM recipient.

8.3.5 WSMP WAVE Element ID

This field indicates the type of WAVE Short Message that follows. It is coded as specified in Annex E. Values of 128 and higher are designated for use in WSMP.

8.3.6 Length

The *Length* includes four bits reserved for future use in WSMP and a 12-bit *WSMLength* field as illustrated in Figure 23. The *WSMLength* indicates the length in octets of the following *WSMData* field.

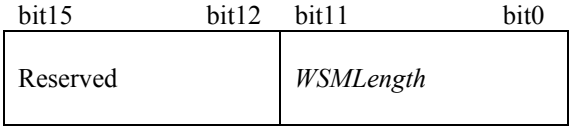


Figure 23—Length

8.3.7 WSMData

The *WSMData* contains the higher layer information being transferred. The *WSMData* is an octet string of length *WSMLength*.

Annex A

(informative)

WME MIB table

Table A.1 provides a summary of the contents of the WME MIB specified in Annex B. 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. These tools may be accessed online using the following URL: http://www.ibr.cs.tu-bs.de/bin/smitools.cgi. Using these tools, the MIB compiled without generating warnings of severity 4 or lower.

Table A.1—WME MIB contents

MIB segment	MIB item	Table entry	Contents	Type
General	*LocalInfo*		*NumberOfChannelsSupported, AdvertiserIdentifier, RegistrationPort, WsmMaxLength*	Control
General	*CchServiceRequestTable*	SEQUENCE OF *CchServiceRequest TableEntry*	*CchServiceRequestTableIndex, CchServiceRequestChannelInterval, CchServiceRequestPriority CchServiceRequestStatus*	Status
General	*WsmServiceRequestTable*	SEQUENCE OF *WsmServiceRequest TableEntry*	*WsmServiceRequestTableIndex, WsmServiceRequestPsid*	Status
Provider	*ProviderService RequestTable*	SEQUENCE OF *ProviderService RequestTableEntry*	*ProviderServiceRequestTableIndex, WsaType, ProviderServiceIdentifier, ProviderServiceContext, ProviderServiceRequestPriority, ProviderServiceSpecificPermissions, ProviderSecurityServiceIdentifier, ProviderChannelAccess, ProviderBestAvailable, ProviderOperatingClass, ProviderChannelNumber, ProviderRepeatRate, ProviderIpService, ProviderIpv6Address, ProviderMacAddress, ProviderServicePort, ProviderRcpiThreshold, ProviderWsaCountThreshold, ProviderWsaCountThresholdInterval, ProviderServiceStatus*	Status
Provider	*ProviderChannelInfo Table*	SEQUENCE OF *ProviderChannelInfo TableEntry*	*ProviderChannelInfoTableIndex, ProviderChannelInfoOperatingClass, ProviderChannelInfoChannelNumber, ProviderChannelInfoAdaptable, ProviderChannelInfoDataRate, ProviderChannelInfoTransmitPower Level, EDCA Parameter Set: ProviderChannelInfoEdcaBkCWmin, ProviderChannelInfoEdcaBkCWmax, ProviderChannelInfoEdcaBkAifsn,*	Capability

MIB segment	MIB item	Table entry	Contents	Type
			ProviderChannelInfoEdcaBkTxop Limit, ProviderChannelInfoEdcaBk Mandatory, ProviderChannelInfoEdcaBeCWmin, ProviderChannelInfoEdcaBeCWmax, ProviderChannelInfoEdcaBeAifsn, ProviderChannelInfoEdcaBeTxop Limit, ProviderChannelInfoEdcaBe Mandatory, ProviderChannelInfoEdcaViCWmin, ProviderChannelInfoEdcaViCWmax, ProviderChannelInfoEdcaViAifsn, ProviderChannelInfoEdcaViTxopLimit, ProviderChannelInfoEdcaVi Mandatory, ProviderChannelInfoEdcaVoCWmin, ProviderChannelInfoEdcaVoCWmax, ProviderChannelInfoEdcaVoAifsn, ProviderChannelInfoEdcaVoTxop Limit, ProviderChannelInfoEdcaVo Mandatory	
Provider	*ProviderWaveRouting Advertisement*		*ProviderWaveRoutingAdvertisement RouterLifetime, ProviderWaveRoutingAdvertisement IpPrefix, ProviderWaveRoutingAdvertisement PrefixLength, ProviderWaveRoutingAdvertisement DefaultGateway, ProviderWaveRoutingAdvertisement GatewayMACAddress, ProviderWaveRoutingAdvertisement PrimaryDns, ProviderWaveRoutingAdvertisement SecondaryDns*	Capability
User	*UserServiceRequestTable*	SEQUENCE OF *UserServiceRequest TableEntry*	*UserServiceRequestTableIndex, UserServiceRequestType, UserServiceRequestProviderService Identifier, UserServiceRequestProviderService Context, UserServiceRequestPriority, UserServiceRequestWsaTypes, UserServiceRequestSourceMac Address, UserServiceRequestAdvertiser Identifier, UserServiceRequestOperatingClass, UserServiceRequestChannelNumber, UserServiceRequestLinkQuality, UserServiceRequestImmediateAccess, UserServiceRequestExtendedAccess, UserServiceStatus*	Status
User	*UserAvailableService Table*	SEQUENCE OF *UserAvailableService TableEntry*	*UserAvailableServiceTableIndex, UserAvailableWsaType, UserAvailableResultCode, UserAvailableGenerationTime,*	Status

MIB segment	MIB item	Table entry	Contents	Type
			UserAvailableLifetime, *UserAvailableExpectedCrlTime* *UserAvailableSourceMacAddress,* *UserAvailableProviderService Identifier,* *UserAvailableServiceSpecific Permissions,* *UserAvailableServicePriority,* *UserAvailableProviderServiceContext,* *UserAvailableIpv6Address,* *UserAvailableServicePort,* *UserAvailableProviderMacAddress,* *UserAvailableRcpiThreshold,* *UserAvailableRcpi,* *UserAvailableWsaCountThreshold,* *UserAvailableOperatingClass,* *UserAvailableChannelNumber,* *UserAvailableAdaptable,* *UserAvailableDataRate,* *UserAvailableTransmitPowerLevel,* *UserAvailableChannelAccess,* *UserAvailableAdvertiserIdentifier,* *UserAvailableTxLatitude,* *UserAvailableTxLongitude,* *UserAvailableTxElevation,* *UserAvailableTxPositionConfidence,* *UserAvailableTxElevationConfidence,* *UserAvailableTxPositionAccuracy,* *UserAvailableLinkQuality,* *UserAvailableServiceStatus* *EDCA Parameter Set:* *UserAvailableEdcaBkCWmin,* *UserAvailableEdcaBkCWmax,* *UserAvailableEdcaBkAifsn,* *UserAvailableEdcaBkTxopLimit,* *UserAvailableEdcaBkMandatory,* *UserAvailableEdcaBeCWmin,* *UserAvailableEdcaBeCWmax,* *UserAvailableEdcaBeAifsn,* *UserAvailableEdcaBeTxopLimit,* *UserAvailableEdcaBeMandatory,* *UserAvailableEdcaViCWmin,* *UserAvailableEdcaViCWmax,* *UserAvailableEdcaViAifsn,* *UserAvailableEdcaViTxopLimit,* *UserAvailableEdcaViMandatory,* *UserAvailableEdcaVoCWmin,* *UserAvailableEdcaVoCWmax,* *UserAvailableEdcaVoAifsn,* *UserAvailableEdcaVoTxopLimit,* *UserAvailableEdcaVoMandatory*	

Annex B

(normative)

ASN.1 encoding of the WME MIB

```
--
*********************************************************************
-- * IEEE P1609.3 Management Information Base
--
*********************************************************************
IEEE1609dot3-MIB DEFINITIONS ::= BEGIN
IMPORTS
 MODULE-IDENTITY, OBJECT-TYPE, Integer32, Unsigned32 FROM SNMPv2-SMI
 MacAddress, TruthValue FROM SNMPv2-TC
 Ipv6Address, Ipv6AddressPrefix FROM IPV6-TC
 MODULE-COMPLIANCE, OBJECT-GROUP FROM SNMPv2-CONF;

--
*********************************************************************
-- * MODULE IDENTITY
--
*********************************************************************
ieee1609dot3v2mib MODULE-IDENTITY
LAST-UPDATED "201008060000Z"
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@moing.net"
DESCRIPTION
"The MIB module for IEEE 1609.3 full use entities.
iso(1) iso-identified-organization(3) ieee(111)
standards-association-numbered-series-standards(2) wave-stds(1609)
dot3(3) v2mib(2)"
REVISION "201008060000Z"
DESCRIPTION
"Consistent with 1609.3 full use standard."
::= { ieeeP1609 3 2}
ieeeP1609 OBJECT IDENTIFIER ::=
{1 iso-identified-organization (3) ieee (111)
standards-association-numbered-series-standards (2) 1609}

--
*********************************************************************
-- This MIB includes general info, as well as info relevant to the
-- provider role and user role.
```

```
-- General info:
--    Implementation capabilities information
--    Local info
--    CCH service request info
--    WSM service request info
-- Provider info:
--    Provider service request info
--    Channel info
--    WAVE routing advertisement (IPv6 configuration) info
-- User info:
--    Available service info
--    User service request info
-- Conformance groups

dot3generalInfo OBJECT IDENTIFIER ::= { ieee1609dot3v2mib 1 }
dot3providerInfo OBJECT IDENTIFIER ::= { ieee1609dot3v2mib 2 }
dot3userInfo OBJECT IDENTIFIER ::= { ieee1609dot3v2mib 3 }
dot3conformance OBJECT IDENTIFIER ::= { ieee1609dot3v2mib 4 }


--
************************************************************************
--
************************************************************************
-- * General Device Information
--
************************************************************************
--
************************************************************************

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

dot3StationConfigTable OBJECT-TYPE
SYNTAX SEQUENCE OF Dot3StationConfigEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Station Configuration attributes pertinent to networking services.
In tabular form to allow for multiple instances on an agent."
::= { dot3generalInfo 1 }

dot3StationConfigEntry OBJECT-TYPE
SYNTAX Dot3StationConfigEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
            "An entry in the dot3StationConfigTable. Each entry
            corresponds to one of multiple IEEE 802.11 interfaces
            (if present)."
INDEX { dot3StationConfigIndex }
::= { dot3StationConfigTable 1 }

Dot3StationConfigEntry ::=
```

```
SEQUENCE {
            dot3StationConfigIndex INTEGER,
            dot3CchAccessImplemented TruthValue,
            dot3SchAccessImplemented TruthValue,
            dot3IpV6Implemented TruthValue,
            dot3UdpImplemented TruthValue,
            dot3TcpImplemented TruthValue,
            dot3WsmpImplemented TruthValue,
            dot3WsmpSImplemented TruthValue,
            dot3UserRoleImplemented TruthValue,
            dot3ProviderRoleImplemented TruthValue,
            dot3TimingAdvertisementServiceImplemented TruthValue,
            dot3ManagementDataServiceImplemented TruthValue
            }

dot3StationConfigIndex OBJECT-TYPE
SYNTAX INTEGER (1..32)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"Index to the Station Config Table."
::= { dot3StationConfigEntry 1 }

dot3CchAccessImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports operation on the
            control channel."
::= { dot3StationConfigEntry 2 }

dot3SchAccessImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports operation on any
            service channel."
::= { dot3StationConfigEntry 3 }

dot3IpV6Implemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports IPv6 data exchange."
::= { dot3StationConfigEntry 4 }

dot3UdpImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports the User Datagram
            Protocol."
::= { dot3StationConfigEntry 5 }
```

```
dot3TcpImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports the Transmisison
            Control Protocol."
::= { dot3StationConfigEntry 6 }

dot3WsmpImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports the WAVE Short
            Message Protocol."
::= { dot3StationConfigEntry 7 }

dot3WsmpSImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports the WAVE Short
            Message Protocol safety supplement."
::= { dot3StationConfigEntry 8 }

dot3UserRoleImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device implements the User role."
::= { dot3StationConfigEntry 9 }

dot3ProviderRoleImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates  whether  the  device  implements  the  Provider
role."
::= { dot3StationConfigEntry 10 }

dot3TimingAdvertisementServiceImplemented OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports the generation of
            Timing  Advertisement  frames  at  the  request  of  higher
layers."
::= { dot3StationConfigEntry 11 }

dot3ManagementDataServiceImplemented OBJECT-TYPE
SYNTAX TruthValue
```

```
MAX-ACCESS read-only
STATUS current
DESCRIPTION
            "Indicates whether the device supports the generation of
            Vendor Specific Action frames carrying management data
            at the request of higher layers."
::= { dot3StationConfigEntry 12 }


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

--
************************************************************************
-- * WAVE Local Device Information
--
************************************************************************

-- WAVE Local Device Information
dot3LocalInfo OBJECT IDENTIFIER ::= { dot3generalInfo 2}
-- DEFINED AS "Parameters controlling local device operation";

dot3NumberOfChannelsSupported OBJECT-TYPE
            SYNTAX  INTEGER (1..126)
            MAX-ACCESS read-write
            STATUS current
            DESCRIPTION
            "The number of radio channels that may be simultaneously
            accessed by the device."
::= { dot3LocalInfo 1 }

dot3AdvertiserIdentifier OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(1..31))
            MAX-ACCESS read-write
            STATUS current
            DESCRIPTION
            "A text string optionally sent in the WSA identifying
            the service provider device."
::= { dot3LocalInfo 2 }

dot3RegistrationPort OBJECT-TYPE
            SYNTAX  INTEGER
            MAX-ACCESS read-write
            STATUS current
            DESCRIPTION
            "UDP port number used for registration of external
            applications."
::= { dot3LocalInfo 3 }

dot3WsmMaxLength OBJECT-TYPE
            SYNTAX  INTEGER
            MAX-ACCESS read-write
            STATUS current
            DESCRIPTION
```

```
            "Maximum size in octets of the variable length portion
            of a WSM, including Data.
            The default value is 1400."
::= { dot3LocalInfo 4 }

--
********************************************************************
-- * END WAVE General Device Information
--
********************************************************************

--
********************************************************************
-- * WAVE CCH Service Request Table
--
********************************************************************

-- WAVE CCH Service Request Table
-- DEFINED AS "Parameters populated by a CCH service request";

dot3CchServiceRequestTable OBJECT-TYPE
            SYNTAX SEQUENCE OF Dot3CchServiceRequestTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "CCH Service Request Table."
::= { dot3generalInfo 3}

dot3CchServiceRequestTableEntry OBJECT-TYPE
            SYNTAX Dot3CchServiceRequestTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Requests for CCH service."
            INDEX {dot3CchServiceRequestTableIndex}
::= { dot3CchServiceRequestTable 1}

Dot3CchServiceRequestTableEntry ::= SEQUENCE {
            dot3CchServiceRequestTableIndex INTEGER,
            dot3CchServiceRequestChannelInterval INTEGER,
            dot3CchServiceRequestPriority INTEGER,
            dot3CchServiceRequestStatus INTEGER}

dot3CchServiceRequestTableIndex OBJECT-TYPE
            SYNTAX  INTEGER (0..127)
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Table index."
::= { dot3CchServiceRequestTableEntry 1 }

dot3CchServiceRequestChannelInterval OBJECT-TYPE
        SYNTAX  INTEGER {
                        cchInterval (1),
                        schInterval (2),
                        both (3)
                }
```

```
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Channel interval(s) to which the request applies."
::= { dot3CchServiceRequestTableEntry 2 }

dot3CchServiceRequestPriority OBJECT-TYPE
        SYNTAX  INTEGER (0..63)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Service priority associated with the request."
::= { dot3CchServiceRequestTableEntry 3 }

dot3CchServiceRequestStatus OBJECT-TYPE
        SYNTAX  INTEGER {
                        pending (0),
                        satisfied (1),
                        partiallySatisfied (2)
                    }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Current status of the request."
::= { dot3CchServiceRequestTableEntry 4 }

--
************************************************************************
-- * END WAVE CCH Service Request Table
--
************************************************************************
--
************************************************************************
-- * WSM Service Request Table
--
************************************************************************
-- WSM Service Request Table
--  DEFINED AS "Parameters populated by a WAVE Short Message service
request";

dot3WsmServiceRequestTable OBJECT-TYPE
            SYNTAX SEQUENCE OF Dot3WsmServiceRequestTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "WSM Service Request Table."
::= { dot3generalInfo 4}

dot3WsmServiceRequestTableEntry OBJECT-TYPE
            SYNTAX Dot3WsmServiceRequestTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Requests for WSM service."
            INDEX {dot3WsmServiceRequestTableIndex}
::= { dot3WsmServiceRequestTable 1}
```

```
Dot3WsmServiceRequestTableEntry ::= SEQUENCE {
            dot3WsmServiceRequestTableIndex INTEGER,
            dot3WsmServiceRequestPsid OCTET STRING}

dot3WsmServiceRequestTableIndex OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Table index."
::= { dot3WsmServiceRequestTableEntry 1 }

dot3WsmServiceRequestPsid OBJECT-TYPE
        SYNTAX  OCTET STRING (SIZE(1..8))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Provider Service Identifier to be internally routed
            on receipt."
::= { dot3WsmServiceRequestTableEntry 2 }

--
********************************************************************
-- * END WSM Service Request Table
--
********************************************************************

--
********************************************************************
--
********************************************************************
-- * END General Device Information
--
********************************************************************
--
********************************************************************

--
********************************************************************
--
********************************************************************
-- * Provider Information
--          - Provider Service Request Table
--          - Provider Channel Info Table
--          - Provider WAVE Routing Advertisement Table
--
********************************************************************
--
********************************************************************

--
********************************************************************
-- *  Provider Service Request Table
--
********************************************************************
-- WAVE Provider Service Request Table
-- DEFINED AS "Parameters populated by a provider service request";
```

```
dot3ProviderServiceRequestTable OBJECT-TYPE
            SYNTAX SEQUENCE OF Dot3ProviderServiceRequestTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Provider Service Request Table."
::= { dot3providerInfo 1}

dot3ProviderServiceRequestTableEntry OBJECT-TYPE
            SYNTAX Dot3ProviderServiceRequestTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Requests for provider services."
            INDEX {dot3ProviderServiceRequestTableIndex}
::= { dot3ProviderServiceRequestTable 1}

Dot3ProviderServiceRequestTableEntry ::= SEQUENCE {
            dot3ProviderServiceRequestTableIndex INTEGER,
            dot3WsaType INTEGER,
            dot3ProviderServiceIdentifier OCTET STRING,
            dot3ProviderServiceContext OCTET STRING,
            dot3ProviderServiceRequestPriority INTEGER,
            dot3ProviderServiceSpecificPermissions OCTET STRING,
            dot3ProviderSecurityServiceIdentifier OCTET STRING,
            dot3ProviderChannelAccess INTEGER,
            dot3ProviderOperatingClass INTEGER,
            dot3ProviderBestAvailable TruthValue,
            dot3ProviderChannelNumber INTEGER,
            dot3ProviderRepeatRate INTEGER,
            dot3ProviderIpService TruthValue,
            dot3ProviderIpv6Address Ipv6Address,
            dot3ProviderMacAddress MacAddress,
            dot3ProviderServicePort INTEGER,
            dot3ProviderRcpiThreshold INTEGER,
            dot3ProviderWsaCountThreshold INTEGER,
            dot3ProviderWsaCountThresholdInterval INTEGER,
            dot3ProviderServiceStatus INTEGER
            }

dot3ProviderServiceRequestTableIndex OBJECT-TYPE
            SYNTAX  INTEGER (0..127)
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Table index."
::= { dot3ProviderServiceRequestTableEntry 1 }

dot3WsaType OBJECT-TYPE
        SYNTAX  INTEGER {
                        securedWsa (1),
                        unsecuredWsa (2)
                }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
```

```
            "Security processing to be applied to the transmitted WSA."
::= { dot3ProviderServiceRequestTableEntry 2 }

dot3ProviderServiceIdentifier OBJECT-TYPE
        SYNTAX  OCTET STRING (SIZE(1..8))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "PSID associated with the request."
::= { dot3ProviderServiceRequestTableEntry 3 }

dot3ProviderServiceContext OBJECT-TYPE
        SYNTAX  OCTET STRING (SIZE(0..31))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "PSC info provided by the requester."
::= { dot3ProviderServiceRequestTableEntry 4 }

dot3ProviderServiceRequestPriority OBJECT-TYPE
        SYNTAX  INTEGER (0..63)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Service priority associated with the request."
::= { dot3ProviderServiceRequestTableEntry 5 }

dot3ProviderServiceSpecificPermissions OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(1..255))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Used in security processing for signed WSAs."
::= { dot3ProviderServiceRequestTableEntry 6 }

dot3ProviderSecurityServiceIdentifier OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(16))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Used in security processing for signed WSAs."
::= { dot3ProviderServiceRequestTableEntry 7 }

dot3ProviderChannelAccess OBJECT-TYPE
        SYNTAX  INTEGER {
                        continuous (1),
                        alternating  (2)
                }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Access type associated with the request."
::= { dot3ProviderServiceRequestTableEntry 8 }

dot3ProviderBestAvailable OBJECT-TYPE
        SYNTAX  TruthValue
            MAX-ACCESS read-only
```

```
            STATUS current
            DESCRIPTION
            "When true, indicates the best available channel
            should be used;
            when false indicates the dot3ProviderChannelNumber
            should be used."
::= { dot3ProviderServiceRequestTableEntry 9 }

dot3ProviderOperatingClass OBJECT-TYPE
        SYNTAX  INTEGER
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Operating class of the associated service channel,
            as defined in IEEE Std 802.11 and IEEE P802.11REVmb."
::= { dot3ProviderServiceRequestTableEntry 10 }

dot3ProviderChannelNumber OBJECT-TYPE
        SYNTAX  INTEGER (0..200)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Indicate a requested service channel number,
            as defined in IEEE 802.11. "
::= { dot3ProviderServiceRequestTableEntry 11 }

dot3ProviderRepeatRate OBJECT-TYPE
        SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Number of WSAs to be transmitted each 5 seconds."
::= { dot3ProviderServiceRequestTableEntry 12 }

dot3ProviderIpService OBJECT-TYPE
        SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Indicates whether or not the requested service supports
            IP traffic."
::= { dot3ProviderServiceRequestTableEntry 13 }

dot3ProviderIpv6Address OBJECT-TYPE
        SYNTAX  Ipv6Address
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "IP address of the server associated with the requested
            service."
::= { dot3ProviderServiceRequestTableEntry 14 }

dot3ProviderMacAddress OBJECT-TYPE
        SYNTAX  MacAddress
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
```

```
            "MAC address of the server associated with the requested
            service     if different from the transmitting device."
::= { dot3ProviderServiceRequestTableEntry 15 }

dot3ProviderServicePort OBJECT-TYPE
        SYNTAX  INTEGER
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Transport layer port number associated with the requested
            service."
::= { dot3ProviderServiceRequestTableEntry 16 }

dot3ProviderRcpiThreshold OBJECT-TYPE
        SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "The recommended power of received WSAs for accepting the
            advertised service. Coded per IEEE 802.11k/17.3.10.6."
::= { dot3ProviderServiceRequestTableEntry 17 }

dot3ProviderWsaCountThreshold OBJECT-TYPE
        SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "The recommended number of received WSAs before
            accepting the advertised service."
::= { dot3ProviderServiceRequestTableEntry 18 }

dot3ProviderWsaCountThresholdInterval OBJECT-TYPE
        SYNTAX  INTEGER (1..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "The number of 100ms intervals over which to count received
            WSAs before accepting the advertised service."
::= { dot3ProviderServiceRequestTableEntry 19 }

dot3ProviderServiceStatus OBJECT-TYPE
        SYNTAX  INTEGER {
                        pending (0),
                        satisfied (1),
                        partiallySatisfied (2)
                    }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Current status of the requested provider service request."
::= { dot3ProviderServiceRequestTableEntry 20 }

--
**********************************************************************
-- * END Provider Service Request Table
--
**********************************************************************
```

```
--
********************************************************************
-- *  Provider Channel Info Table
--
********************************************************************

-- WAVE Provider Channel Info Table
-- DEFINED AS "Parameters used to populate channel info in a WSA";

dot3ProviderChannelInfoTable OBJECT-TYPE
            SYNTAX SEQUENCE OF Dot3ProviderChannelInfoTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Provider Channel Info Table."
::= { dot3providerInfo 2}

dot3ProviderChannelInfoTableEntry OBJECT-TYPE
            SYNTAX Dot3ProviderChannelInfoTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Channel information."
            INDEX {dot3ProviderChannelInfoTableIndex}
::= { dot3ProviderChannelInfoTable 1}

Dot3ProviderChannelInfoTableEntry ::= SEQUENCE {
            dot3ProviderChannelInfoTableIndex INTEGER,
            dot3ProviderChannelInfoOperatingClass INTEGER,
            dot3ProviderChannelInfoChannelNumber INTEGER,
            dot3ProviderChannelInfoAdaptable TruthValue,
            dot3ProviderChannelInfoDataRate INTEGER,
            dot3ProviderChannelInfoTransmitPowerLevel INTEGER,
            dot3ProviderChannelInfoEdcaBkCWmin INTEGER,
            dot3ProviderChannelInfoEdcaBkCWmax INTEGER,
            dot3ProviderChannelInfoEdcaBkAifsn INTEGER,
            dot3ProviderChannelInfoEdcaBkTxopLimit INTEGER,
            dot3ProviderChannelInfoEdcaBkMandatory INTEGER,
            dot3ProviderChannelInfoEdcaBeCWmin INTEGER,
            dot3ProviderChannelInfoEdcaBeCWmax INTEGER,
            dot3ProviderChannelInfoEdcaBeAifsn INTEGER,
            dot3ProviderChannelInfoEdcaBeTxopLimit INTEGER,
            dot3ProviderChannelInfoEdcaBeMandatory INTEGER,
            dot3ProviderChannelInfoEdcaViCWmin INTEGER,
            dot3ProviderChannelInfoEdcaViCWmax INTEGER,
            dot3ProviderChannelInfoEdcaViAifsn INTEGER,
            dot3ProviderChannelInfoEdcaViTxopLimit INTEGER,
            dot3ProviderChannelInfoEdcaViMandatory INTEGER,
            dot3ProviderChannelInfoEdcaVoCWmin INTEGER,
            dot3ProviderChannelInfoEdcaVoCWmax INTEGER,
            dot3ProviderChannelInfoEdcaVoAifsn INTEGER,
            dot3ProviderChannelInfoEdcaVoTxopLimit INTEGER,
            dot3ProviderChannelInfoEdcaVoMandatory INTEGER
            }

dot3ProviderChannelInfoTableIndex OBJECT-TYPE
```

```
            SYNTAX  INTEGER (0..127)
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Table index."
::= { dot3ProviderChannelInfoTableEntry 1 }

dot3ProviderChannelInfoOperatingClass OBJECT-TYPE
            SYNTAX  INTEGER
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Operating Class."
::= { dot3ProviderChannelInfoTableEntry 2 }

dot3ProviderChannelInfoChannelNumber OBJECT-TYPE
            SYNTAX  INTEGER (0..200)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Channel Number."
::= { dot3ProviderChannelInfoTableEntry 3 }

dot3ProviderChannelInfoAdaptable OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Indicates whether data rate and tx power level
            are fixed or maximum values."
::= { dot3ProviderChannelInfoTableEntry 4 }

dot3ProviderChannelInfoDataRate OBJECT-TYPE
            SYNTAX  INTEGER (2..127)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Channel data rate in increments of 500kbit/s, from
            1 Mb/s to 63.5 Mb/s."
::= { dot3ProviderChannelInfoTableEntry 5 }

dot3ProviderChannelInfoTransmitPowerLevel OBJECT-TYPE
            SYNTAX  INTEGER (-127..127)
            UNITS "dBm"
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Channel transmit power."
::= { dot3ProviderChannelInfoTableEntry 6 }

dot3ProviderChannelInfoEdcaBkCWmin OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            minimum size of the window that shall be used by
```

```
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3ProviderChannelInfoTableEntry 7 }

dot3ProviderChannelInfoEdcaBkCWmax OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            maximum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3ProviderChannelInfoTableEntry 8 }

dot3ProviderChannelInfoEdcaBkAifsn OBJECT-TYPE
            SYNTAX  INTEGER (2..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the number of slots,
            after a SIFS duration, that the STA shall sense the
            medium idle either before transmitting or executing a
            backoff."
::= { dot3ProviderChannelInfoTableEntry 9 }

dot3ProviderChannelInfoEdcaBkTxopLimit OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the maximum number
            of microseconds of an EDCA TXOP."
::= { dot3ProviderChannelInfoTableEntry 10 }

dot3ProviderChannelInfoEdcaBkMandatory OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute, when TRUE, indicates that admission
            control is mandatory for the given AC. When False,
            this attribute indicates that the admission control
            is not mandatory for the given AC."
::= { dot3ProviderChannelInfoTableEntry 11 }

dot3ProviderChannelInfoEdcaBeCWmin OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            minimum size of the window that shall be used by
```

```
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3ProviderChannelInfoTableEntry 12 }

dot3ProviderChannelInfoEdcaBeCWmax OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            maximum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3ProviderChannelInfoTableEntry 13 }

dot3ProviderChannelInfoEdcaBeAifsn OBJECT-TYPE
            SYNTAX  INTEGER (2..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the number of slots,
            after a SIFS duration, that the STA shall sense the
            medium idle either before transmitting or executing a
            backoff."
::= { dot3ProviderChannelInfoTableEntry 14 }

dot3ProviderChannelInfoEdcaBeTxopLimit OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the maximum number
            of microseconds of an EDCA TXOP."
::= { dot3ProviderChannelInfoTableEntry 15 }

dot3ProviderChannelInfoEdcaBeMandatory OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute, when TRUE, indicates that admission
            control is mandatory for the given AC. When False,
            this attribute indicates that the admission control
            is not mandatory for the given AC."
::= { dot3ProviderChannelInfoTableEntry 16 }

dot3ProviderChannelInfoEdcaViCWmin OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            minimum size of the window that shall be used by
```

```
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3ProviderChannelInfoTableEntry 17 }

dot3ProviderChannelInfoEdcaViCWmax OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            maximum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3ProviderChannelInfoTableEntry 18 }

dot3ProviderChannelInfoEdcaViAifsn OBJECT-TYPE
            SYNTAX  INTEGER (2..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the number of slots,
            after a SIFS duration, that the STA shall sense the
            medium idle either before transmitting or executing a
            backoff."
::= { dot3ProviderChannelInfoTableEntry 19 }

dot3ProviderChannelInfoEdcaViTxopLimit OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the maximum number
            of microseconds of an EDCA TXOP."
::= { dot3ProviderChannelInfoTableEntry 20 }

dot3ProviderChannelInfoEdcaViMandatory OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute, when TRUE, indicates that admission
            control is mandatory for the given AC. When False,
            this attribute indicates that the admission control
            is not mandatory for the given AC."
::= { dot3ProviderChannelInfoTableEntry 21 }

dot3ProviderChannelInfoEdcaVoCWmin OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            minimum size of the window that shall be used by
```

```
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3ProviderChannelInfoTableEntry 22 }

dot3ProviderChannelInfoEdcaVoCWmax OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            maximum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3ProviderChannelInfoTableEntry 23 }

dot3ProviderChannelInfoEdcaVoAifsn OBJECT-TYPE
            SYNTAX  INTEGER (2..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the number of slots,
            after a SIFS duration, that the STA shall sense the
            medium idle either before transmitting or executing a
            backoff."
::= { dot3ProviderChannelInfoTableEntry 24 }

dot3ProviderChannelInfoEdcaVoTxopLimit OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the maximum number
            of microseconds of an EDCA TXOP."
::= { dot3ProviderChannelInfoTableEntry 25 }

dot3ProviderChannelInfoEdcaVoMandatory OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute, when TRUE, indicates that admission
            control is mandatory for the given AC. When False,
            this attribute indicates that the admission control
            is not mandatory for the given AC."
::= { dot3ProviderChannelInfoTableEntry 26 }

--
************************************************************************
-- * END Provider Channel Info Table
--
************************************************************************
```

```
--
**********************************************************************
-- *  Provider WAVE Routing Advertisement
--
**********************************************************************

-- WAVE Routing Advertisement
-- DEFINED AS "Parameters used to populate WRA in a WSA";

dot3ProviderWaveRoutingAdvertisement    OBJECT    IDENTIFIER    ::=    {
dot3providerInfo 3}
--  DEFINED  AS  "Parameters  controlling  operation  in  the  WAVE  IP
network";

dot3ProviderWaveRoutingAdvertisementRouterLifetime OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Router lifetime."
::= { dot3ProviderWaveRoutingAdvertisement 1 }

dot3ProviderWaveRoutingAdvertisementIpPrefix OBJECT-TYPE
            SYNTAX  Ipv6AddressPrefix
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "IP prefix."
::= { dot3ProviderWaveRoutingAdvertisement 2 }

dot3ProviderWaveRoutingAdvertisementPrefixLength OBJECT-TYPE
            SYNTAX  INTEGER (1..128)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "IP prefix length."
::= { dot3ProviderWaveRoutingAdvertisement 3 }

dot3ProviderWaveRoutingAdvertisementDefaultGateway OBJECT-TYPE
            SYNTAX  Ipv6Address
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Default gateway IP address."
::= { dot3ProviderWaveRoutingAdvertisement 4 }

dot3ProviderWaveRoutingAdvertisementGatewayMACAddress OBJECT-TYPE
            SYNTAX  MacAddress
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Default gateway MAC address."
::= { dot3ProviderWaveRoutingAdvertisement 5 }

dot3ProviderWaveRoutingAdvertisementPrimaryDns OBJECT-TYPE
            SYNTAX  Ipv6Address
            MAX-ACCESS read-only
```

```
            STATUS current
            DESCRIPTION
            "Primary DNS server IP address."
::= { dot3ProviderWaveRoutingAdvertisement 6 }

dot3ProviderWaveRoutingAdvertisementSecondaryDns OBJECT-TYPE
            SYNTAX  Ipv6Address
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Secondary DNS server IP address."
::= { dot3ProviderWaveRoutingAdvertisement 7 }


--
************************************************************************
-- * END Provider WAVE Routing Advertisement Table
--
************************************************************************

--
************************************************************************
--
************************************************************************
-- * END Provider Information
--
************************************************************************
--
************************************************************************


--
************************************************************************
--
************************************************************************
-- * User Information
--          - User Service Request Table
--          - User Available Service Table
--
************************************************************************
--
************************************************************************

--
************************************************************************
-- *  User Service Request Table
--
************************************************************************

-- WAVE User Service Request Table
-- DEFINED AS "Parameters populated by a user service request";

dot3UserServiceRequestTable OBJECT-TYPE
            SYNTAX SEQUENCE OF Dot3UserServiceRequestTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
```

```
            "User Service Request Table."
::= { dot3userInfo 1}

dot3UserServiceRequestTableEntry OBJECT-TYPE
            SYNTAX Dot3UserServiceRequestTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Requests for user services."
            INDEX {dot3UserServiceRequestTableIndex}
::= { dot3UserServiceRequestTable 1}

Dot3UserServiceRequestTableEntry ::= SEQUENCE {
            dot3UserServiceRequestTableIndex INTEGER,
            dot3UserServiceRequestType INTEGER,
            dot3UserServiceRequestProviderServiceIdentifier        OCTET
STRING,
            dot3UserServiceRequestProviderServiceContext OCTET STRING ,
            dot3UserServiceRequestPriority INTEGER,
            dot3UserServiceRequestWsaTypes INTEGER,
            dot3UserServiceRequestSourceMacAddress MacAddress,
            dot3UserServiceRequestAdvertiserIdentifier OCTET STRING ,
            dot3UserServiceRequestOperatingClass INTEGER,
            dot3UserServiceRequestChannelNumber INTEGER,
            dot3UserServiceRequestLinkQuality INTEGER,
            dot3UserServiceRequestImmediateAccess TruthValue,
            dot3UserServiceRequestExtendedAccess TruthValue,
            dot3UserServiceStatus INTEGER
            }

dot3UserServiceRequestTableIndex OBJECT-TYPE
            SYNTAX  INTEGER (0..4095)
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Table index."
::= { dot3UserServiceRequestTableEntry 1 }

dot3UserServiceRequestType OBJECT-TYPE
            SYNTAX  INTEGER {
                        autoAccessOnMatch (0),
                        autoAccessUnconditional (1),
                        noSchAccess (2)
                        }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Request type."
::= { dot3UserServiceRequestTableEntry 2 }

dot3UserServiceRequestProviderServiceIdentifier OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(1..8))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Requested PSID for matching."
::= { dot3UserServiceRequestTableEntry 3 }
```

```
dot3UserServiceRequestProviderServiceContext OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(0..31))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Requested PSC for matching."
::= { dot3UserServiceRequestTableEntry 4 }

dot3UserServiceRequestPriority OBJECT-TYPE
            SYNTAX  INTEGER (0..63)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Request priority."
::= { dot3UserServiceRequestTableEntry 5 }

dot3UserServiceRequestWsaTypes OBJECT-TYPE
            SYNTAX  INTEGER {
                        securedWsa (1),
                        unsecuredWsa (2),
                        securedOrUnsecured (3),
                        any (4)
                        }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Requested WSA type(s) for matching."
::= { dot3UserServiceRequestTableEntry 6 }

dot3UserServiceRequestSourceMacAddress OBJECT-TYPE
            SYNTAX  MacAddress
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Requested provider for matching."
::= { dot3UserServiceRequestTableEntry 7 }

dot3UserServiceRequestAdvertiserIdentifier OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(0..32))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Requested Advertiser Identifier for matching."
::= { dot3UserServiceRequestTableEntry 8 }

dot3UserServiceRequestOperatingClass OBJECT-TYPE
            SYNTAX  INTEGER
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Requested Operating Class for matching."
::= { dot3UserServiceRequestTableEntry 9 }

dot3UserServiceRequestChannelNumber OBJECT-TYPE
            SYNTAX  INTEGER (0..200)
            MAX-ACCESS read-only
```

```
            STATUS current
            DESCRIPTION
            "Requested Channel Number for matching."
::= { dot3UserServiceRequestTableEntry 10 }

dot3UserServiceRequestLinkQuality OBJECT-TYPE
            SYNTAX  INTEGER
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Requested Link Quality for matching."
::= { dot3UserServiceRequestTableEntry 11 }

dot3UserServiceRequestImmediateAccess OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Indicates a need for immediate SCH access."
::= { dot3UserServiceRequestTableEntry 12 }

dot3UserServiceRequestExtendedAccess OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Indicates a need for extended SCH access."
::= { dot3UserServiceRequestTableEntry 13 }

dot3UserServiceStatus OBJECT-TYPE
            SYNTAX  INTEGER {
                        pending (0),
                        satisfied (1),
                        partiallySatisfied (2)
                    }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Indicates the current status of the request."
::= { dot3UserServiceRequestTableEntry 14 }

--
*********************************************************************
-- * END User Service Request Table
--
*********************************************************************

--
*********************************************************************
-- *  User Available Service Table
--
*********************************************************************

-- WAVE User Available Service Table
-- DEFINED AS "Available services detected";

dot3UserAvailableServiceTable OBJECT-TYPE
```

```
            SYNTAX SEQUENCE OF Dot3UserAvailableServiceTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "User Available Service  Table."
::= { dot3userInfo 2}

dot3UserAvailableServiceTableEntry OBJECT-TYPE
            SYNTAX Dot3UserAvailableServiceTableEntry
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Available services."
            INDEX {dot3UserAvailableServiceTableIndex}
::= { dot3UserAvailableServiceTable 1}

Dot3UserAvailableServiceTableEntry ::= SEQUENCE {
            dot3UserAvailableServiceTableIndex INTEGER,
            dot3UserAvailableWsaType INTEGER,
            dot3UserAvailableResultCode INTEGER,
            dot3UserAvailableGenerationTime OCTET STRING,
            dot3UserAvailableLifetime OCTET STRING,
            dot3UserAvailableExpectedCrlTime OCTET STRING,
            dot3UserAvailableSourceMacAddress MacAddress,
            dot3UserAvailableProviderServiceIdentifier OCTET STRING,
            dot3UserAvailableServiceSpecificPermissions OCTET STRING,
            dot3UserAvailableServicePriority INTEGER,
            dot3UserAvailableProviderServiceContext OCTET STRING,
            dot3UserAvailableIpv6Address Ipv6Address,
            dot3UserAvailableServicePort INTEGER,
            dot3UserAvailableProviderMacAddress MacAddress,
            dot3UserAvailableRcpiThreshold INTEGER,
            dot3UserAvailableRcpi INTEGER,
            dot3UserAvailableWsaCountThreshold INTEGER,
            dot3UserAvailableOperatingClass INTEGER,
            dot3UserAvailableChannelNumber INTEGER,
            dot3UserAvailableAdaptable TruthValue,
            dot3UserAvailableDataRate INTEGER,
            dot3UserAvailableTransmitPowerLevel INTEGER,
            dot3UserAvailableChannelAccess INTEGER,
            dot3UserAvailableAdvertiserIdentifier OCTET STRING,
            dot3UserAvailableTxLatitude INTEGER,
            dot3UserAvailableTxLongitude INTEGER,
            dot3UserAvailableTxElevation OCTET STRING,
            dot3UserAvailableTxPositionConfidence INTEGER,
            dot3UserAvailableTxElevationConfidence INTEGER,
            dot3UserAvailableTxPositionAccuracy OCTET STRING,
            dot3UserAvailableLinkQuality INTEGER,
            dot3UserAvailableServiceStatus INTEGER,
            dot3UserAvailableEdcaBkCWmin INTEGER,
            dot3UserAvailableEdcaBkCWmax INTEGER,
            dot3UserAvailableEdcaBkAifsn INTEGER,
            dot3UserAvailableEdcaBkTxopLimit INTEGER,
            dot3UserAvailableEdcaBkMandatory INTEGER,
            dot3UserAvailableEdcaBeCWmin INTEGER,
            dot3UserAvailableEdcaBeCWmax INTEGER,
            dot3UserAvailableEdcaBeAifsn INTEGER,
```

```
            dot3UserAvailableEdcaBeTxopLimit INTEGER,
            dot3UserAvailableEdcaBeMandatory INTEGER,
            dot3UserAvailableEdcaViCWmin INTEGER,
            dot3UserAvailableEdcaViCWmax INTEGER,
            dot3UserAvailableEdcaViAifsn INTEGER,
            dot3UserAvailableEdcaViTxopLimit INTEGER,
            dot3UserAvailableEdcaViMandatory INTEGER,
            dot3UserAvailableEdcaVoCWmin INTEGER,
            dot3UserAvailableEdcaVoCWmax INTEGER,
            dot3UserAvailableEdcaVoAifsn INTEGER,
            dot3UserAvailableEdcaVoTxopLimit INTEGER,
            dot3UserAvailableEdcaVoMandatory INTEGER
            }

dot3UserAvailableServiceTableIndex OBJECT-TYPE
            SYNTAX  INTEGER (0..4095)
            MAX-ACCESS not-accessible
            STATUS current
            DESCRIPTION
            "Table index."
::= { dot3UserAvailableServiceTableEntry 1 }

dot3UserAvailableWsaType OBJECT-TYPE
            SYNTAX  INTEGER {
                        securedWsa (1),
                        unsecuredWsa (2)
                    }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Security level of the received WSA."
::= { dot3UserAvailableServiceTableEntry 2 }

dot3UserAvailableResultCode OBJECT-TYPE
            SYNTAX  INTEGER         {
                        success (1),
                        invalidInput (2),
                        certificateNotFound (3),
                        unsupportedSignerType (4),
                        notMostRecentWSA (5),
                        couldNotConstructChain (6),
                        incorrectCaCertificateType (7),
                        inconsistentCertificateSubjectType (8),
                        inconsistentPermissions (9),
                        inconsistentGeographicScope (10),
                        unauthorizedGenerationLocation (11),
                        unauthorizedPSIDandPriorityInWSA (12),
                        revokedCertificate (13),
                        noUpToDateCRL (14),
                        unavailablePermissions (15),
                        unavailableGeographicScope (16),
                        certificateVerificationFailed (17),
                        messageVerificationFailed (18),
                        otherFailure (19)
                        }
            MAX-ACCESS read-only
            STATUS current
```

```
            DESCRIPTION
            "Security level of the received WSA."
::= { dot3UserAvailableServiceTableEntry 3 }

dot3UserAvailableGenerationTime OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(8))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Generation time of the WSA security information per
            IEEE Std 1609.2."
::= { dot3UserAvailableServiceTableEntry 4 }

dot3UserAvailableLifetime OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(8))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Lifetime of the WSA security information per
            IEEE Std 1609.2."
::= { dot3UserAvailableServiceTableEntry 5 }

dot3UserAvailableExpectedCrlTime OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(8))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Expected certificate revocation list time
            of the WSA security information per IEEE Std 1609.2."
::= { dot3UserAvailableServiceTableEntry 6 }

dot3UserAvailableSourceMacAddress OBJECT-TYPE
            SYNTAX  MacAddress
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Transmitter of the WSA."
::= { dot3UserAvailableServiceTableEntry 7 }

dot3UserAvailableProviderServiceIdentifier OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(1..8))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "PSID of the available service."
::= { dot3UserAvailableServiceTableEntry 8 }

dot3UserAvailableServiceSpecificPermissions OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(1..32))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
"Availably for user service to determine if provider service meets
the user's security criteria."
::= { dot3UserAvailableServiceTableEntry 9 }

dot3UserAvailableServicePriority OBJECT-TYPE
```

```
            SYNTAX  INTEGER (0..63)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Priority of the available service."
::= { dot3UserAvailableServiceTableEntry 10 }

dot3UserAvailableProviderServiceContext OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(0..31))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "PSC of the available service."
::= { dot3UserAvailableServiceTableEntry 11 }

dot3UserAvailableIpv6Address OBJECT-TYPE
            SYNTAX  Ipv6Address
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "IP address of the service provider."
::= { dot3UserAvailableServiceTableEntry 12 }

dot3UserAvailableServicePort OBJECT-TYPE
            SYNTAX  INTEGER
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Transport protocol port number of the available service."
::= { dot3UserAvailableServiceTableEntry 13 }

dot3UserAvailableProviderMacAddress OBJECT-TYPE
            SYNTAX  MacAddress
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "MAC address of the service provider, if different from
            the provider."
::= { dot3UserAvailableServiceTableEntry 14 }

dot3UserAvailableRcpiThreshold OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Suggested reception level of the received WSA.
            Coded per IEEE 802.11k/17.3.10.6."
::= { dot3UserAvailableServiceTableEntry 15 }

dot3UserAvailableRcpi OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Actual level of the received WSA.
            Coded per IEEE 802.11k/17.3.10.6."
::= { dot3UserAvailableServiceTableEntry 16 }
```

```
dot3UserAvailableWsaCountThreshold OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Suggested number of received WSA
            for service acceptance."
::= { dot3UserAvailableServiceTableEntry 17 }

dot3UserAvailableOperatingClass OBJECT-TYPE
            SYNTAX  INTEGER
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Operating Class of the channel."
::= { dot3UserAvailableServiceTableEntry 18 }

dot3UserAvailableChannelNumber OBJECT-TYPE
            SYNTAX  INTEGER (0..200)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Number of the SCH associated with the service."
::= { dot3UserAvailableServiceTableEntry 19 }

dot3UserAvailableAdaptable OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Indicates whether data rate and transmit power
            are interpreted as maximum allowed values."
::= { dot3UserAvailableServiceTableEntry 20 }

dot3UserAvailableDataRate OBJECT-TYPE
            SYNTAX  INTEGER (2..127)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Data rate on the indicated channel, in increments
            of 500kbit/s from 1 Mb/s to 63.5 Mb/s."
::= { dot3UserAvailableServiceTableEntry 21 }

dot3UserAvailableTransmitPowerLevel OBJECT-TYPE
            SYNTAX  INTEGER (-127..127)
            UNITS "dBm"
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Power level on the indicated channel."
::= { dot3UserAvailableServiceTableEntry 22 }

dot3UserAvailableChannelAccess OBJECT-TYPE
            SYNTAX  INTEGER {
                        schOnly (0),
                        schAndCch (1)
```

```
                }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Indicates the duty cycle of the advertised service."
::= { dot3UserAvailableServiceTableEntry 23 }

dot3UserAvailableAdvertiserIdentifier OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(0..32))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Advertiser Identifier of the advertising device."
::= { dot3UserAvailableServiceTableEntry 24 }

dot3UserAvailableTxLatitude OBJECT-TYPE
            SYNTAX  INTEGER (-900000000..900000001)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Latitude of the advertising device
            LSB = 1/10 micro degree
            Providing a range of plus-minus 90 degrees,
            with 900000001 indicating unavailable."
::= { dot3UserAvailableServiceTableEntry 25 }

dot3UserAvailableTxLongitude OBJECT-TYPE
            SYNTAX  INTEGER (-1800000000..1800000001)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Longitude of the advertising device
            LSB = 1/10 micro degree
            Providing a range of plus-minus 180 degrees,
            with 1800000001 indicating unavailable."
::= { dot3UserAvailableServiceTableEntry 26 }

dot3UserAvailableTxElevation OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(2))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Elevation of the advertising device.
            1 decimeter LSB (10 cm)
            Encode elevations from 0 to 6143.9 meters
            above the reference ellipsoid as 0x0000 to 0xEFFF.
            Encode elevations from -409.5 to -0.1 meters,
            i.e. below the reference ellipsoid, as 0xF001 to 0xFFFF
            unknown as 0xF000."
::= { dot3UserAvailableServiceTableEntry 27 }

dot3UserAvailableTxPositionConfidence OBJECT-TYPE
            SYNTAX  INTEGER (0..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Confidence in the position of the advertising device.
```

```
            unavailable (0),  -- B'0000  Not Equipped or unavailable
            a500m   (1), -- 500m  or about 5 * 10 ^ -3 decimal degrees
            a200m   (2), -- 200m  or about 2 * 10 ^ -3 decimal degrees
            a100m   (3), -- 100m  or about 1 * 10 ^ -3 decimal degrees
            a50m    (4), -- 50m   or about 5 * 10 ^ -4 decimal degrees
            a20m    (5), -- 20m   or about 2 * 10 ^ -4 decimal degrees
            a10m    (6), -- 10m   or about 1 * 10 ^ -4 decimal degrees
            a5m     (7), -- 5m    or about 5 * 10 ^ -5 decimal degrees
            a2m     (8), -- 2m    or about 2 * 10 ^ -5 decimal degrees
            a1m     (9), -- 1m    or about 1 * 10 ^ -5 decimal degrees
            a50cm  (10), -- 0.50m or about 5 * 10 ^ -6 decimal degrees
            a20cm  (11), -- 0.20m or about 2 * 10 ^ -6 decimal degrees
            a10cm  (12), -- 0.10m or about 1 * 10 ^ -6 decimal degrees
            a5cm   (13), -- 0.05m or about 5 * 10 ^ -7 decimal degrees
            a2cm   (14), -- 0.02m or about 2 * 10 ^ -7 decimal degrees
            a1cm   (15)  -- 0.01m or about 1 * 10 ^ -7 decimal degrees"
::= { dot3UserAvailableServiceTableEntry 28 }

dot3UserAvailableTxElevationConfidence OBJECT-TYPE
            SYNTAX  INTEGER (0..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Confidence in the elevation of the advertising device.
            unavailable (0),  -- B'0000  Not Equipped or unavailable
            elev-500-00 (1),  -- B'0001  (500 m)
            elev-200-00 (2),  -- B'0010  (200 m)
            elev-100-00 (3),  -- B'0011  (100 m)
            elev-050-00 (4),  -- B'0100  (50 m)
            elev-020-00 (5),  -- B'0101  (20 m)
            elev-010-00 (6),  -- B'0110  (10 m)
            elev-005-00 (7),  -- B'0111  (5 m)
            elev-002-00 (8),  -- B'1000  (2 m)
            elev-001-00 (9),  -- B'1001  (1 m)
            elev-000-50 (10), -- B'1010  (50 cm)
            elev-000-20 (11), -- B'1011  (20 cm)
            elev-000-10 (12), -- B'1100  (10 cm)
            elev-000-05 (13), -- B'1101  (5 cm)
            elev-000-02 (14), -- B'1110  (2 cm)
            elev-000-01 (15)  -- B'1111  (1 cm)"
::= { dot3UserAvailableServiceTableEntry 29 }

dot3UserAvailableTxPositionAccuracy OBJECT-TYPE
            SYNTAX  OCTET STRING (SIZE(4))
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Byte 1: semi-major accuracy at one standard dev
            range 0-12.7 meter, LSB = .05m
            0xFE=254=any value equal or greater than 12.70 meter
            0xFF=255=unavailable semi-major value

            Byte 2: semi-minor accuracy at one standard dev
            range 0-12.7 meter, LSB = .05m
            0xFE=254=any value equal or greater than 12.70 meter
            0xFF=255=unavailable semi-minor value
```

```
            Bytes 3-4: orientation of semi-major axis
            relative to true north (0~359.9945078786 degrees)
            LSB units of 360/65535 deg  = 0.0054932479
            a value of 0x0000 =0 shall be 0 degrees
            a value of 0x0001 =1 shall be 0.0054932479degrees
            a value of 0xFFFE =65534 shall be 359.9945078786 deg
            a value of 0xFFFF =65535 shall be used for orientation
            unavailable (In NMEA GPGST)"
::= { dot3UserAvailableServiceTableEntry 30 }

dot3UserAvailableLinkQuality OBJECT-TYPE
            SYNTAX  INTEGER
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Estimated link quality associated with the
            advertised service."
::= { dot3UserAvailableServiceTableEntry 31 }

dot3UserAvailableServiceStatus OBJECT-TYPE
            SYNTAX  INTEGER {
                        available (0),
                        active (1)
                    }
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "Indicates whether device resources have been allocated
            in support of the service."
::= { dot3UserAvailableServiceTableEntry 32 }

dot3UserAvailableEdcaBkCWmin OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            minimum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3UserAvailableServiceTableEntry  33 }

dot3UserAvailableEdcaBkCWmax OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            maximum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3UserAvailableServiceTableEntry  34 }
```

```
dot3UserAvailableEdcaBkAifsn OBJECT-TYPE
            SYNTAX  INTEGER (2..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the number of slots,
            after a SIFS duration, that the STA shall sense the
            medium idle either before transmitting or executing a
            backoff."
::= { dot3UserAvailableServiceTableEntry  35 }

dot3UserAvailableEdcaBkTxopLimit OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the maximum number
            of microseconds of an EDCA TXOP."
::= { dot3UserAvailableServiceTableEntry  36 }

dot3UserAvailableEdcaBkMandatory OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute, when TRUE, indicates that admission
            control is mandatory for the given AC. When False,
            this attribute indicates that the admission control
            is not mandatory for the given AC."
::= { dot3UserAvailableServiceTableEntry  37 }

dot3UserAvailableEdcaBeCWmin OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            minimum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3UserAvailableServiceTableEntry  38 }

dot3UserAvailableEdcaBeCWmax OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            maximum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3UserAvailableServiceTableEntry  39 }
```

```
dot3UserAvailableEdcaBeAifsn OBJECT-TYPE
            SYNTAX  INTEGER (2..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the number of slots,
            after a SIFS duration, that the STA shall sense the
medium idle either before transmitting or executing a backoff."
::= { dot3UserAvailableServiceTableEntry  40 }

dot3UserAvailableEdcaBeTxopLimit OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the maximum number
            of microseconds of an EDCA TXOP."
::= { dot3UserAvailableServiceTableEntry  41 }

dot3UserAvailableEdcaBeMandatory OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute, when TRUE, indicates that admission
            control is mandatory for the given AC. When False,
            this attribute indicates that the admission control
            is not mandatory for the given AC."
::= { dot3UserAvailableServiceTableEntry  42 }

dot3UserAvailableEdcaViCWmin OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            minimum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3UserAvailableServiceTableEntry  43 }

dot3UserAvailableEdcaViCWmax OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            maximum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3UserAvailableServiceTableEntry  44 }

dot3UserAvailableEdcaViAifsn OBJECT-TYPE
```

```
            SYNTAX  INTEGER (2..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the number of slots,
            after a SIFS duration, that the STA shall sense the
            medium idle either before transmitting or executing a
            backoff."
::= { dot3UserAvailableServiceTableEntry  45 }

dot3UserAvailableEdcaViTxopLimit OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the maximum number
            of microseconds of an EDCA TXOP."
::= { dot3UserAvailableServiceTableEntry  46 }

dot3UserAvailableEdcaViMandatory OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute, when TRUE, indicates that admission
            control is mandatory for the given AC. When False,
            this attribute indicates that the admission control
            is not mandatory for the given AC."
::= { dot3UserAvailableServiceTableEntry  47 }

dot3UserAvailableEdcaVoCWmin OBJECT-TYPE
            SYNTAX  INTEGER (0..255)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            minimum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3UserAvailableServiceTableEntry  48 }

dot3UserAvailableEdcaVoCWmax OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the value of the
            maximum size of the window that shall be used by
            a STA for a particular AC for generating a random
            number for the backoff. The value of this attribute
            shall be such that it could always be expressed in the
            form of 2X - 1, where X is an integer."
::= { dot3UserAvailableServiceTableEntry  49 }

dot3UserAvailableEdcaVoAifsn OBJECT-TYPE
```

```
            SYNTAX  INTEGER (2..15)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the number of slots,
            after a SIFS duration, that the STA shall sense the
medium idle either before transmitting or executing a backoff."
::= { dot3UserAvailableServiceTableEntry  50 }

dot3UserAvailableEdcaVoTxopLimit OBJECT-TYPE
            SYNTAX  INTEGER (0..65535)
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute shall specify the maximum number
            of microseconds of an EDCA TXOP."
::= { dot3UserAvailableServiceTableEntry  51 }

dot3UserAvailableEdcaVoMandatory OBJECT-TYPE
            SYNTAX  TruthValue
            MAX-ACCESS read-only
            STATUS current
            DESCRIPTION
            "This attribute, when TRUE, indicates that admission
            control is mandatory for the given AC. When False,
            this attribute indicates that the admission control
            is not mandatory for the given AC."
::= { dot3UserAvailableServiceTableEntry  52 }

--
********************************************************************
-- *  END User Available Service Table
--
********************************************************************

--
********************************************************************
--
********************************************************************
-- * END User Information
--
********************************************************************
--
********************************************************************

-- ********************************************************************
-- * Conformance Segment
-- ********************************************************************
dot3Compliances OBJECT IDENTIFIER ::= { dot3conformance 1 }
dot3Groups OBJECT IDENTIFIER ::= { dot3conformance 2 }

dot3Compliance MODULE-COMPLIANCE
            STATUS current
            DESCRIPTION
            "Describes the requirements for conformance to the IEEE
            1609.3 MIB."
            MODULE -- this module
```

```
            MANDATORY-GROUPS {
            dot3CapabilitiesGroup
            }

GROUP dot3IpV6Group
DESCRIPTION
"This group is mandatory for WAVE devices capable of IPv6 operation."

GROUP dot3WsmpGroup
DESCRIPTION
"This group is mandatory for WAVE devices capable of WSMP operation."

GROUP dot3RegistrationPortGroup
DESCRIPTION
"This group is mandatory for implementations supporting remote service
request over an IP network."

GROUP dot3WavRoutingAdvertisement
DESCRIPTION
"This group is mandatory for Providers sending WRAs."

GROUP dot3ChannelInfo
DESCRIPTION
"This group is mandatory for Providers sending WSA Channel Info."

GROUP dot3ProviderGroup
DESCRIPTION
"This group is optional for implementations supporting Provider
operation."

GROUP dot3ProviderIpGroup
DESCRIPTION
"This group is optional for Providers of IPv6 services."

GROUP dot3ProviderSecuredWsaGroup
DESCRIPTION
"This group is optional for Providers sending secured WSAs."

GROUP dot3SecuredUserGroup
DESCRIPTION
"This group is optional for User operation with secured WSAs."

GROUP dot3UserGroup
DESCRIPTION
"This group is optional for implementations supporting User operation."

GROUP dot3CchRequestGroup
DESCRIPTION
"This group is optional for implementations supporting CCH service
requests."

::= { dot3Compliances 1 }

-- ****************************************************************
-- * Conformance Groups
-- ****************************************************************
dot3CapabilitiesGroup OBJECT-GROUP
```

```
            OBJECTS {
            dot3NumberOfChannelsSupported,
            dot3CchAccessImplemented,
            dot3SchAccessImplemented,
            dot3IpV6Implemented,
            dot3WsmpImplemented,
            dot3UserRoleImplemented,
            dot3ProviderRoleImplemented,
            dot3TimingAdvertisementServiceImplemented,
            dot3ManagementDataServiceImplemented            }
            STATUS current
            DESCRIPTION
            "Capabilities objects"
            ::= { dot3Groups 1 }

dot3IpV6Group OBJECT-GROUP
            OBJECTS {
            dot3UdpImplemented,
            dot3TcpImplemented
            }
            STATUS current
            DESCRIPTION
            "IP protocol objects"
            ::= { dot3Groups 2 }

dot3WsmpGroup OBJECT-GROUP
            OBJECTS {
            dot3WsmpSImplemented,
            dot3WsmMaxLength,
            dot3WsmServiceRequestPsid
            }
            STATUS current
            DESCRIPTION
            "WSMP protocol objects"
            ::= { dot3Groups 3 }

dot3ProviderGroup OBJECT-GROUP
            OBJECTS {
            dot3AdvertiserIdentifier,
            dot3WsaType,
            dot3ProviderServiceIdentifier,
            dot3ProviderServiceContext,
            dot3ProviderServiceRequestPriority,
            dot3ProviderChannelAccess,
            dot3ProviderBestAvailable,
            dot3ProviderOperatingClass,
            dot3ProviderChannelNumber,
            dot3ProviderRepeatRate,
            dot3ProviderMacAddress,
            dot3ProviderRcpiThreshold,
            dot3ProviderWsaCountThreshold,
            dot3ProviderWsaCountThresholdInterval,
            dot3ProviderServiceStatus
            }
            STATUS current
            DESCRIPTION
            "Provider request and channel objects."
```

```
            ::= { dot3Groups 4 }

dot3ProviderIpGroup OBJECT-GROUP
            OBJECTS {
            dot3ProviderIpService,
            dot3ProviderIpv6Address,
            dot3ProviderServicePort
            }
            STATUS current
            DESCRIPTION
            "Provider IPv6-related objects"
            ::= { dot3Groups 5 }

dot3ProviderSecuredWsaGroup OBJECT-GROUP
            OBJECTS {
            dot3ProviderServiceSpecificPermissions,
            dot3ProviderSecurityServiceIdentifier
            }
            STATUS current
            DESCRIPTION
            "Provider objects for secured WSA"
            ::= { dot3Groups 6 }

dot3UserGroup OBJECT-GROUP
            OBJECTS {
            dot3UserServiceRequestType,
            dot3UserServiceRequestProviderServiceIdentifier,
            dot3UserServiceRequestProviderServiceContext,
            dot3UserServiceRequestPriority,
            dot3UserServiceRequestWsaTypes,
            dot3UserServiceRequestSourceMacAddress,
            dot3UserServiceRequestAdvertiserIdentifier,
            dot3UserServiceRequestOperatingClass,
            dot3UserServiceRequestChannelNumber,
            dot3UserServiceRequestLinkQuality,
            dot3UserServiceRequestImmediateAccess,
            dot3UserServiceRequestExtendedAccess,
            dot3UserServiceStatus,
            dot3UserAvailableWsaType,
            dot3UserAvailableSourceMacAddress,
            dot3UserAvailableProviderServiceIdentifier,
            dot3UserAvailableServiceSpecificPermissions,
            dot3UserAvailableServicePriority,
            dot3UserAvailableProviderServiceContext,
            dot3UserAvailableIpv6Address,
            dot3UserAvailableServicePort,
            dot3UserAvailableProviderMacAddress,
            dot3UserAvailableRcpiThreshold,
            dot3UserAvailableRcpi,
            dot3UserAvailableWsaCountThreshold,
            dot3UserAvailableOperatingClass,
            dot3UserAvailableChannelNumber,
            dot3UserAvailableAdaptable,
            dot3UserAvailableDataRate,
            dot3UserAvailableTransmitPowerLevel,
            dot3UserAvailableChannelAccess,
            dot3UserAvailableAdvertiserIdentifier,
```

```
            dot3UserAvailableTxLatitude,
            dot3UserAvailableTxLongitude,
            dot3UserAvailableTxElevation,
            dot3UserAvailableTxPositionConfidence,
            dot3UserAvailableTxElevationConfidence,
            dot3UserAvailableTxPositionAccuracy,
            dot3UserAvailableLinkQuality,
            dot3UserAvailableServiceStatus,
            dot3UserAvailableEdcaBkCWmin,
            dot3UserAvailableEdcaBkCWmax,
            dot3UserAvailableEdcaBkAifsn,
            dot3UserAvailableEdcaBkTxopLimit,
            dot3UserAvailableEdcaBkMandatory,
            dot3UserAvailableEdcaBeCWmin,
            dot3UserAvailableEdcaBeCWmax,
            dot3UserAvailableEdcaBeAifsn,
            dot3UserAvailableEdcaBeTxopLimit,
            dot3UserAvailableEdcaBeMandatory,
            dot3UserAvailableEdcaViCWmin,
            dot3UserAvailableEdcaViCWmax,
            dot3UserAvailableEdcaViAifsn,
            dot3UserAvailableEdcaViTxopLimit,
            dot3UserAvailableEdcaViMandatory,
            dot3UserAvailableEdcaVoCWmin,
            dot3UserAvailableEdcaVoCWmax,
            dot3UserAvailableEdcaVoAifsn,
            dot3UserAvailableEdcaVoTxopLimit,
            dot3UserAvailableEdcaVoMandatory
            }
            STATUS current
            DESCRIPTION
            "User service request and available service objects"
            ::= { dot3Groups 7 }

dot3SecuredUserGroup OBJECT-GROUP
            OBJECTS {
            dot3UserAvailableResultCode,
            dot3UserAvailableGenerationTime,
            dot3UserAvailableLifetime,
            dot3UserAvailableExpectedCrlTime
            }
            STATUS current
            DESCRIPTION
            "User available service objects related to secured WSAs"
            ::= { dot3Groups 8 }

dot3CchRequestGroup OBJECT-GROUP
            OBJECTS {
            dot3CchServiceRequestChannelInterval,
            dot3CchServiceRequestPriority,
            dot3CchServiceRequestStatus
            }
            STATUS current
            DESCRIPTION
            "CCH service request objects"
            ::= { dot3Groups 9 }
```

```
dot3RegistrationPortGroup OBJECT-GROUP
            OBJECTS {
            dot3RegistrationPort
            }
            STATUS current
            DESCRIPTION
            "Service registration port objects"
            ::= { dot3Groups 10 }

dot3WavRoutingAdvertisement OBJECT-GROUP
            OBJECTS {
            dot3ProviderWaveRoutingAdvertisementRouterLifetime,
            dot3ProviderWaveRoutingAdvertisementIpPrefix,
            dot3ProviderWaveRoutingAdvertisementPrefixLength,
            dot3ProviderWaveRoutingAdvertisementDefaultGateway,
            dot3ProviderWaveRoutingAdvertisementGatewayMACAddress,
            dot3ProviderWaveRoutingAdvertisementPrimaryDns,
            dot3ProviderWaveRoutingAdvertisementSecondaryDns
            }
            STATUS current
            DESCRIPTION
            "WAVE Routing Advertisement objects"
            ::= { dot3Groups 11 }

dot3ChannelInfo OBJECT-GROUP
            OBJECTS {
            dot3ProviderChannelInfoOperatingClass,
            dot3ProviderChannelInfoChannelNumber,
            dot3ProviderChannelInfoAdaptable,
            dot3ProviderChannelInfoDataRate,
            dot3ProviderChannelInfoTransmitPowerLevel,
            dot3ProviderChannelInfoEdcaBkCWmin,
            dot3ProviderChannelInfoEdcaBkCWmax,
            dot3ProviderChannelInfoEdcaBkAifsn,
            dot3ProviderChannelInfoEdcaBkTxopLimit,
            dot3ProviderChannelInfoEdcaBkMandatory,
            dot3ProviderChannelInfoEdcaBeCWmin,
            dot3ProviderChannelInfoEdcaBeCWmax,
            dot3ProviderChannelInfoEdcaBeAifsn,
            dot3ProviderChannelInfoEdcaBeTxopLimit,
            dot3ProviderChannelInfoEdcaBeMandatory,
            dot3ProviderChannelInfoEdcaViCWmin,
            dot3ProviderChannelInfoEdcaViCWmax,
            dot3ProviderChannelInfoEdcaViAifsn,
            dot3ProviderChannelInfoEdcaViTxopLimit,
            dot3ProviderChannelInfoEdcaViMandatory,
            dot3ProviderChannelInfoEdcaVoCWmin,
            dot3ProviderChannelInfoEdcaVoCWmax,
            dot3ProviderChannelInfoEdcaVoAifsn,
            dot3ProviderChannelInfoEdcaVoTxopLimit,
            dot3ProviderChannelInfoEdcaVoMandatory
            }
            STATUS current
            DESCRIPTION
            "WAVE Routing Advertisement objects"
            ::= { dot3Groups 12 }
```

```
--
************************************************************************
-- * End of 1609.3 MIB
--
************************************************************************
END
```

Annex C

(normative)

Protocol Implementation Conformance Statement (PICS) proforma[9]

Notes:

- Entries in the Item column may be hierarchical. Thus, an entry of the form N<a>.<b> indicates the item is part of the group identified by the item N<a> where all members of the group are subject to the conditions of applicability of N<a> (i.e., features lower in the numbering hierarchy (N<a>.<b>) are only applicable if the next higher level feature (N<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 only one of the items that have status C<n> is mandatory.

[9] *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
N1.	**General capabilities**		—	—	
N1.1.	***Channel access capabilities***		—	—	
N1.1.1.	CCH access		6.2	O1	
N1.1.2.	SCH access		6.2	O1	
N1.1.3.	Continuous channel access		6.2	M	
N1.1.4.	Alternating CCH/SCH access		6.2	O	
N1.1.5.	Extended SCH access		6.2	O	
N1.1.6.	Immediate SCH access		6.2	O	
N1.1.7.	Simultaneous operation on multiple channels	( )[a]	6.2	O	
N2.	**DATA PLANE**		—	—	
N2.1.	LLC		5.2	M	
N2.1.1.	SNAP		5.2	M	
N2.1.2.	LLC extensions for WSMP		7.5	N2.3:M	
N2.2.	**IPv6**		5.3, 6.5	O2	
N2.2.1.	Service channel usage only		6.5.1	M	
N2.2.2.	Use stateless configuration		6.5	O	
N2.2.3.	IP readdressing		6.5.2	M	
N2.2.4.	Send IP datagrams		5.3	O3	
N2.2.5.	Receive IP datagrams		5.3	O3	
N2.2.5.1.	Receive by link-local address		6.5	M	
N2.2.5.2.	Receive by global address		6.5	M	
N2.2.5.3.	Receive by host multicast addresses		6.5	O4	
N2.2.5.4.	Receive by router multicast addresses		6.5	O4	
N2.2.6.	UDP		5.4	O	
N2.2.7.	TCP		5.4	O	
N2.2.8.	Other IETF protocols	( )[b]	5.4	O	
N2.3.	**WSMP**		5.5	O2	
N2.3.1.	***WSMP reception***		5.5.3	O5	
N2.3.1.1.	Check WSMP version number	( )[c]	5.5.3	O	
N2.3.1.2.	Extension WAVE element IDs		8.1.1	M	
N2.3.1.3.	WSMP WAVE element IDs		5.5.3	M	
N2.3.1.4.	Deliver message based on PSID		5.5.3, 8.3.3	M	
N2.3.2.	***WSM transmission***		5.5.2	O5	
N2.3.2.1.	Insert WSMP version number		8.3.2	M	
N2.3.2.2.	Insert PSID		5.5.2, 8.3.3	M	
N2.3.2.3.	Outbound message size	( )[d]	5.5.2, 8.3.6	M	
N2.3.2.4.	Transmit channel number		8.3.4.2	O	
N2.3.2.5.	Transmit data rate		8.3.4.3	O	
N2.3.2.6.	Transmit Power Used		8.3.4.4	O	
N2.3.2.7.	WSMP WAVE Element ID		8.3.5	M	
N2.3.3.	**WSMP-S**		—		
N2.3.3.1.	WSMP-S delivery on receipt		F.2.3	O	
N2.3.3.2.	WSMP-S transmission		F.2.2	O	
N3.	**MANAGEMENT PLANE**		—	—	
N3.1.	**User role**		6.2.1	O	
N3.1.1.	Verify and accept Secured WSA		6.4.2	O6	
N3.1.2.	Accept unsecured WSA		6.4.2	O6	
N3.1.3.	Extension WAVE element IDs		8.1.1	M	
N3.1.4.	Calculate available service link quality		6.4.3	O	
N3.1.5.	***WSA header***		8.2.2		
N3.1.5.1.	Check WAVE version number	( )[e]	8.2.2.2	M	
N3.1.5.2.	Check Change Count		8.2.2.3	O	
N3.1.5.3.	WSA header extension fields		8.2.2.4	—	

Item	Feature	Value	Reference	Status	Support
N3.1.5.3.1.	Transmit power		8.2.2.4.3	O	
N3.1.5.3.2.	2DLocation		8.2.2.4.4	O	
N3.1.5.3.3.	3DLocationAndConfidence		8.2.2.4.5	O	
N3.1.5.3.4.	AdvertiserIdentifier		8.2.2.4.6	O	
N3.1.5.3.5.	Country String		8.2.2.4.7	O	
N3.1.5.3.6.	Other extension fields	( )[f]	8.2.2.4	O	
N3.1.6.	***Service Info***		8.2.3	M	
N3.1.6.1.	Number of Service Info instances	( )[g]	8.2.3.1	M	
N3.1.6.2.	Service Info extension fields		8.2.3.6	—	
N3.1.6.2.1.	PSC		8.2.3.6.1	O	
N3.1.6.2.2.	IPv6Address		8.2.3.6.2	O	
N3.1.6.2.3.	Service Port		8.2.3.6.3	O	
N3.1.6.2.4.	Provider MAC Address		8.2.3.6.4	O	
N3.1.6.2.5.	RCPI Threshold		8.2.3.6.5	O	
N3.1.6.2.6.	WSA Count Threshold		8.2.3.6.6	O	
N3.1.6.2.6.1.	WSA Count Threshold Interval		8.2.3.6.7	O	
N3.1.6.2.7.	Other extension fields	( )[h]	8.2.3.6	O	
N3.1.7.	***Channel Info***		8.2.4	M	
N3.1.7.1.	Number of Channel Info instances	( )[i]	8.2.4	M	
N3.1.7.2.	Channel Info extension fields		8.2.4.7	—	
N3.1.7.2.1.	EDCA Parameter Set		8.2.4.7.2	O	
N3.1.7.2.2.	Channel Access		8.2.4.7.3	O	
N3.1.7.2.3.	Other extension fields	( )[j]	8.2.4.7	O	
N3.1.8.	***WRA***		8.2.5	O	
N3.1.8.1.	WRA extension fields		8.2.5.8	—	
N3.1.8.1.1.	Secondary DNS		8.2.5.8.2	O	
N3.1.8.1.2.	Gateway MAC Address		8.2.5.8.3	O	
N3.1.8.1.3.	Other extension fields	( )[k]	8.2.5.8	O	
N3.2.	**Provider role**		6.2.1	O	
N3.2.1.	Send Service Advertisements		6.2.3.3	M	
N3.2.1.1.	Send secured WSA		6.2.4.2.1, 8.2.1	O8	
N3.2.1.2.	Send unsecured WSA		6.2.4.2.1, 8.2.1	O8	
N3.2.2.	Send repeated advertisements		6.2.4.2.1	O	
N3.2.3.	Change ongoing advertisements		6.2.4.2.2	O	
N3.2.4.	Delete service		6.2.3.7, 6.2.4.5.1	O	
N3.2.5.	***WSA header***		8.2.2	M	
N3.2.5.1.	Set WAVE Version		8.2.2.2	M	
N3.2.5.2.	Set Change Count		8.2.2.3	M	
N3.2.6.	WSA header extension fields		8.2.2.4	—	
N3.2.6.1.	Repeat Rate		8.2.2.4.2	O	
N3.2.6.2.	Transmit power		8.2.2.4.3	O	
N3.2.6.3.	2DLocation		8.2.2.4.4	O	
N3.2.6.4.	3DLocationAndConfidence		8.2.2.4.5	O	
N3.2.6.5.	AdvertiserIdentifier		8.2.2.4.6	O	
N3.2.6.6.	Country String		8.2.2.4.7	O	
N3.2.6.7.	Other extended fields	( )[l]	8.2.2.4	O	
N3.2.7.	***Service Info***		8.2.3	M	
N3.2.8.	Number of Service Info instances	( )[m]	8.2.3	M	
N3.2.9.	Service Info extension fields		8.2.3.6	O	
N3.2.9.1.	PSC		8.2.3.6.1	O	
N3.2.9.2.	IPv6Address		8.2.3.6.2	O	
N3.2.9.3.	Service Port		8.2.3.6.3	O	
N3.2.9.4.	Provider MAC Address		8.2.3.6.4	O	
N3.2.9.5.	RCPI Threshold		8.2.3.6.5	O	

N3.2.9.6.	WSA Count Threshold		8.2.3.6.6	O	
N3.2.9.6.1.	WSA Count Threshold Interval		8.2.3.6.7	O	
N3.2.9.7.	Other extension fields	( )[n]	8.2.3.6	O	
N3.2.10.	***Channel Info***		8.2.4	M	
N3.2.11.	Number of Channel Info instances	( )[o]	8.2.4	M	
N3.2.12.	Channel Info extension fields		8.2.4.7	O	
N3.2.12.1.	EDCA Parameter Set		8.2.4.7.2	O	
N3.2.12.2.	Channel Access		8.2.4.7.3	O	
N3.2.12.3.	Other extension fields	( )[p]	8.2.4.7	O	
N3.2.13.	***Send WRA***		8.2.5	O	
N3.2.13.1.	WRA extension fields		8.2.5.8	O	
N3.2.13.1.1.	Secondary DNS		8.2.5.8.2	O	
N3.2.13.1.2.	Gateway MAC address		8.2.5.8.3	O	
N3.2.13.1.3.	Other extension fields	( )[q]	8.2.5.8	O	
N3.3.	**Management data transfer**		—		
N3.3.1.	Management data delivery on receipt		6.3	O	
N3.3.2.	VSA generation for management data delivery		6.2.4.4	O	
N3.4.	**Timing advertisement**		—		
N3.4.1.	Timing Advertisement generation		6.2.4.3	O	
N3.5.	**MIB maintenance**		6.6	—	
N3.5.1.	Managed WAVE device		3.1, 6.6	O	
N3.5.2.	MIB per standard		6.6	N3.5.1:M	
N3.5.3.	Other MIB	( )[r]	6.6	O	

[a]Enter number of simultaneous channels.
[b]List protocols supported.
[c]List version numbers supported.
[d]Enter maximum WAVE Short Message length.
[e]List version numbers supported.
[f]List WSA header extension fields processed on reception.
[g]Enter maximum number of Service Info instances processed on reception.
[h]List any other Service Info extensions processed on reception.
[i] Enter maximum number of Channel Info instances processed on reception.
[j]List any other Channel Info extensions processed on reception.
[k]List any other WAVE routing advertisement extensions processed on reception.
[l]List any other WSA header extensions supported on transmission.
[m]Enter maximum number of Service Info instances supported on transmission.
[n]List any other Service Info extensions supported on transmission.
[o]Enter maximum number of Channel Info instances supported on transmission.
[p]List any other Channel Info extensions supported on transmission.
[q]List any other WAVE routing advertisement extensions supported on transmission.
[r]List any other MIBs supported.

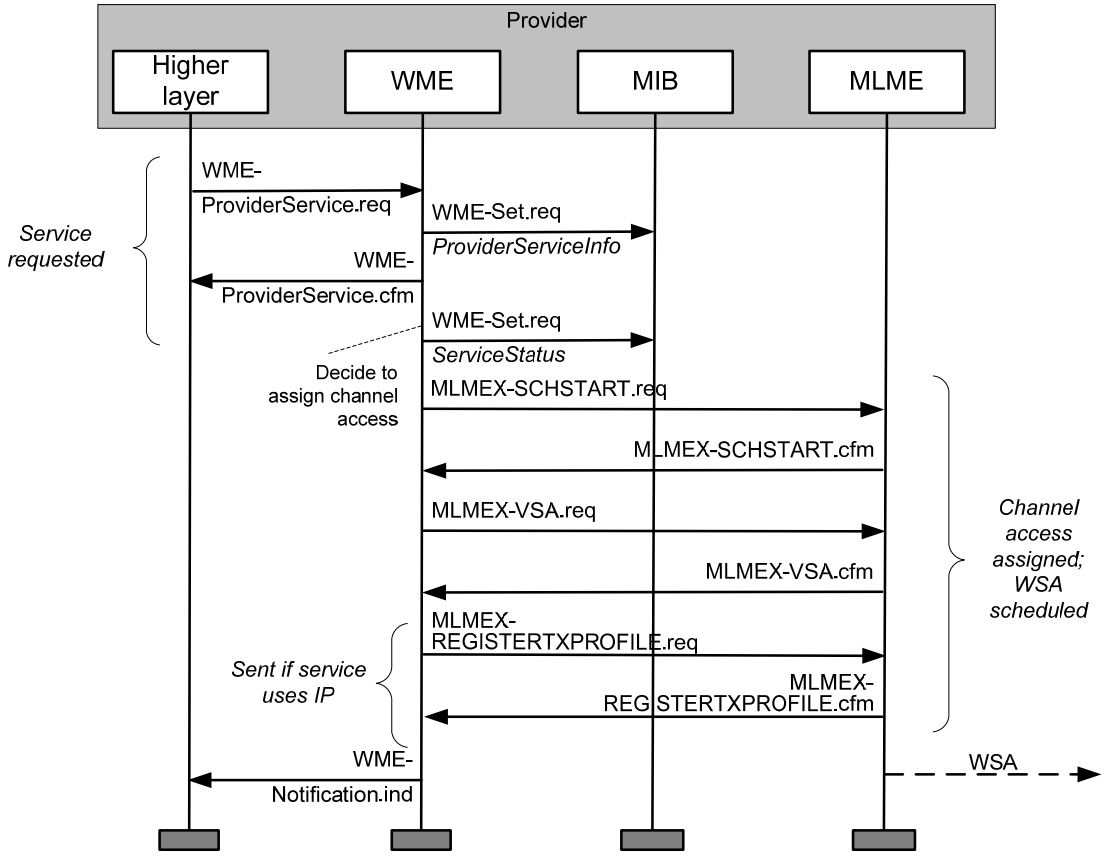
Annex D

(informative)

Service usage examples

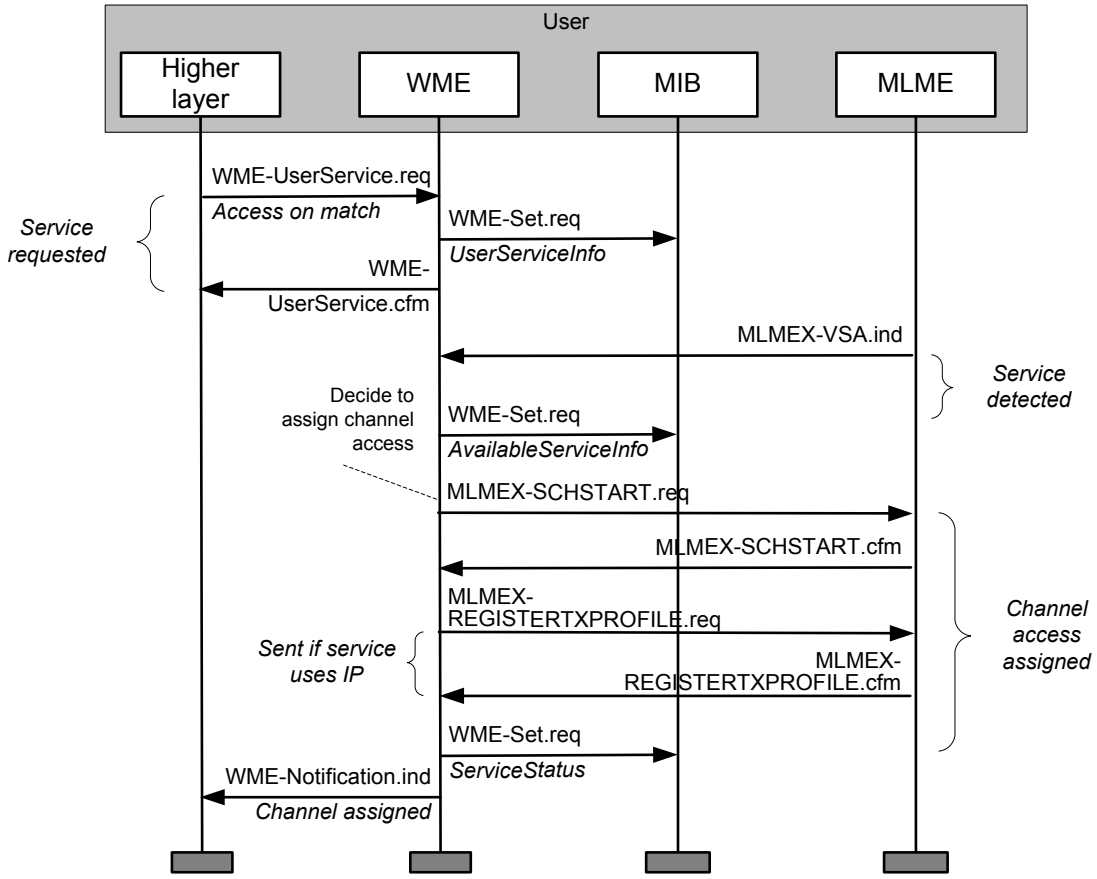
D.1 Provider service request

In this example, a higher layer entity generates a provider service request. The WME accepts the request, assigns SCH access, and triggers the MLME to start generating WSAs. This is similar to the nominal provider service request processing in the Trial Use version of WAVE.



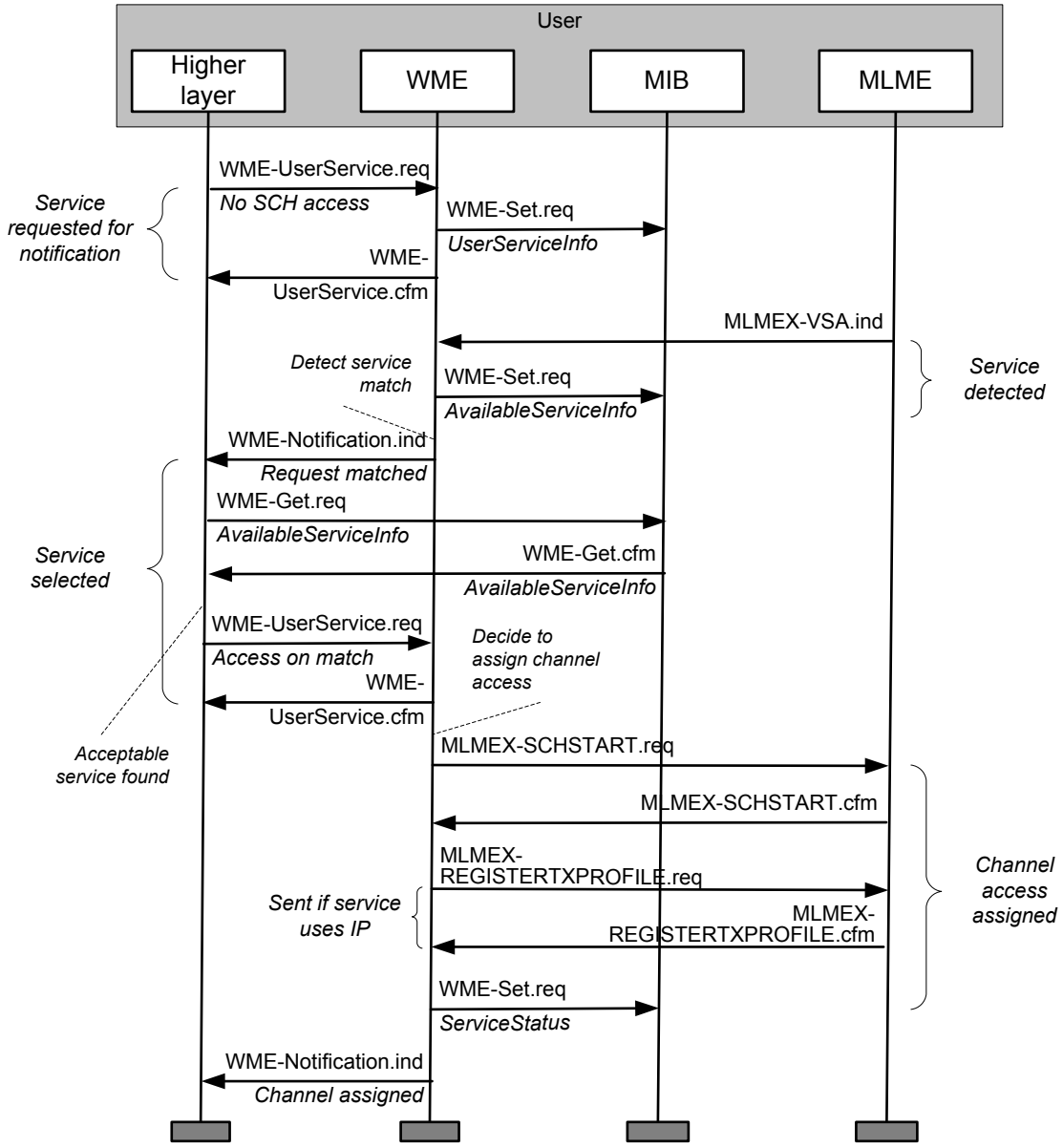
D.2 User service request with automatic channel assignment

In this example, a higher layer entity generates a user service request, with automatic SCH access on a service match. The WME, in its ongoing monitoring of WSAs, detects a matching available service and assigns SCH access. This is similar to the nominal user request processing in the Trial Use version of the WME.



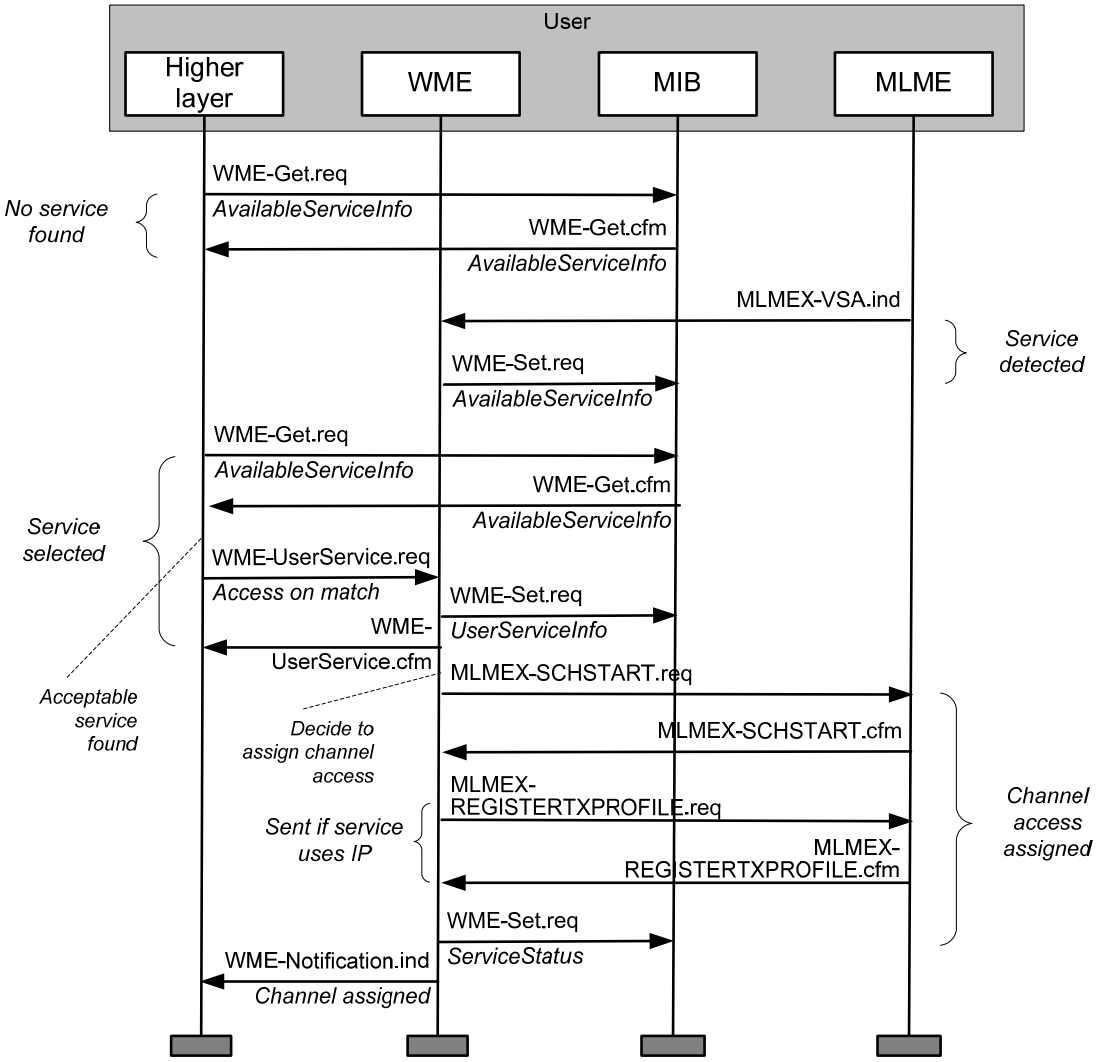
D.3 User service request with notification

In this example, a higher layer entity generates a user service request with no auto access. The WME, in its ongoing monitoring of WSAs, detects a matching available service and notifies the higher layer entity. The higher layer entity examines the available service information in the MIB, decides it is acceptable, and updates the service request to auto access. The WME notes the match with auto access and assigns SCH access.



D.4 MIB monitoring by higher layer in support of user service request

In this example, a higher layer entity requires a user service but does not send a request to the WME. Instead, it queries the WME MIB for available services. On first attempt, it finds none. On the next attempt, it detects an available service with desirable attribute, which was inserted in the MIB as part of the WME's ongoing WSA monitoring process. The higher layer entity sends a user service request to the WME with parameters identifying the desirable available service. On receipt, WME assigns SCH access.



D.5 Multi-PHY and multichannel operation

In this example, one device encompasses two distinct PHYs, allowing simultaneous operation on two separate radio channels as follows:

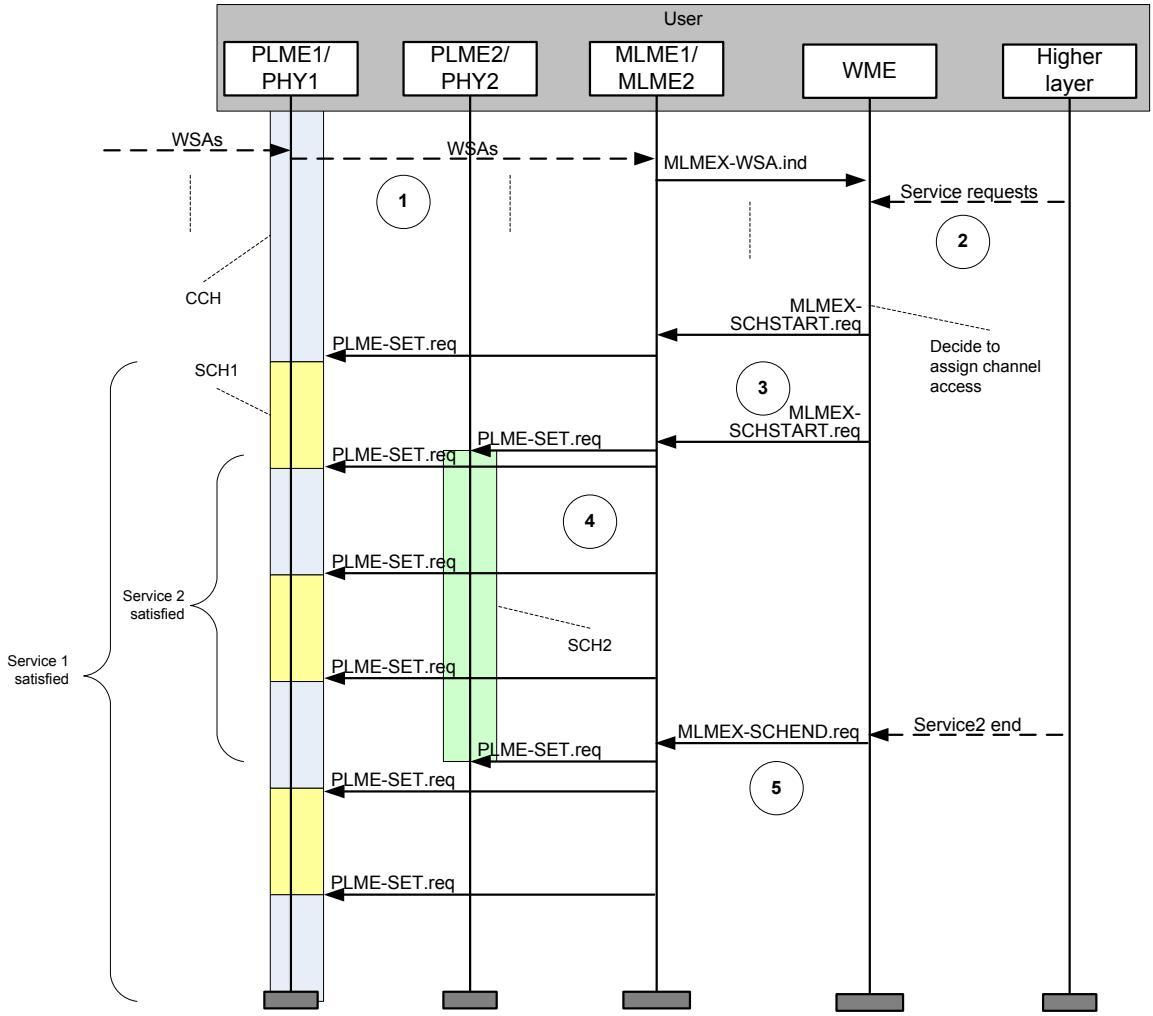
a) PHY1 is tuned to the CCH and receives WSAs, which are in turn passed to the MLME and then to the WME. PHY2 is inactive.

b) The WME accepts user service requests from higher layer entities.

c) The WME matches available services with requested services and assigns channel access, sending SCH start commands to the MAC sublayer management entities associated with the two physical layers.

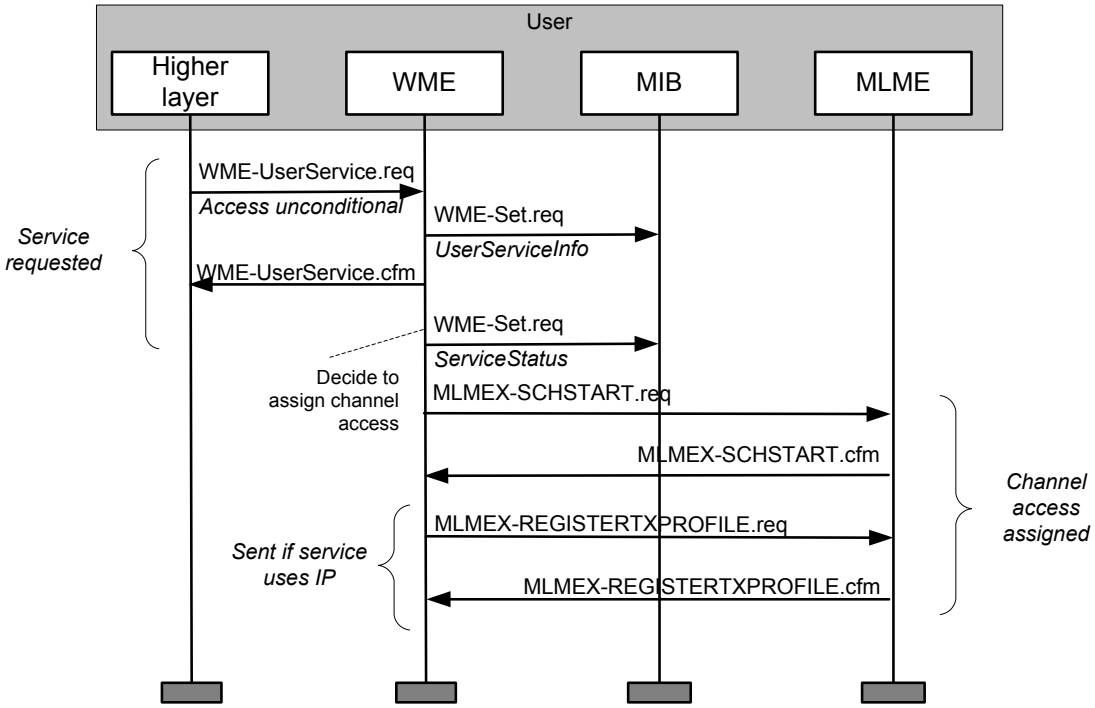
d) The MLME controls the channel switching of the two PHYs. PHY1 alternates between CCH and SCH1 on channel interval boundaries; PHY2 remains tuned to SCH2 for the duration of its service.

e) On receipt of a subsequent request from the higher layer, WME ends the SCH2 assignment and the MLME in turn causes PHY2 to return to its default state.



D.6 User service, unconditional

In this example, a higher layer entity generates a user service request that is not dependent on any received service advertisements. The WME accepts the request and assigns SCH access.



Annex E

(normative)

Allocated WAVE Element IDs

WAVE Element ID values used to identify elements in WSAs and WSM headers shall be as specified in the table. The "Used in" entries define the permitted use of the elements. The "Reference" entries define where the elements are specified.

	Element ID value	Used in	Reference
WSA WAVE Elements			
	0	Reserved	
Service Info	1	WSA Service Info	8.2.1
Channel Info	2	WSA Channel Info	8.2.1
WRA	3	WSA WRA	8.2.1
Extension WAVE Elements			
Transmit power used	4	WSA header, WSMP header	8.2.2.4, 8.3.4
2D Location	5	WSA header	8.2.2.4
3DLocationAndConfidence	6	WSA header	8.2.2.4
Advertiser identifier	7	WSA header	8.2.2.4
Provider Service Context	8	WSA Service Info	8.2.3.6
IPv6 address	9	WSA Service Info	8.2.3.6
Service port	10	WSA Service Info	8.2.3.6
Provider MAC address	11	WSA Service Info	8.2.3.6
EDCA Parameter Set	12	WSA Channel Info	8.2.5.8
Secondary DNS	13	WSA WRA	8.2.5.8
Gateway MAC Address	14	WSA WRA	8.2.5.8
Channel Number	15	WSMP header	8.3.4
DataRate	16	WSMP header	8.3.4
Repeat Rate	17	WSA header	8.2.2.4
Country String	18	WSA header	8.2.2.4
RCPI Threshold	19	WSA Service Info	8.2.3.6
WSA Count Threshold	20	WSA Service Info	8.2.3.6
Channel Access	21	WSA Channel Info	8.2.4.7
WSA Count Threshold Interval	22	WSA Service Info	8.2.3.6
	23 to 127	Reserved	
WSMP WAVE Elements			
WAVE Short Message	128	WSMP header	5.5, 7.3.2
WSMP-S	129	WSMP header	Annex F
WSMP identity supplement (WSMP-I)	130	WSMP header	IEEE P1609.1
	131 to 255	Reserved	

Annex F

(normative)

WSMP safety supplement

F.1 Introduction

The optional WAVE data plane WSMP safety supplement (WSMP-S) is specified in this annex. The purpose of WSMP-S is to provide information to remote devices about the channel switching operation of the local device. This potentially allows the remote devices to make choices concerning the transmit channel and timing of their safety WSMs to maximize channel capacity while avoiding missed messages due to channel switching. Note that only the sending side processing and information formats are specified at this time. The WAVE Short Message information passed to the higher layer at the receiving side is the same regardless of whether WSMP-S is used.

F.2 Operation

F.2.1 Overview of operation

WSMP-S adds control information to the short messages transferred via WSMP, which is specified in 5.5. Refer to Figure F.1:

a) The recipient higher layer entity registers its interest in one or more PSIDs via the usual WSM service request (see 7.4.2.6). Any WAVE Short Message information received with that PSID is delivered to the recipient whether or not WSMP-S is employed.

b) The sending higher layer entity generates information, including *Payload*, to be sent in a WAVE Short Message. The sending entity is configured to use WSMP-S, so it includes the information in a WSMS-WaveShortMessage.request primitive that it sends to the WSMP-S.

c) WSMP-S populates the WSMP-S *Control* field and prepends it to the message *Payload* as shown in F.4. WSMP-S then sends the resulting *WSMData* out through the standard WSMP process via a WSM-WaveShortMessage.request as described in 7.2 and 5.5.2. WSMP sets the *WSMP WAVE Element ID* value in the WSMP header to indicate that the WSM comes from WSMP-S (see Annex E) and otherwise treats it as a standard WSM.

d) The message transits the air interface and is delivered to WSMP on the receiving side.

e) The receiving WSMP reads the WSMP header and notes a *WSMP WAVE Element ID* indicating WSMP-S. It passes the *WSMData* to WSMP-S using the WSM-WaveShortMessage.indication primitive.

f) The receiving WSMP-S extracts the WSMP-S *Control* field and delivers the *Payload* to the intended recipient (based on PSID) via WSMS-WaveShortMessage.indication.

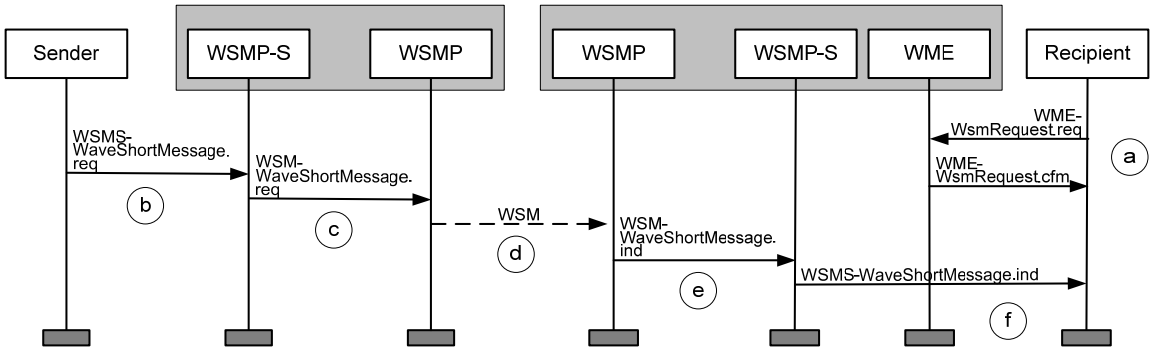


Figure F.1—Flow of information using the WSMP safety supplement

F.2.2 WSM transmission

On receipt of WSMS-WaveShortMessage.request from a higher layer, WSMP-S shall construct the *WSMP-S Control* field and prepend it to the *Payload* parameter to form the *WSMData* parameter of a WSM-WaveShortMessage.request primitive, which it sends to WSMP. Formats are specified in F.4.

F.2.3 WSM reception

On receipt of WSM-WaveShortMessage.indication from WSMP, WSMP-S shall remove the *WSMP-S Control* field from the *Data* parameter. The remaining *Payload* is passed to the destination higher layer entity via a WSMS-WaveShortMessage.indication, with the destination higher layer entity determined from the WSMP header *ProviderServiceIdentifier* and the MIB *WsmServiceRequestTable*.

F.3 Service primitives

F.3.1 General

The WSMP-S primitives allow higher layer entities to send and receive WSMs carrying WSMP-S control information.

F.3.2 WSMS-WaveShortMessage.request

F.3.2.1 Function

The WSMS-WaveShortMessage.request primitive is used by a higher layer entity to request sending a WAVE Short Message with WSMP-S functionality.

F.3.2.2 Semantics of the service primitive

The parameters of the primitive are as follows:

WSMS-WaveShortMessage.request (

Channel Number,
DataRate,
Transmit Power Level,
ProviderServiceIdentifier,
User Priority,
WsmExpiryTime,
Length,
Payload,
Peer MAC address,
WSMP header extensions
)

Payload contains the higher layer information being transferred. Other parameters are described in 7.3.2.

F.3.2.3 When generated

The primitive is generated by a higher layer entity to request sending a WAVE Short Message with WSMP-S functionality.

F.3.2.4 Effect of receipt

Receipt of the primitive triggers a WSM-WaveShortMessage.request.

F.3.3 WSMS-WaveShortMessage.confirm

F.3.3.1 Function

This primitive is used to confirm the acceptance of a WSMS-WaveShortMessage.request.

F.3.3.2 Semantics of the service primitive

The contents of the primitive are as follows:

WSMS-WaveShortMessage.confirm (
ResultCode
)

ResultCode is described in 7.3.3.

F.3.3.3 When generated

This primitive is generated in response to the receipt of a WSMS-WaveShortMessage.request primitive.

F.3.3.4 Effect of receipt

No behavior is specified.

F.3.4 WSMS-WaveShortMessage.indication

F.3.4.1 Function

The primitive indicates to a higher layer entity that a WAVE Short Message with WSMP-S functionality has been received.

F.3.4.2 Semantics of the service primitive

The parameters of the WSMS-WaveShortMessage.indication primitive are as follows:

WSMS-WaveShortMessage.indication (
WsmpVersion
Channel Number,
DataRate,
Transmit Power Used,
ProviderServiceIdentifier,
User Priority,
Length,
Payload,
Peer MAC address
)

Payload contains the higher layer information being transferred. Other parameters are described in 7.3.4.

F.3.4.3 When generated

The primitive is generated by the WSMP-S to deliver *Payload* and associated information to a higher layer entity.

F.3.4.4 Effect of receipt

No processing is specified.

F.4 Information formats

WSMP-S adds a *WSMP-S Control* field to the sender payload as illustrated in Figure F.2.

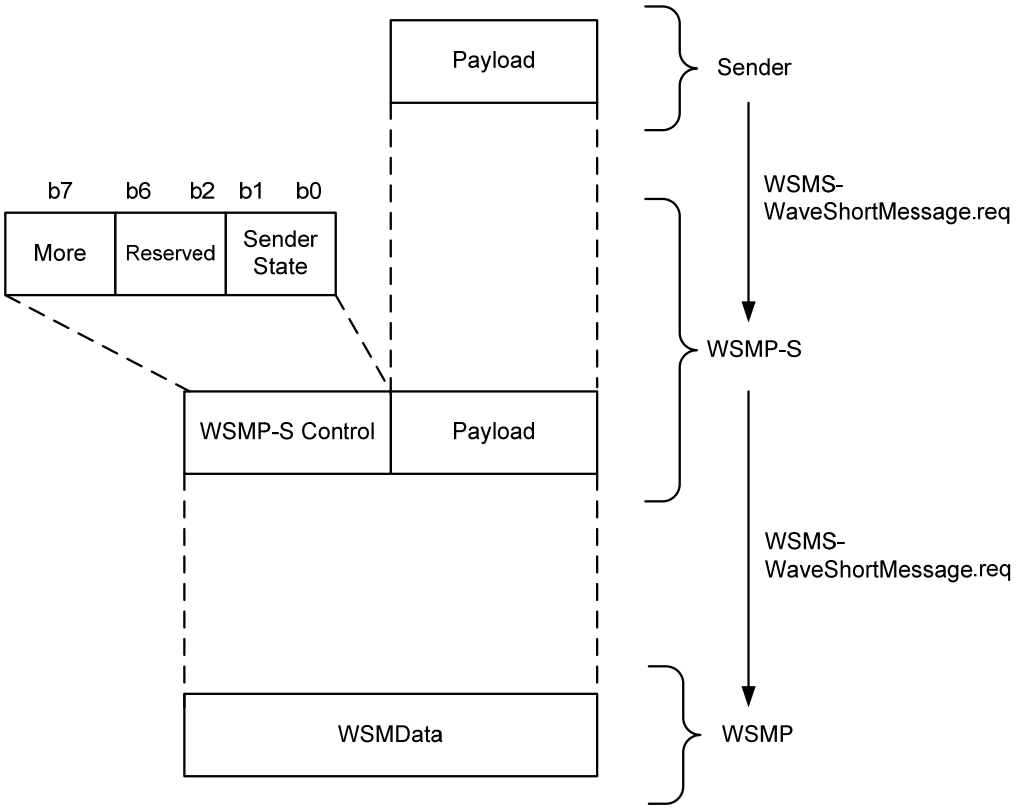


Figure F.2—*WSMP-S Control* field

The *WSMP-S Control* field is encoded as specified in Table F.1.

Table F.1—*WSMP-S Control* field coding

Name	Values	Notes
More	0 = Last octet of WSMP-S control information 1 = The next octet contains additional WSMP-S control information	The first bit of any additional control octets also will be a *More* bit. No additional control octets are defined in this standard.
Sender State	0 = Sender requires others' safety messages to be sent on the CCH during the CCH interval 1 = Sender requires others' safety messages to be sent on the CCH, but has no time interval constraint 2 = Sender is capable of receiving others' safety messages on a designated safety channel that is distinct from the CCH 3 = Sender is not capable of processing received safety messages	See the subsequent discussion for further specification of values.

The information kept by the WME channel access assignment algorithm (6.2.3.2) may be used to set the *Sender State* as shown in Table F.2. "PSID requested" is true if the WSMS-WaveShortMessage.request *ProviderServiceIdentifier* is found in the MIB *WSMServiceRequestTable*.

Table F.2—*WSMP-S Control* field coding

***Sender State* value**	**PSID requested**	**Safety channel access**	**CCH access**
0	True	Not continuous	Alternating
1	True	Not continuous	Continuous
2	True	Continuous	Don't care
3		All others	

Annex G

(informative)

Packet format examples

G.1 WSA example

This subclause provides an example of the contents of the WSA as specified in 8.1.3. In this example, the WSA is sent by a service provider who wants to provide two services. One is a lowest priority service about weather information. Another is a highest priority service about accident alert. Both services are provided on channel 172 using 12 Mb/s data rate. It is transmitted 100 times per 5 s. The transmit power is 30 dBm. All extension fields (WSA header extension field, Service Info extension field, Channel Info extension field, WRA extension field) have the structure shown in Figure 15, where field lengths are shown in octets. *WAVE Element ID* values are specified in Annex E.

Field name			**Length (octets)**	**Value (hex)**	**Description**
WSA header	*WAVE Version/Change Count*		1	06	WAVE Version = 1 (6 bits) Change Count = 2 (2 bits) 0x06 = 0b(000001)(10)
	WSA header extension fields	*Repeat Rate*	3	11 01 64	WAVE Element ID = 17 Length = 1 100 times
		Transmit Power Used	3	04 01 1E	WAVE Element ID = 4 Length = 1 30 dBm
		3DLocationAndConfidence	17	06 0F 01 7A 12 AC 07 36 F8 BB 03 E8 36 FF FF FF FF	WAVE Element ID = 6 Length = 15 latitude (4 octets): +24.777388° longitude (4 octets): +121.043131° elevation (2 octets): +100m position confidence (4 bits): 3 = 100 m elevation confidence (4 bits): 6 = 10 m positional accuracy (4 octets): 'unavailable'
		Advertiser Identifier	7	07 05 49 54 52 49 00	WAVE Element ID = 7 Length = 5 ASCII content: 'ITRI' 0x49 = I 0x54 = T 0x52 = R 0x49 = I 0x00 = NULL
		Country String	5	12 03 54 57 4F	WAVE Element ID = 18 Length = 3 Country Code: 'TW' (Taiwan) 0x54 = T 0x57 = W 0x4F = 'O': outdoor environment

Service Info	*Service Info WAVE element ID*		1	01	per Annex E
	Provider Service Identifier		1	03	PSID: 03
	ServicePriority		1	00	priority: 0 (lowest)
	Channel Index		1	01	1st set
	Service Info extension fields	*Provider Service Context*	15	08 0D 77 65 61 74 68 65 72 20 61 6E 66 6F 00	WAVE Element ID = 8 Length = 13 ASCII content: 'weather info' 0x77 = w 0x65 = e 0x61 = a etc.
	Service Info WAVE element ID		1	01	per Annex E
	Provider Service Identifier		2	80 03	PSID: 80-03
	ServicePriority		1	3F	priority: 63 (highest)
	Channel Index		1	01	1st set
	Service Info extension fields	*Provider Service Context*	17	08 0F 61 63 63 69 64 65 6E 74 20 61 6C 65 72 74 00	WAVE Element ID = 8 Length = 15 ASCII content: 'accident alert' 0x61 = a 0x63 = c 0x63 = c etc.
		IPv6 Address	18	09 10 10 80 00 00 00 00 00 00 00 08 08 00 20 0C 41 7A	WAVE Element ID = 9 Length = 16 1080:0:0:0:8:800:200C:417A
		Service Port	4	0A 02 04 D2	WAVE Element ID = 10 Length = 2 port: 1234
		Provider MAC Address	8	0B 06 00 22 C3 00 00 AB	WAVE Element ID = 11 Length = 6 00:22:C3:00:00:AB
		RCPI Threshold	3	13 01 C8	WAVE Element ID = 19 Length = 1 200 (Power = –10 dBm)
		WSA Count Threshold	3	14 01 32	WAVE Element ID = 20 Length = 1 50 times
		WSA Count Threshold Interval	3	16 01 1E	WAVE Element ID = 22 Length = 1 30 (Time = 3 s)
Channel Info	*Channel Info WAVE element ID*		1	02	per Annex E
	Operating Class		1	0E	Operating Class: 14
	Channel Number		1	AC	channel: 172
	Adaptable		1	00	0: fixed
	DataRate		1	0C	Data Rate: 6 Mb/s
	Transmit Power Level		1	1E	30 dBm

	Channel Info extension fields	*EDCA Parameter Set*	22	0C 14 0C 12 00 00 06 A4 00 00 29 A4 00 00 43 43 00 00 62 32 00 00 	WAVE Element ID = 12 Length = 20 per IEEE Std 802.11: IEEE 802.11 Element ID = 12 Length = 18 QoS Info = 0 Reserved = 0 AC_BE: Reserved (1 bit) = 0 ACI (2 bits) = 0 ACM (1bit) = 0 AIFSN (4 bits) = 6 ECWmax (4 bits) = 10 ECWmin (4 bits) = 4 TXOP limit (16 bits) = 0 AC_BK: Reserved (1 bit) = 0 ACI (2 bits) = 1 ACM (1bit) = 0 AIFSN (4 bits) = 9 ECWmax (4 bits) = 10 ECWmin (4 bits) = 4 TXOP limit (16 bits) = 0 AC_VI: Reserved (1 bit) = 0 ACI (2 bits) = 2 ACM (1bit) = 0 AIFSN (4 bits) = 3 ECWmax (4 bits) = 4 ECWmin (4 bits) = 3 TXOP limit (16 bits) = 0 AC_VO: Reserved (1 bit) = 0 ACI (2 bits) = 3 ACM (1bit) = 0 AIFSN (4 bits) = 2 ECWmax (4 bits) = 3 ECWmin (4 bits) = 2 TXOP limit (16 bits) = 0
		Channel access	3	15 01 01	WAVE Element ID = 21 Length = 1 1: alternating access

WAVE Routing Advertisement	*WRA WAVE element ID*		1	03	per Annex F
	Router Lifetime		2	07 08	1800 s
	IpPrefix		16	10 80 00 00 00 00 00 00 00 08 00 00 00 00 00 00	1080:0:0:0:8:0:0:0
	Prefix Length		1	50	80 bits
	Default Gateway		16	10 80 00 00 00 00 00 00 00 08 08 00 20 0C FF FE	1080:0:0:0:8:800:200C:FFFE
	Primary DNS		16	10 80 00 00 00 00 00 00 00 08 08 00 00 01 00 01	1080:0:0:0:8:800:1:1
	WRA Extension	*Gateway MAC Address*	8	0E 06 00 22 C3 00 00 CD	WAVE Element ID = 14 Length = 6 00:22:C3:00:00:CD

G.2 WSM example

This subclause provides an example of the contents of the WSM as specified in 8.3. The WSM is sent on channel 172 using 6 Mb/s data rate. The transmit power is 30 dBm.

Field name		Length (octets)	Value (hex)	Description
WSMP Version		1	02	The version of the WSM protocol
Provider Service Identifier		3	C0 03 05	PSID: C0-03-05
WSM extension fields	*Channel Number*	3	0F 01 AC	WAVE Element ID = 15 Length = 1 Channel: 172
	DataRate	3	10 01 0C	WAVE Element ID = 16 Length = 1 Data rate: 6 Mb/s
	Transmit Power Used	3	04 01 1E	WAVE Element ID = 4 Length = 1 30 dBm
WAVE Element ID		1	80	WAVE Short Message, per Annex E
WSMLength		2	00 0D	Indicates the length in octets (13) of the following *WSMData* field
WSMData		13	48 65 6C 6C 6F 20 57 6F 72 6C 64 21 00	ASCII content: 'Hello World!' 0x48 = H 0x65 = e 0X6C = l etc.

Annex H

(informative)

PSID examples

Figure H.1 shows examples of three PSID values in hexadecimal representation as they would be found in, for example, a WSA or WSM. The bits are depicted in each octet from left to right, most significant (e.g., b7) to least significant (e.g., b0). The octets are depicted in order of transmission (octet 0 first).

In the first example, the PSID value 03 is one octet in length (indicated by the "0" in the most significant bit).

In the second example, the PSID value 80-03 is two octets in length (indicated by the "10" in the most significant bits of the first octet).

In the third example, the PSID value C0-03-05 is three octets in length (indicated by the "110" in the most significant bits of the first octet).

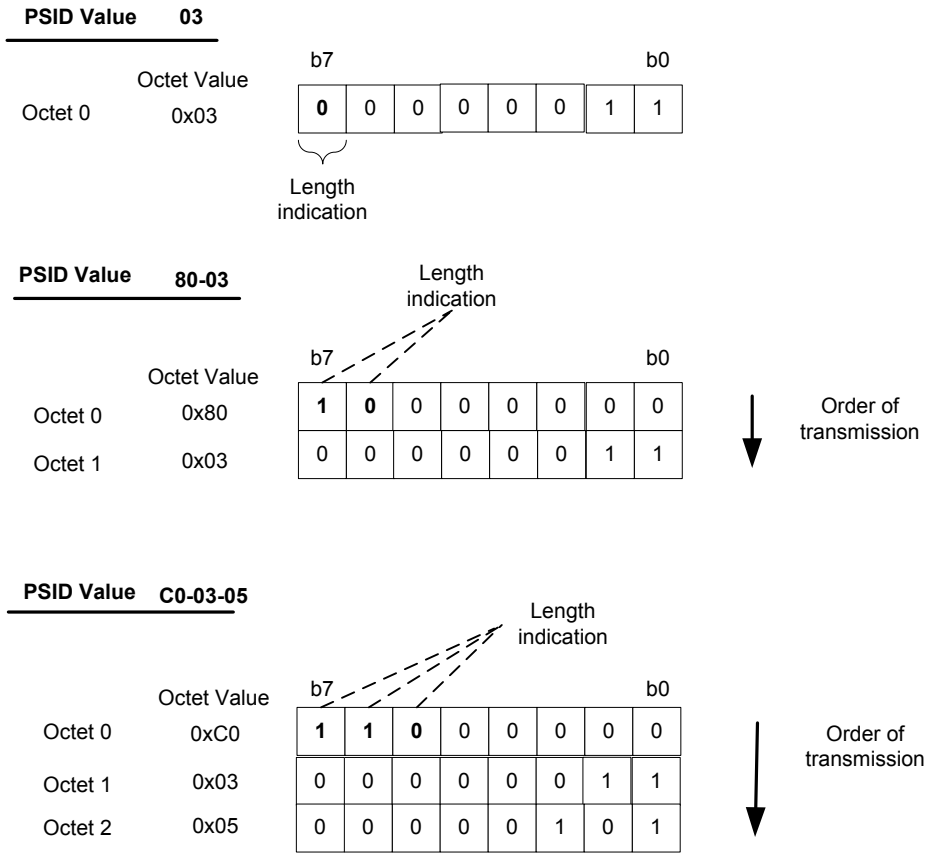


Figure H.1—PSID examples

Annex I

(informative)

Bibliography

[B1] IEEE P1609.1™/D1.3, May 2010, IEEE Draft Standard for Wireless Access in Vehicular Environments (WAVE)—Remote Management Services.[10, 11]

[B2] *IEEE Standards Style Manual*. Piscataway, NJ: IEEE, 2009.

[10] This IEEE standards project was not approved by the IEEE-SA Standards Board at the time this publication went to press. For information about obtaining a draft, contact the IEEE.
[11] IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, Piscataway, NJ 08854, USA (http://standards.ieee.org/).