

# ETSI GS MEC 028 V2.2.1 (2021-07)



## **Multi-access Edge Computing (MEC); WLAN Access Information API**

### *Disclaimer*

---

The present document has been produced and approved by the Multi-access Edge Computing (MEC) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.  
It does not necessarily represent the views of the entire ETSI membership.

---

**Reference**

RGS/MEC-0028v221WlanAPI

---

**Keywords**

API, MEC, service, WLAN

**ETSI**

---

650 Route des Lucioles  
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B  
Association à but non lucratif enregistrée à la  
Sous-Préfecture de Grasse (06) N° w061004871

---

**Important notice**

The present document can be downloaded from:

<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at [www.etsi.org/deliver](http://www.etsi.org/deliver).

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:

<https://portal.etsi.org/People/CommitteeSupportStaff.aspx>

---

**Notice of disclaimer & limitation of liability**

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

---

**Copyright Notification**

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2021.  
All rights reserved.

# Contents

Intellectual Property Rights .....	6
Foreword.....	6
Modal verbs terminology.....	6
1 Scope .....	7
2 References .....	7
2.1 Normative references .....	7
2.2 Informative references.....	8
3 Definition of terms, symbols and abbreviations.....	8
3.1 Terms.....	8
3.2 Symbols.....	8
3.3 Abbreviations .....	9
4 Overview .....	10
5 Description of the service (informative).....	10
5.1 WLAN Access Information Service introduction .....	10
5.2 Sequence diagrams .....	11
5.2.1 Introduction.....	11
5.2.2 Sending a query for Access Point information .....	11
5.2.2.1 General query procedure .....	11
5.2.2.2 Sending a query for a list of Access Points .....	11
5.2.2.3 Sending a query for WLAN capabilities .....	11
5.2.2.4 Sending a query for BSS Load.....	12
5.2.2.5 Sending a query for WAN metrics.....	12
5.2.2.6 Sending a query for AP Location.....	12
5.2.2.7 Void.....	12
5.2.2.8 Sending a query for OBSS Load .....	12
5.2.3 Sending a query for Station information.....	12
5.2.3.1 General query procedure .....	12
5.2.3.2 Sending a query for a list of stations .....	13
5.2.3.3 Sending a query for channel used by station(s).....	13
5.2.3.4 Sending a query for RSSI of station(s).....	13
5.2.3.5 Sending a query for station data rates .....	13
5.2.3.6 Sending a query for station statistics.....	14
5.2.3.7 Sending a query for Neighbor Report .....	14
5.2.3.8 Sending a query for Channel Load.....	14
5.2.4 REST based subscribe-notify model.....	14
5.2.4.1 Subscribing to WLAN event notifications .....	14
5.2.4.2 Receiving notification on expiry of WLAN event subscription .....	15
5.2.4.3 Updating subscription for WLAN event notifications .....	16
5.2.4.4 Unsubscribing from WLAN event notifications .....	16
5.2.5 Receiving WLAN event notifications about station data rates .....	17
5.2.6 Receiving WLAN event notifications about associated stations.....	18
5.2.7 Measurement Configuration .....	19
5.2.7.1 Creating a Measurement configuration .....	19
5.2.7.2 Updating a Measurement Configuration .....	20
5.2.7.3 Deleting a Measurement Configuration .....	21
5.2.7.4 Example of using a Measurement Configuration.....	21
6 Data Model .....	21
6.1 General .....	21
6.2 Resource data types .....	22
6.2.1 Introduction.....	22
6.2.2 Type: ApInfo .....	22
6.2.3 Type: StaInfo .....	22
6.2.4 Type: MeasurementConfig .....	23

6.2.5	Type: MeasurementConfigLinkList.....	23
6.3	Subscription data types.....	24
6.3.1	Introduction.....	24
6.3.2	Type: AssocStaSubscription .....	24
6.3.3	Type: StaDataRateSubscription .....	24
6.3.4	Type: SubscriptionLinkList .....	25
6.3.5	Type: MeasurementReportSubscription .....	26
6.3.6	Type: WebsocketNotifConfig.....	26
6.4	Notifications data types .....	27
6.4.1	Introduction.....	27
6.4.2	Type: AssocStaNotification .....	27
6.4.3	Type: StaDataRateNotification.....	27
6.4.4	Type: ExpiryNotification.....	27
6.4.5	Type: MeasurementReportNotification .....	28
6.4.6	Type: TestNotification.....	28
6.5	Referenced structured data types.....	28
6.5.1	Introduction.....	28
6.5.2	Type: TimeStamp .....	29
6.5.3	Type: ApIdentity.....	29
6.5.4	Type: WlanCapabilities .....	29
6.5.5	Void .....	29
6.5.6	Type: WanMetrics .....	29
6.5.7	Type: BssLoad .....	30
6.5.8	Type: ExtBssLoad .....	30
6.5.9	Type: ApLocation .....	31
6.5.10	Type: NeighborReport .....	31
6.5.11	Type: StaIdentity .....	32
6.5.12	Type: ApAssociated.....	32
6.5.13	Type: StaStatistics .....	32
6.5.14	Type: HtCapabilities .....	33
6.5.15	Type: VhtCapabilities .....	33
6.5.16	Type: HeCapabilities .....	33
6.5.17	Type: DmgCapabilities .....	33
6.5.18	Type: EdmgCapabilities .....	34
6.5.19	Type: GeoLocation .....	34
6.5.20	Type: CivicLocation .....	34
6.5.21	Type: Rssi .....	35
6.5.22	Type: StaDataRate .....	36
6.5.23	Type: LinkType .....	36
6.5.24	Type: ChannelLoadConfig .....	36
6.5.25	Type: BeaconRequestConfig .....	36
6.5.26	Type: StaStatisticsConfig .....	37
6.5.27	Type: BeaconReport .....	37
6.5.28	Type: BeaconReportingConfig .....	38
6.5.29	Type: ReportedBeaconFrameInfo.....	39
6.5.30	Type: BssidInfo .....	39
6.5.31	Type: BssCapabilities .....	40
6.5.32	Type: NeighborReportConfig .....	41
6.5.33	Type: STACounterTriggerCondition.....	41
6.5.34	Type: StaStatisticsGroupZeroData .....	41
6.5.35	Type: StaStatisticsGroupOneData .....	42
6.5.36	Type: StaStatisticsGroup2to9Data.....	42
6.5.37	Type: ReportingReasonStaCounters .....	42
6.5.38	Type: ReportingReasonQoSCounters .....	43
6.5.39	Type: ChannelLoad.....	43
6.5.40	Type: OBssLoad .....	43
6.5.41	Type: MeasurementInfo.....	44
6.6	Referenced simple data types and enumerations.....	44
7	API definition.....	44
7.1	Introduction .....	44
7.2	Global definitions and resource structure.....	44

7.3	Resource: ap_information .....	46
7.3.1	Description.....	46
7.3.2	Resource definition.....	46
7.3.3	Resource Methods .....	46
7.3.3.1	GET.....	46
7.3.3.2	PUT.....	47
7.3.3.3	PATCH .....	48
7.3.3.4	POST.....	48
7.3.3.5	DELETE .....	48
7.4	Resource: sta_information.....	48
7.4.1	Description.....	48
7.4.2	Resource definition.....	48
7.4.3	Resource Methods .....	48
7.4.3.1	GET.....	48
7.3.3.2	PUT.....	49
7.3.3.3	PATCH .....	50
7.3.3.4	POST.....	50
7.3.3.5	DELETE .....	50
7.5	Resource: subscriptions.....	50
7.5.1	Description.....	50
7.5.2	Resource definition.....	50
7.5.3	Resource methods.....	50
7.5.3.1	GET.....	50
7.5.3.2	PUT.....	51
7.5.3.3	PATCH .....	51
7.5.3.4	POST.....	51
7.5.3.5	DELETE .....	53
7.6	Resource: existing subscription.....	53
7.6.1	Description.....	53
7.6.2	Resource definition.....	53
7.6.3	Resource methods.....	53
7.6.3.1	GET.....	53
7.6.3.2	PUT.....	54
7.6.3.3	PATCH .....	56
7.6.3.4	POST.....	56
7.6.3.5	DELETE .....	56
7.7	Resource: measurements .....	57
7.7.1	Description.....	57
7.7.2	Resource definition.....	57
7.7.3	Resource methods.....	57
7.7.3.1	GET.....	57
7.5.3.2	PUT.....	58
7.5.3.3	PATCH .....	58
7.7.3.4	POST.....	58
7.7.3.5	DELETE .....	60
7.8	Resource: existing measurements.....	60
7.8.1	Description.....	60
7.8.2	Resource definition.....	60
7.8.3	Resource methods.....	60
7.8.3.1	GET.....	60
7.8.3.2	PUT.....	61
7.8.3.3	PATCH .....	63
7.8.3.4	POST.....	63
7.8.3.5	DELETE .....	63
<b>Annex A (informative):</b>	<b>Complementary material for API utilization .....</b>	<b>64</b>
History .....		65

---

## Intellectual Property Rights

### Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<https://ipr.etsi.org/>).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

### Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

**DECT™**, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

---

## Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Multi-access Edge Computing (MEC).

---

## Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

---

# 1 Scope

The present document focuses on the WLAN Access Information MEC service. It describes the message flows and the required information. The present document also specifies the RESTful API with the data model.

---

## 2 References

### 2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at <https://docbox.etsi.org/Reference/>.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] Void.

[2] IETF RFC 2818: "HTTP Over TLS".

NOTE: Available at <https://tools.ietf.org/html/rfc2818>.

[3] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

NOTE 1: Available at <https://tools.ietf.org/html/rfc5246>.

NOTE 2: Obsoleted by IETF RFC 8446.

[4] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".

NOTE: Available at <https://tools.ietf.org/html/rfc6749>.

[5] IETF RFC 6750: "The OAuth 2.0 Authorization Framework: Bearer Token Usage".

NOTE: Available at <https://tools.ietf.org/html/rfc6750>.

[6] IETF RFC 6225: "Dynamic Host Configuration Protocol Options for Coordinate-Based Location Configuration Information".

NOTE: Available at <https://tools.ietf.org/html/rfc6225>.

[7] IETF RFC 4776: "Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information".

NOTE: Available at <https://tools.ietf.org/html/rfc4776>.

[8] IEEE 802.11-2016<sup>TM</sup>: "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".

[9] Void.

[10] ETSI GS MEC 009: "Multi-access Edge Computing (MEC); General principles, patterns and common aspects of MEC Service APIs".

## 2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

- [i.1] ETSI GS MEC 011: "Multi-access Edge Computing (MEC); Edge Platform Application Enablement".
- [i.2] OpenAPI™ Specification.

NOTE 1: Available at <https://github.com/OAI/OpenAPI-Specification>.

NOTE 2: OpenAPI Specification and OpenAPI Initiative and their respective logos, are trademarks of the Linux Foundation.

- [i.3] Wi-Fi® Alliance 2014: "Hot Spot 2.0 (Release 2) Technical Specification V1.0.0".
- [i.4] ETSI GS MEC 002: "Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements".
- [i.5] ETSI GS MEC 003: "Multi-access Edge Computing (MEC); Framework and Reference Architecture".
- [i.6] ETSI GS MEC 012: "Multi-access Edge Computing (MEC); Radio Network Information API".
- [i.7] ETSI GS MEC 029: "Multi-access Edge Computing (MEC); Fixed Access Information API".
- [i.8] WiFi Alliance 2019: "Data Elements Specification v1.0".
- [i.9] ISO 3166: "Codes for the representation of names of countries and their subdivisions".
- [i.10] IEEE P802.11ax™: "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 1: Enhancement for High Efficiency WLAN".
- [i.11] IEEE P802.11ay™: "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 1: Enhancement for High Efficiency WLAN - Amendment 2: Enhanced throughput for operation in license-exempt bands above 45 GHz".
- [i.12] ETSI GS MEC 001: "Multi-access Edge Computing (MEC); Terminology".

---

## 3 Definition of terms, symbols and abbreviations

### 3.1 Terms

For the purposes of the present document, the terms given in ETSI GS MEC 001 [i.12] apply.

### 3.2 Symbols

Void.

### 3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GS MEC 001 [i.12] and the following apply:

3GPP	3 <sup>rd</sup> Generation Partnership Project
AID	Association Identifier
A-MPDU	Aggregate MAC Protocol Data Unit
A-MSDU	Aggregate MAC Service Data Unit
AP	Access Point
API	Application Programming Interface
APSD	Automatic Power Save Delivery
ASEL	Antenna Selection
BSS	Basic Service Set
BSSID	Basic Service Set Identifier
DMG	Directional Multi-Gigabit

NOTE: As in Directional Multi-Gigabit WLAN.

DSSS	Direct Sequence Spread Spectrum
EDMG	Enhanced Directional Multi-Gigabit

NOTE: As in Enhanced Directional Multi-Gigabit WLAN.

ERP	Extended Rate PHY
ESS	Extended Service Set
FCS	Frame Check Sequence
FTM	Fine Timing Measurement
HE	High Efficiency

NOTE: As in High Efficiency WLAN.

HT	High Throughput
----	-----------------

NOTE: As in High Throughput WLAN.

HTTP	Hyper Text Transport Protocol
ID	Identifier
IEEE	Institute of Electrical and Electronics Engineers
LMD	Load Measurement Duration
MAC	Medium Access Control
MCS	Modulation and Coding Scheme
MDE	Mobility Domain Element
MPDU	MAC Protocol Data Unit
NSS	Number of Spatial Streams
OBSS	Overlapping Basic Service Set
OFDM	Orthogonal Frequency Division Multiplexing
PBSS	Personal Basic Service Set
PCP	PBSS Control Point
PHY	Physical layer
PPDU	PHY Protocol Data Unit
QoS	Quality of Service
RCPI	Received Channel Power Indicator
RSNI	Received Signal-to-Noise Indicator
RSSI	Receive Signal Strength Indicator
RTS	Request To Send
SC	Single Carrier
SSID	Service Set Identifier
STA	Station
TSF	Timing Synchronization Function
TU	Time Unit
URI	Uniform Resource Identifier

VHT            Very High Throughput

NOTE:    As in Very High Throughput WLAN.

WAI            WLAN Access Information

WAIS          WLAN Access Information Service

WLAN         Wireless Local Area Network

## 4 Overview

The present document specifies the WLAN Access Information (WAI) API to support the requirements defined for Multi-access Edge Computing in ETSI GS MEC 002 [i.4].

Clause 5 provides overview how WLAN Access Information Service (WAIS) may be used by the MEC applications and by the MEC platform. It describes the information flows used for WLAN Access Information Service.

The information that can be exchanged over the WAI API is described in clause 6 which provides detailed description on all information elements that are used for WLAN Access Information.

Clause 7 describes the actual WAI API providing detailed information how information elements are mapped into a RESTful API design.

## 5 Description of the service (informative)

### 5.1 WLAN Access Information Service introduction

Multi-access Edge Computing allows running the MEC applications at the edge of the network where the environment is characterized by low latency, proximity, high bandwidth and exposure to location and up-to-date information from the underlying access networks. The information on current conditions from the WLAN access is shared via WLAN Access Information Service.

WLAN Access Information Service (WAIS) is a service that provides WLAN access related information to service consumers within MEC System. The WLAN Access Information Service is available for authorized MEC applications and is discovered over the Mp1 reference point as specified in ETSI GS MEC 003 [i.5]. The granularity of the WLAN Access Information may be adjusted based on parameters such as information per station (STA), per Access Point (AP) or per Multiple Access Points (Multi-AP).

The WLAN Access Information may be used by the MEC applications and MEC platform to optimize the existing services and to provide new type of services that are based on up to date information from WLAN access possibly combined with the information such as Radio Network Information as specified in ETSI GS MEC 012 [i.6] or Fixed Access Network Information as specified in ETSI GS MEC 029 [i.7] from the other access technologies.

The present document defines the protocol, data model and interface in the form of RESTful Application Programming Interface (APIs) specifications. Information about the Access Points and client stations can be requested either by querying or by subscribing to notifications.

The procedures defined for queries are flexible and cater wide set of use cases from simple queries to queries requesting wide set of information on targets. This flexibility is enabled with concepts of attribute-based filtering and attribute selectors, as specified in ETSI GS MEC 009 [10], and those are described in more detail in clauses 6.18 and 6.19 of ETSI GS MEC 009 [10].

## 5.2 Sequence diagrams

### 5.2.1 Introduction

The service consumers communicate with WLAN Access Information Service over WAI API to get contextual information from the WLAN access network. Both the MEC application and MEC platform may be service consumers and both the MEC platform and MEC application may be providers of WLAN Access Information.

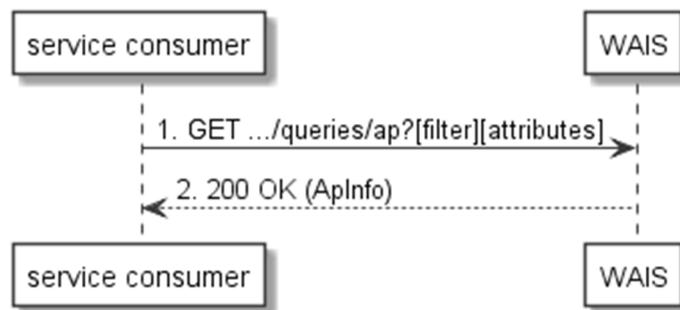
The WAI API supports both queries and subscriptions (pub/sub mechanism) over the RESTful API or over alternative transports such as message bus. Alternative transports are not specified in detail in the present document. When queries are used, the attribute-based filter expression can be used to limit the number of objects returned by query operation and attribute-selectors can be used to limit the number of attributes included in the response.

For RESTful architectural style, the present document defines the HTTP protocol bindings.

### 5.2.2 Sending a query for Access Point information

#### 5.2.2.1 General query procedure

Figure 5.2.2.1-1 shows a scenario where the service consumer (e.g. a MEC application or a MEC platform) sends a query to receive information about Access Points (AP). The response may contain information on one or more access points. The number of queried objects and desired contents can be controlled with an attribute-based filter expression and attribute-selectors as defined in ETSI GS MEC 009 [10].



**Figure 5.2.2.1-1: Flow of service consumer querying Access Point information**

A service consumer requesting Access Point information, as illustrated in Figure 5.2.2.1-1, consists of the following steps:

- 1) Service consumer sends a GET request to the resource representing the Access Point(s) information. The request may contain attribute-filter to limit the number of Access Points whose information is received and attribute-selector to limit the number of attributes included in the response.
- 2) WAIS responds with "200 OK" with the message body containing the requested Access Point information.

#### 5.2.2.2 Sending a query for a list of Access Points

A list of Access Points available in the system can be queried with the flow as in Figure 5.2.2.1-1 by using the attribute selector as follows:

```
GET .../queries/ap?fields=apId
```

#### 5.2.2.3 Sending a query for WLAN capabilities

The WLAN Capabilities of Access Points can be queried with the flow of Figure 5.2.2.1-1 by using the attribute selector as follows:

```
GET .../queries/ap?fields=apId,wlanCap
```

The above query, if successful, would return the identifiers of Access Points available together with their WLAN Capabilities.

As an example, the WLAN Capabilities of the Access Point with an apId equal to "admiralsclub" can be queried using the attribute selector and filter attribute as follows:

```
GET .../queries/ap?filter=(eq,apId,admiralsclub)&fields=apId,wlanCap
```

#### 5.2.2.4 Sending a query for BSS Load

The BSS Load of an Access Point can be queried with the flow of Figure 5.2.2.1-1 by using the attribute selector and filter attribute (in this instance the Access Point with apId equal to "admiralsclub") as follows:

```
GET .../queries/ap?filter=(eq,apId,admiralsclub)&fields=apId,bssLoad
```

More accurate information about BSS Load, for the same Access Point, can be obtained by adding the attribute "extBssLoad" in the list of requested fields as follows:

```
GET .../queries/ap?filter=(eq,apId,admiralsclub)&fields=apId,bssLoad,extBssLoad
```

#### 5.2.2.5 Sending a query for WAN metrics

The WAN metrics of an Access Point can be queried with the flow of Figure 5.2.2.1-1 by using the attribute selector and filter attribute (in this instance the Access Point with apId equal to "admiralsclub") as follows:

```
GET .../queries/ap?filter=(eq,apId,admiralsclub)&fields=apId,wanMetrics
```

#### 5.2.2.6 Sending a query for AP Location

The location of an Access Point can be queried with the flow of Figure 5.2.2.1-1 by using the attribute selector and filter attribute (in this instance the Access Point with apId equal to "admiralsclub") as follows:

```
GET .../queries/ap?filter=(eq,apId,admiralsclub)&fields=apId,apLocation
```

#### 5.2.2.7 Void

#### 5.2.2.8 Sending a query for OBSS Load

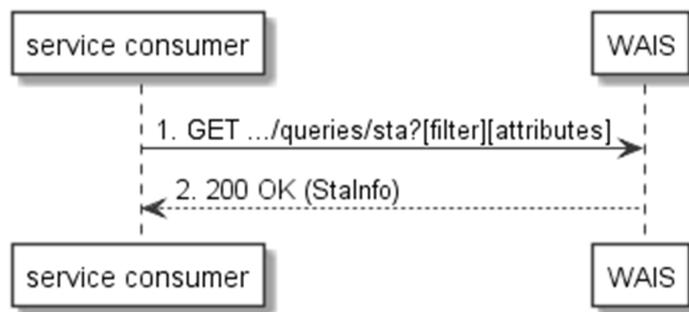
The Overlapping BSS (OBSS) Load of an Access Point can be queried with the flow of Figure 5.2.2.1-1 by using the attribute selector and filter attribute (in this instance the Access Point with apId equal to "admiralsclub") as follows:

```
GET .../queries/ap?filter=(eq,apId,admiralsclub)&fields=apId,obssLoad
```

### 5.2.3 Sending a query for Station information

#### 5.2.3.1 General query procedure

Figure 5.2.3.1-1 shows a scenario where the service consumer (e.g. a MEC application or a MEC platform) sends a request to receive information about client station(s). The response may contain information on one or more stations and the number of queried objects and desired contents can be controlled with attribute-based filtering and attribute-selectors as defined in ETSI GS MEC 009 [10].



**Figure 5.2.3.1-1: Flow of service consumer querying station Info**

A service consumer requesting client station information, as illustrated in Figure 5.2.3.1-1, consists of the following steps:

- 1) Service consumer sends a GET request to the resource representing the station(s) information. The request may contain attribute-filters to limit the number of client stations whose information is received and attribute-selectors to limit the number of attributes included in the response.
- 2) WAIS responds with "200 OK" with the message body containing the requested WLAN station information.

### 5.2.3.2 Sending a query for a list of stations

A list of stations available in the system can be queried with the flow as in Figure 5.2.3.1-1 by using the attribute selector as follows:

```
GET .../queries/sta?fields=staId
```

The above query, if successful, would return the identities of all the stations that are known to be associated in the system. Information about the Access Points that the stations are associated to can be queried as follows:

```
GET .../queries/sta?fields=staId,apAssociated
```

Further, to get the stations associated to a particular Access Point, the following query can be used including the attribute selector and filter attribute (in this instance the stations associated to Access Point with apAssociated equal to "mec123"):

```
GET .../queries/sta?filter=(eq,apAssociated,mec123)&fields=staId,apAssociated
```

### 5.2.3.3 Sending a query for channel used by station(s)

The channel used by stations can be queried with the flow of Figure 5.2.3.1-1 by using the attribute selector as follows:

```
GET .../queries/sta?fields=staId,channel
```

### 5.2.3.4 Sending a query for RSSI of station(s)

The RSSI value of stations can be queried with the flow of Figure 5.2.3.1-1 by using the attribute selector as follows:

```
GET .../queries/sta?fields=staId,rssi
```

The results can be narrowed down to stations under specific Access Point by adding filter attribute (in this instance the Access Point with apId equal to "mec123") to the query as follows:

```
GET .../queries/sta?filter=(eq,apAssociated,mec123)&fields=staId,rssi
```

### 5.2.3.5 Sending a query for station data rates

The physical layer data rate of stations can be queried with the flow of Figure 5.2.3.1-1 by using the attribute selector as follows:

```
GET .../queries/sta?fields=staId,staDataRate
```

As there may be great number of stations in the system, it may be practical to limit the query to consider either stations under specific Access Point or certain specific station by including the attribute filter:

```
GET .../queries/sta?filter=(eq,apAssociated,mec404)&fields=staId,staDataRate
```

```
GET .../queries/sta?filter=(eq,staId,C8:D0:66:08:B6:0F)&fields=staId,staDataRate
```

### 5.2.3.6 Sending a query for station statistics

The statistics of stations can be queried with the flow of Figure 5.2.3.1-1 by using the attribute selector as follows:

```
GET .../queries/sta?fields=staId,staStatistics
```

As there may be great number of stations in the system, it may be practical to limit the query to consider either stations under specific Access Point or certain specific station by including the attribute filter:

```
GET .../queries/sta?filter=(eq,apAssociated,mec404)&fields=staId,staStatistics
```

```
GET .../queries/sta?filter=(eq,staId,C8:D0:66:08:B6:0F)&fields=staId,staStatistics
```

### 5.2.3.7 Sending a query for Neighbor Report

The neighbor report of stations can be queried with the flow of Figure 5.2.3.1-1 by using the attribute selector as follows:

```
GET .../queries/sta?fields=staId,neighborReport
```

As there may be great number of stations in the system, it may be practical to limit the query to consider either stations under specific Access Point or certain specific station by including the attribute filter:

```
GET .../queries/sta?filter=(eq,apAssociated,mec404)&fields=staId,neighborReport
```

```
GET .../queries/sta?filter=(eq,staId,C8:D0:66:08:B6:0F)&fields=staId,neighborReport
```

### 5.2.3.8 Sending a query for Channel Load

The channel load as measured via stations can be queried with the flow of Figure 5.2.3.1-1 by using the attribute selector as follows:

```
GET .../queries/sta?fields=staId,channelLoad
```

As there may be great number of stations in the system, it may be practical to limit the query to consider either stations under specific Access Point or certain specific station by including the attribute filter:

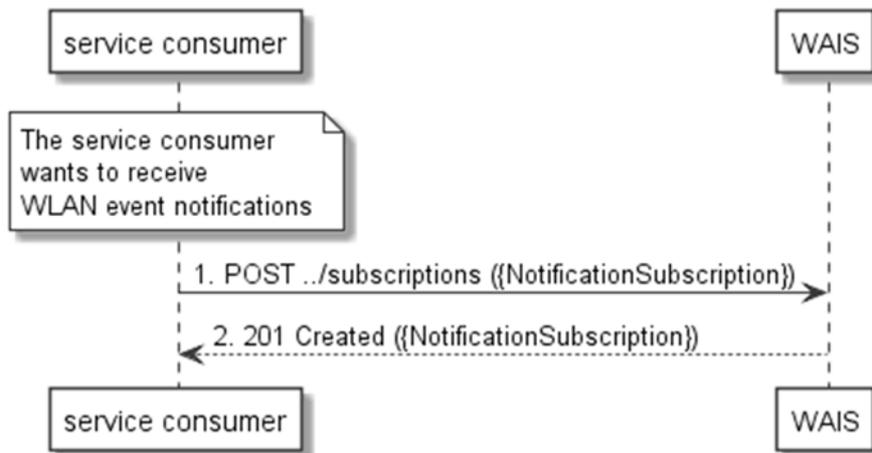
```
GET .../queries/sta?filter=(eq,apAssociated,mec404)&fields=staId,channelLoad
```

```
GET .../queries/sta?filter=(eq,staId,C8:D0:66:08:B6:0F)&fields=staId,channelLoad
```

## 5.2.4 REST based subscribe-notify model

### 5.2.4.1 Subscribing to WLAN event notifications

To receive notifications on selected WLAN events, the service consumer creates a subscription to certain specific event that is available at WAIS. Figure 5.2.4.1-1 shows a scenario where the service consumer uses REST based procedures to create a subscription for WLAN event notifications.



**Figure 5.2.4.1-1: Flow of subscribing to WLAN event notifications**

Subscribing to the WLAN event notifications, as illustrated in Figure 5.2.4.1-1, consists of the following steps.

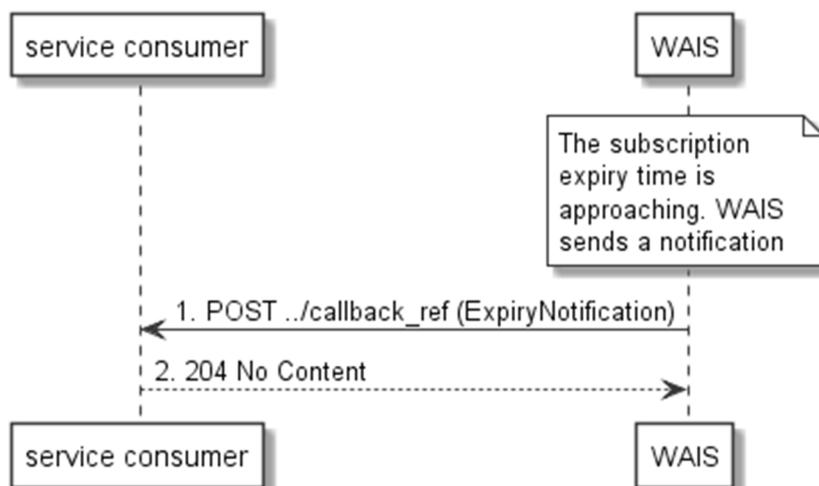
When the service consumer wants to receive notifications about the WLAN events, it creates a subscription to the WLAN event notifications:

- 1) The service consumer sends a POST request with the message body containing the {NotificationSubscription} data structure. The variable {NotificationSubscription} is replaced with the data type specified for different WLAN event subscriptions as specified in clauses 6.3.2 and 6.3.3, and it defines the subscribed event, the filtering criteria and the address where the service consumer wishes to receive the WLAN event notifications.
- 2) WAIS sends "201 Created" response with the message body containing the data structure specific to that WLAN event subscription. The data structure contains the address of the resource created and the subscribed WLAN event type.

#### 5.2.4.2 Receiving notification on expiry of WLAN event subscription

WAIS may define an expiry time for the WLAN event subscription. In case expiry time is used, the time will be included in the {NotificationSubscription} data structure that is included in the response message to the subscription. Prior the expiry, WAIS will also send a notification to the service consumer that owns the subscription.

Figure 5.2.4.2-1 shows a scenario where the service consumer receives a subscription expiry notification for the existing subscription.



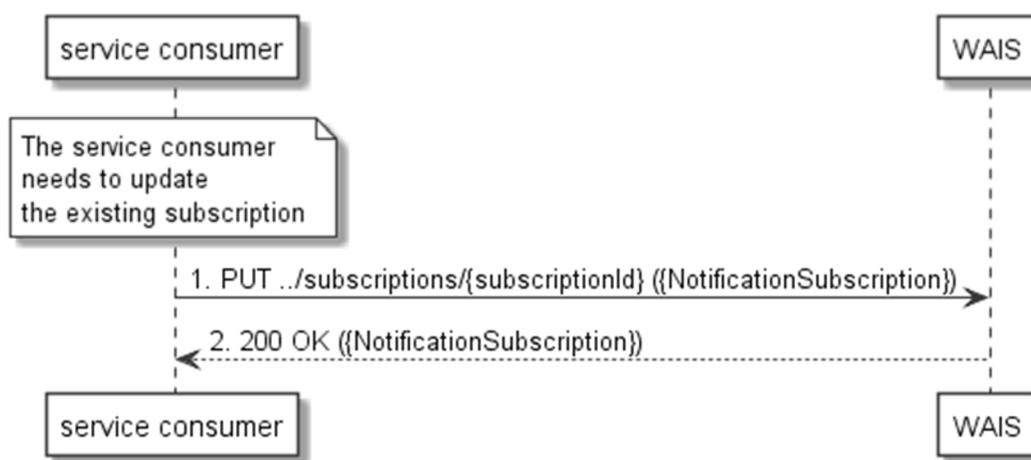
**Figure 5.2.4.2-1: Flow of WAIS sending a notification on expiry of the subscription**

Sending a notification on expiry of the subscription, as illustrated in Figure 5.2.4.2-1 consists of the following steps. If WAIS has defined an expiry time for the subscription, WAIS will send a notification prior the expiry:

- 1) WAIS sends a POST request to the callback reference address included by the service consumer in the subscription request. The POST request contains a data structure ExpiryNotification.
- 2) Service consumer sends a "204 No Content" response.

### 5.2.4.3 Updating subscription for WLAN event notifications

Figure 5.2.4.3-1 shows a scenario where the service consumer needs to update an existing subscription for a WLAN event notification. The subscription update is triggered e.g. by the need to change the existing subscription, or due to the expiry of the subscription.



**Figure 5.2.4.3-1: Flow of service consumer updating subscription for WLAN event notifications**

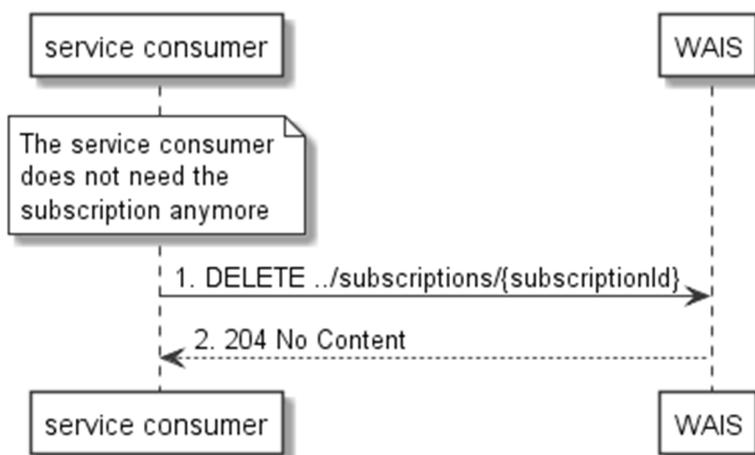
Updating subscription for WLAN event notifications, as illustrated in Figure 5.2.4.3-1, consists of the following steps.

When the service consumer needs to modify an existing subscription for WLAN event notifications, it can update the corresponding subscription as follows:

- 1) Service consumer updates the subscription resource by sending a PUT request to the resource containing all the subscriptions with the modified data structure specific to that WLAN event subscription.
- 2) WAIS returns "200 OK" with the message body containing the accepted data structure specific to that WLAN event subscription.

### 5.2.4.4 Unsubscribing from WLAN event notifications

When the service consumer does not want to receive notifications anymore after subscribing to WLAN events, the service consumer unsubscribes from the WLAN event notifications. Figure 5.2.4.4-1 shows a scenario where the service consumer uses REST based procedures to delete the subscription for WLAN event notifications.



**Figure 5.2.4.4-1: Flow of unsubscribing from the WLAN event notifications**

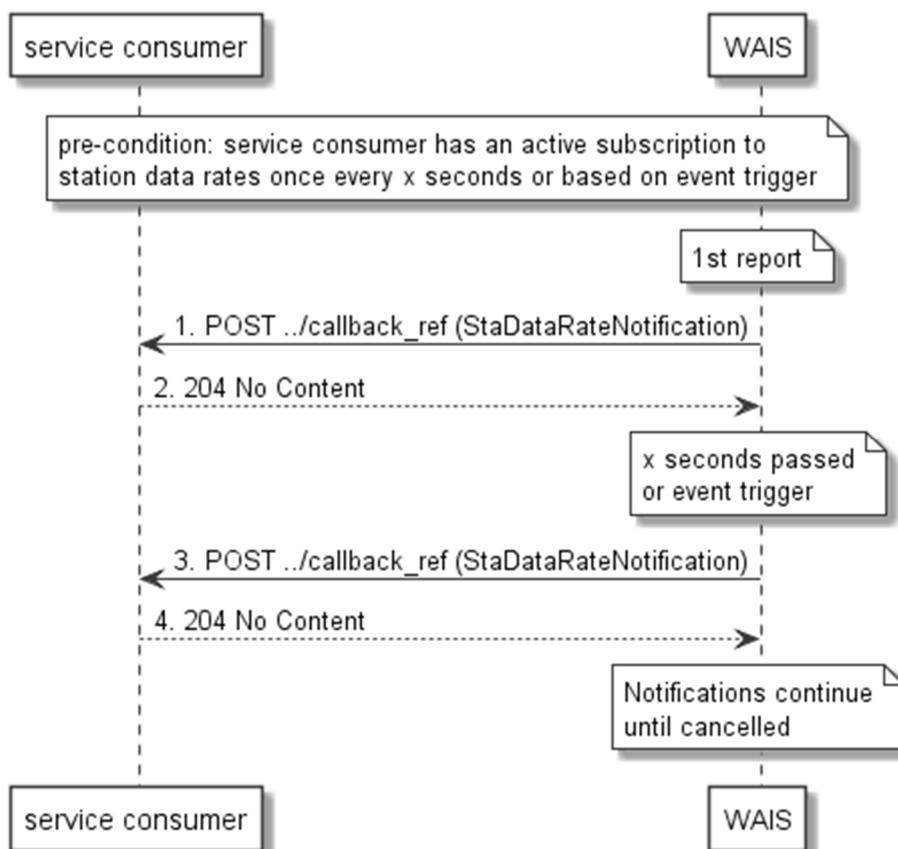
Unsubscribing from the WLAN event notifications, as illustrated in Figure 5.2.4.4-1, consists of the following steps.

When the service consumer does not want to receive the notifications anymore, it can unsubscribe from the WLAN notification events by deleting the subscription:

- 1) Service consumer sends a DELETE request to the resource representing the WLAN event subscription that was created.
- 2) WAIS sends "204 No content" response.

## 5.2.5 Receiving WLAN event notifications about station data rates

Figure 5.2.5-1 presents the scenario where the WAIS sends WLAN event notifications about WLAN station data rates, as defined in IEEE 802.11-2016 [8], to the service consumer.



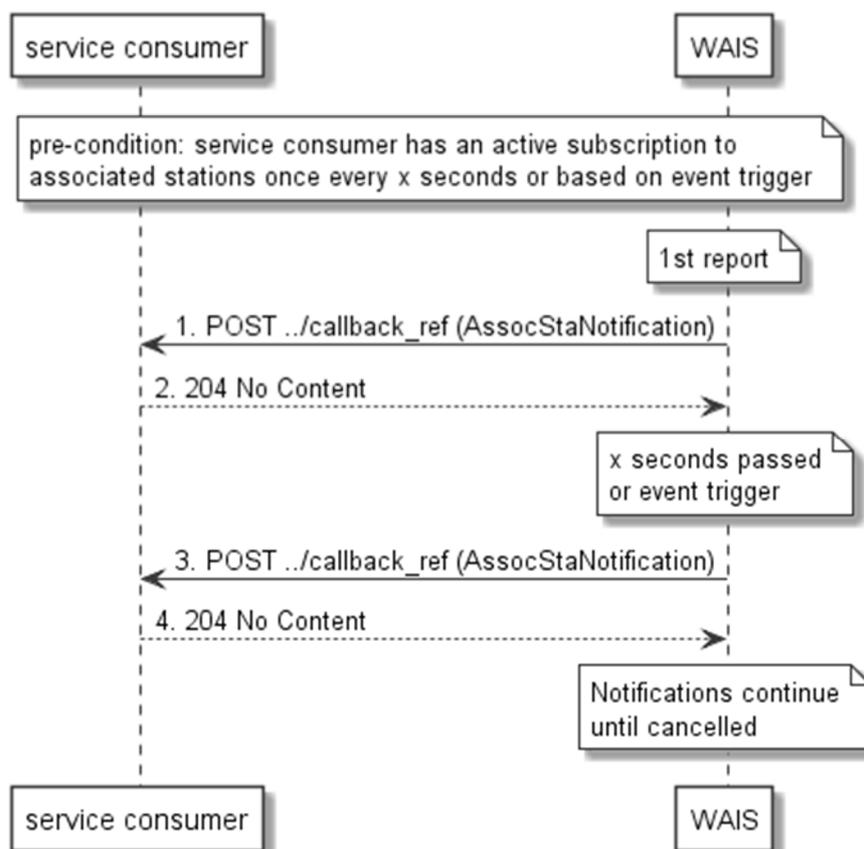
**Figure 5.2.5-1: Flow of receiving WLAN event notifications on station data rates**

Receiving WLAN event notifications on station data rates, as illustrated in Figure 5.2.5-1, consists of the following steps:

- 1) WAIS sends a POST request with the message body containing the StaDataRatesNotification data structure to the callback reference address included by the service consumer in the station data rates event subscription.
- 2) Service consumer sends a "204 No Content" response to the WAIS.

## 5.2.6 Receiving WLAN event notifications about associated stations

Figure 5.2.6-1 presents the scenario where the WAIS sends WLAN event notifications about WLAN stations that are associated with a particular access point, as defined in IEEE 802.11-2016 [8], whose information is requested.



**Figure 5.2.6-1: Flow of receiving WLAN event notifications on associated stations**

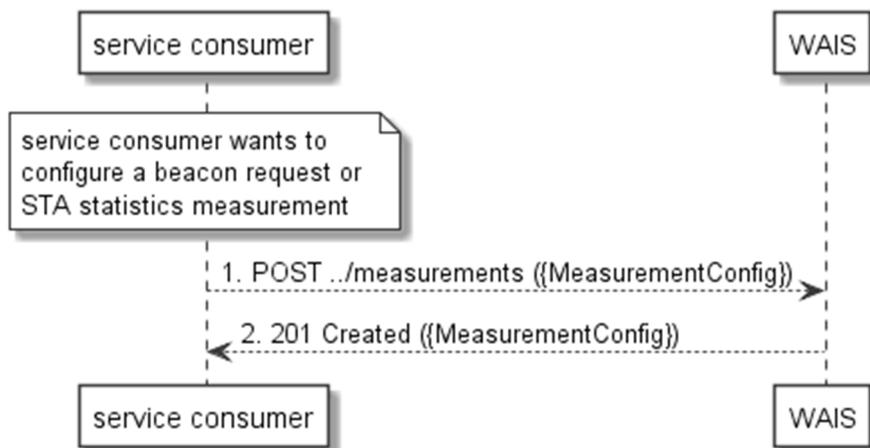
Receiving WLAN event notifications on stations associated with the access points, as illustrated in Figure 5.2.6-1, consists of the following steps:

- 1) WAIS sends a POST request with the message body containing the AssocStaNotification data structure to the callback reference address included by the service consumer in the WLAN event subscription.
- 2) Service consumer sends a "204 No Content" response to the WAIS.

## 5.2.7 Measurement Configuration

### 5.2.7.1 Creating a Measurement configuration

To configure specific characteristics of the Beacon Request or STA statistics measurements, the service consumer (e.g. a MEC application or a MEC platform) creates a new measurement configuration by providing (by a POST) the configuration and receiving an identifier of the configuration to be used in the measurement request. Figure 5.2.7.1-1 shows a scenario where the service consumer uses REST based procedures to create a new measurement configuration.



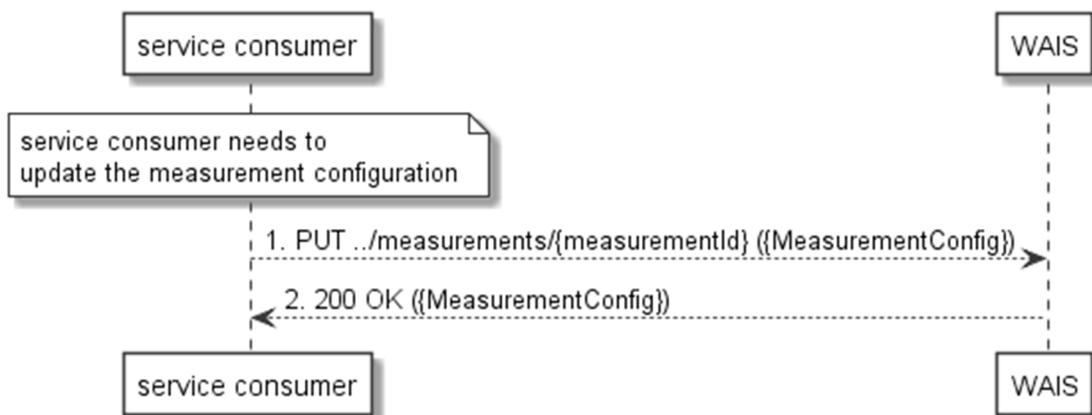
**Figure 5.2.7.1-1: Flow of service consumer creating a measurement configuration**

A service consumer requesting a certain measurement through the Beacon Request of STA statistics, consists of the following steps:

- 1) Service consumer configures measurement by creating a new measurement configuration MeasurementConfig, which includes measurementId, by sending a POST request to WAIS.
- 2) WAIS responds with a "201 Created", with the message body including the accepted MeasurementConfig structure.
- 3) Service consumer includes the measurementId in the attributes of the query to get specific information such as the staStatistics of the StaInfo resource data type or the apNeighbor of the ApInfo resource data type.

### 5.2.7.2 Updating a Measurement Configuration

Figure 5.2.7.2-1 shows a scenario where the service consumer needs to update an existing Measurement Configuration. The update is triggered e.g. by the need to change the existing measurement to a different channel or station.



**Figure 5.2.7.2-1: Flow of service consumer updating a measurement configuration**

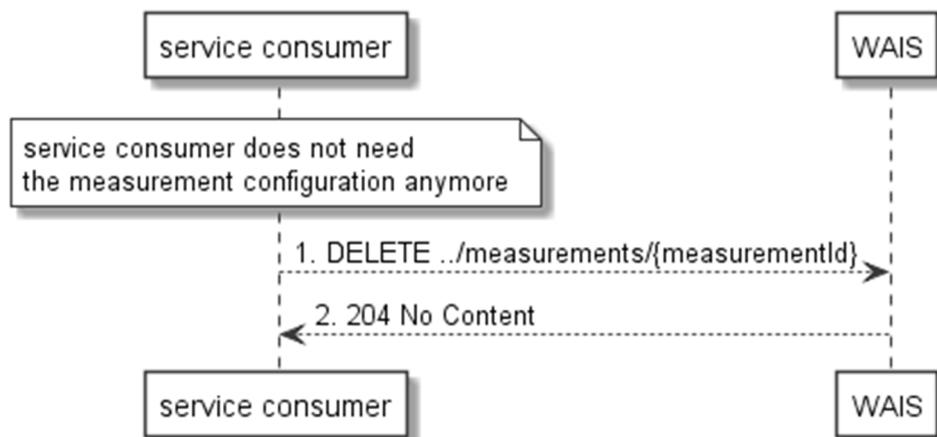
Updating subscription for WLAN event notifications, as illustrated in Figure 5.2.7.2-1, consists of the following steps.

When the service consumer needs to modify an existing measurement for WLAN, it can update the corresponding measurement as follows:

- 1) Service consumer updates the measurement configuration by sending a PUT request to the resource (i.e. MeasurementConfig) containing all the measurement configurations with the modified data structure specific to that WLAN measurement configuration.
- 2) WAIS returns "200 OK" with the message body containing the accepted data structure (i.e. MeasurementConfig) specific to that WLAN measurement configuration.

### 5.2.7.3 Deleting a Measurement Configuration

When the service consumer does not want to use a measurement configuration anymore, the service consumer DELETES the Measurement Configuration. Figure 5.2.7.3-1 shows a scenario where the service consumer uses REST based procedures to delete the Measurement Configuration.



**Figure 5.2.7.3-1: Flow of deletion of a Measurement Configuration**

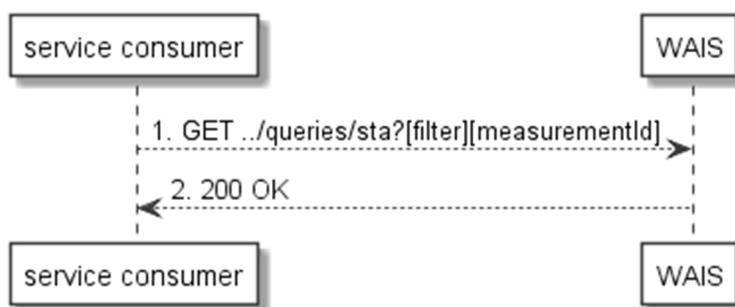
Deletion of a Measurement Configuration, as illustrated in Figure 5.2.7.3-1, consists of the following steps.

When the service consumer does not want to use the measurement configuration anymore, it can delete it:

- 1) Service consumer sends a DELETE request to the resource representing the WLAN measurement configuration that was created.
- 2) WAIS sends "204 No content" response.

### 5.2.7.4 Example of using a Measurement Configuration

Once configured, a measurementId can be used as an attribute to any query for measurement, such as ApInfo or StaInfo. In the following diagram, the flow that can be used for StaInfo is shown.



**Figure 5.2.7.4-1: Use of measurementId as a query attribute to StaInfo**

## 6 Data Model

### 6.1 General

The following clauses provide the description of the data model.

## 6.2 Resource data types

### 6.2.1 Introduction

This clause defines data structures that shall be used in resource representations.

### 6.2.2 Type: ApInfo

This type represents the information on Access Points available from the WLAN Access Information Service.

The attributes of the ApInfo shall follow the notations provided in Table 6.2.2-1.

**Table 6.2.2-1: Attributes for ApInfo**

Attribute name	Data type	Cardinality	Description
timeStamp	TimeStamp	0..1	Time stamp.
apId	ApIdentity	1	Identifier(s) to uniquely specify the Access Point whose information is exposed within this data type.
channel	Uint32	0..1	Channel configured for the Access Point.
wlanCap	WlanCapabilities	0..1	WLAN capabilities of Access Point.
wanMetrics	WanMetrics	0..1	WAN Metrics element provides information about the WAN link connecting an IEEE 802.11 Access Node and the Internet. Transmission characteristics such as the speed of the WAN connection to the Internet are included in Hotspot 2.0 Technical Specification v1.0.0 [i.3].
bssLoad	BssLoad	0..1	BSS Load attribute contains information on the current STA population and traffic levels in the BSS as defined in IEEE 802.11-2016 [8].
extBssLoad	ExtBssLoad	0..1	Extended BSS Load attribute contains more detailed information on the current STA population and traffic levels in the BSS as per ETSI GS MEC 002 [i.4].
oBssLoad	OBssLoad	0..1	Overlapping BSS Load attribute contains information related to the contribution of channel usage by Access Points in proximity to the reporting Access Point and operating on the same channel.
apLocation	ApLocation	0..1	The location on the Access Point.

### 6.2.3 Type: StaInfo

This type represents the information on wireless stations available from the WLAN Access Information Service.

The attributes of the StaInfo shall follow the notations provided in Table 6.2.3-1.

Table 6.2.3-1: Attributes for StaInfo

Attribute name	Data type	Cardinality	Description
timeStamp	TimeStamp	0..1	Time stamp.
stald	Stalidentity	1	Identifier(s) uniquely specify the station whose information is exposed within this data type.
channel	Uint32	0..1	Channel currently used by the station to connect with its associated Access Point.
apAssociated	ApAssociated	0..1	Information about the station's associated Access Point.
rsssi	Rssi	0..1	Receive Signal Strength Indicator.
staDataRate	StaDataRate	0..1	Station Data Rate as defined in IEEE 802.11-2016 [8].
staStatistics	StaStatistics	0..N	Statistics as defined in IEEE 802.11-2016 [8] for the client station collected over measurement duration.
beaconReport	BeaconReport	0..N	Beacon Report as defined in IEEE 802.11-2016 [8].
neighborReport	NeighborReport	0..N	Information about neighbor Access Points seen by the station as defined IEEE 802.11-2016 [8].
channelLoad	ChannelLoad	0..N	Channel Load reports as seen by the station as defined IEEE 802.11-2016 [8]. Channel Load reports may be configured for any channel, including the station's current channel for association.

## 6.2.4 Type: MeasurementConfig

This type represents the different measurement configurations available from the WLAN Access Information Service. Each measurement configuration is identified by a unique measurement ID. This ID is included in associated measurement reports for identification.

The attributes of the MeasurementConfig shall follow the notations provided in Table 6.2.4-1.

Table 6.2.4-1: Attributes for MeasurementConfig

Attribute name	Data type	Cardinality	Description
_links	Structure (inlined)	0..1	Hyperlink related to the resource.
>self	LinkType	1	Self-referring URI. The URI shall be unique within the WLAN Access Information API as it acts as an Id for the measurement configuration.
stald	Stalidentity	1..N	Identifier(s) to uniquely specify the target client station(s) for the measurement configuration.
measurementId	String	1	Unique identifier allocated by the service consumer to identify measurement reports (within sta_information query), associated with this measurement configuration.
measurementInfo	MeasurementInfo	1	Information used to configure this measurement.

## 6.2.5 Type: MeasurementConfigLinkList

This type represents a list of links related to existing measurement configurations for the service consumer. This information is returned when sending a request to receive current measurement configurations.

Table 6.2.5-1: Attributes of the MeasurementConfigLinkList

Attribute name	Data type	Cardinality	Description
_links	Structure (inlined)	1	Hyperlink related to the resource.
>self	LinkType	1	Self-referring URI.
measurementConfig	Structure (inlined)	0..N	
>href	Uri	1	The URI referring to a measurement configuration.
>measurementId	String	1	Unique identifier allocated by the service consumer to identify measurement reports associated with this measurement configuration.

## 6.3 Subscription data types

### 6.3.1 Introduction

This clause defines data structures for subscriptions.

### 6.3.2 Type: AssocStaSubscription

This type represents a subscription to get updates on client stations that are associated to an Access Point.

**Table 6.3.2-1: Attributes of the AssocStaSubscription**

Attribute name	Data type	Cardinality	Description
subscriptionType	String	1	Shall be set to "AssocStaSubscription".
callbackReference	Uri	0..1	URI selected by the service consumer to receive notifications on the subscribed WLAN Access Information. This shall be included both in the request and in response.  If not present, the service consumer is requesting the use of a Websocket for notifications. See note 1.
requestTestNotification	Boolean	0..1	Set to TRUE by the service consumer to request a test notification on the callbackReference URI to determine if it is reachable by the WAIS for notifications.
websocketNotifConfig	WebsocketNotifConfig	0..1	Provides details to negotiate and signal the use of a Websocket connection between the WAIS and the service consumer for notifications, either in place of the callbackReference URI or if it is not reachable via the test notification.
_links	Structure (inlined)	0..1	Hyperlink related to the resource. This shall be only included in the HTTP responses and in HTTP PUT requests.
>self	LinkType	1	Self-referring URI. The URI shall be unique within the WLAN Access Information API as it acts as an ID for the subscription.
apId	ApIdentity	1	Identifier(s) to uniquely specify the target Access Point for the subscription
notificationPeriod	UInt8	0..1	Set for periodic notification reporting. Value indicates the notification period in seconds.
notificationEvent	Structure (inline)	0..1	Set for trigger-based event notification reporting.
>trigger	Enum (inline)	1	Trigger for the notification: 1 = Notification issued when the number of connected stations is greater than or equal to the threshold. 2 = Notification issued when the number of connected stations is less than or equal to the threshold.
>threshold	UInt8	1	Number of connected stations threshold for trigger-based event reporting.
expiryDeadline	TimeStamp	0..1	The expiration time of the subscription determined by the WLAN Access Information Service.
NOTE 1: At least one of callbackReference and websocketNotifConfig shall be provided by the service consumer. If both are provided, it is up to WAIS to select the method to be used for notifications and to return only that method in the response.			
NOTE 2: AssocStaSubscription shall include either notificationPeriod or notificationEvent.			
NOTE 3: If both notificationPeriod and notificationEvent attributes are set, notifications are issued periodically when the trigger threshold is satisfied.			

### 6.3.3 Type: StaDataRateSubscription

This type represents a subscription to get updates on the Data Rate of targeted client station(s).

Table 6.3.3-1: Attributes of the StaDataRateSubscription

Attribute name	Data type	Cardinality	Description
subscriptionType	String	1	Shall be set to "StaDataRateSubscription".
callbackReference	Uri	0..1	URI selected by the service consumer to receive notifications on the subscribed WLAN Access Information Service. This shall be included both in the request and in response.  If not present, the service consumer is requesting the use of a Websocket for notifications. See note 1.
requestTestNotification	Boolean	0..1	Set to TRUE by the service consumer to request a test notification on the callbackReference URI to determine if it is reachable by the WAIS for notifications.
websocketNotifConfig	WebsocketNotifConfig	0..1	Provides details to negotiate and signal the use of a Websocket connection between the WAIS and the service consumer for notifications, either in place of the callbackReference URI or if it is not reachable via the test notification.
_links	Structure (inlined)	0..1	Hyperlink related to the resource. This shall be only included in the HTTP responses and in HTTP PUT requests.
>self	LinkType	1	Self-referring URI. The URI shall be unique within the WLAN Access Information API as it acts as an ID for the subscription.
stald	Stalidentity	1..N	Identifier(s) to uniquely specify the target client station(s) for the subscription.
notificationPeriod	Uint8	0..1	Set for periodic notification reporting. Value indicates the notification period in seconds.
notificationEvent	Structure (inline)	0..1	Set for trigger-based event notification reporting.
>trigger	Enum (inline)	1	Trigger event for the notification: 1 = Notification issued when the STA's downlink data rate is greater than or equal to the downlink threshold. 2 = Notification issued when the STA's downlink data rate is less than or equal to the downlink threshold. 3 = Notification issued when the STA's uplink data rate is greater than or equal to the uplink threshold. 4 = Notification issued when the STA's uplink data rate is less than or equal to the uplink threshold. 5 = Notification issued when the STA's downlink and uplink data rate is greater than or equal to their thresholds. 6 = Notification issued when the STA's downlink and uplink data rate is less than or equal to their thresholds. 7 = Notification issued when the STA's downlink or uplink data rate is greater than or equal to their thresholds. 8 = Notification issued when the STA's downlink or uplink data rate is less than or equal to their thresholds.
>downlinkRateThreshold	Uint32	0..1	Downlink data rate threshold for StaDataRate reporting.
>uplinkRateThreshold	Uint32	0..1	Uplink data rate threshold for StaDataRate reporting.
expiryDeadline	TimeStamp	0..1	The expiration time of the subscription determined by the WLAN Access Information Service.
NOTE 1: At least one of callbackReference and websocketNotifConfig shall be provided by the service consumer. If both are provided, it is up to WAIS to select the method to be used for notifications and to return only that method in the response.			
NOTE 2: StaDataRateSubscription shall include either notificationPeriod or notificationEvent.			
NOTE 3: If both notificationPeriod and notificationEvent attributes are set, notifications are issued periodically when the trigger threshold is satisfied.			

### 6.3.4 Type: SubscriptionLinkList

This type represents a list of links related to currently existing subscriptions for the service consumer. This information is returned when sending a request to receive current subscriptions.

**Table 6.3.4-1: Attributes of the SubscriptionLinkList**

Attribute name	Data type	Cardinality	Description
links	Structure (inlined)	1	List of hyperlinks related to the resource.
>self	LinkType	1	Self-referring URI.
subscription	Structure (inlined)	0..N	
>href	Uri	1	The URI referring to the subscription.
>subscriptionType	String	1	Type of the subscription. The string shall be set according to the "subscriptionType" attribute of the associated subscription data type defined in clauses 6.3.2, 6.3.3 and 6.3.5: <ul style="list-style-type: none"> <li>• "AssocStaSubscription"</li> <li>• "StaDataRateSubscription"</li> <li>• "MeasurementReportSubscription".</li> </ul>

### 6.3.5 Type: MeasurementReportSubscription

This type represents a subscription to get measurement reports (Channel Load, Beacon Request, STA Statistics, or Neighbor Report) from targeted client station(s).

**Table 6.3.5-1: Attributes of the MeasurementReportSubscription**

Attribute name	Data type	Cardinality	Description
subscriptionType	String	1	Shall be set to "MeasurementReportSubscription".
callbackReference	Uri	0..1	URI selected by the service consumer to receive notifications on the subscribed WLAN Access Information Service. This shall be included both in the request and in response.  If not present, the service consumer is requesting the use of a Websocket for notifications. See note.
requestTestNotification	Boolean	0..1	Set to TRUE by the service consumer to request a test notification on the callbackReference URI to determine if it is reachable by the WAIS for notifications.
websocketNotifConfig	WebsocketNotifConfig	0..1	Provides details to negotiate and signal the use of a Websocket connection between the WAIS and the service consumer for notifications, either in place of the callbackReference URI or if it is not reachable via the test notification.
_links	Structure (inlined)	0..1	Hyperlink related to the resource. This shall be only included in the HTTP responses and in HTTP PUT requests.
>self	LinkType	1	Self-referring URI. The URI shall be unique within the WLAN Access Information API as it acts as an ID for the subscription.
stald	Stalidentity	1..N	Identifier(s) to uniquely specify the target client station(s) for the subscription.
measurementId	String	1	Unique identifier allocated by the service consumer to identify measurement reports associated with this measurement subscription.
measurementInfo	MeasurementInfo	1	Information used to configure this measurement.
expiryDeadline	TimeStamp	0..1	The expiration time of the subscription determined by the WLAN Access Information Service.
NOTE:	At least one of callbackReference and websocketNotifConfig shall be provided by the service consumer. If both are provided, it is up to WAIS to select the method to be used for notifications and to return only that method in the response.		

### 6.3.6 Type: WebsocketNotifConfig

This type represents configuration for the delivery of subscription notifications over Websockets per the pattern defined in defined in clause 6.12a of ETSI GS MEC 009 [10].

**Table 6.3.6-1: Attributes of the WebsockNotifConfig**

Attribute name	Data type	Cardinality	Description
websocketUri	Uri	0..1	Set by WAIS to indicate to the service consumer the WebSocket URI to be used for delivering notifications.
requestWebSocketUri	Boolean	0..1	Set to true by the service consumer to indicate that WebSocket delivery is requested.

## 6.4 Notifications data types

### 6.4.1 Introduction

This clause defines data structures that define notifications.

### 6.4.2 Type: AssocStaNotification

This type represents a notification from WLAN Access Information Service with regards to client stations associated to the targeted Access Point.

The attributes of the AssocStaNotification shall follow the indications provided in Table 6.4.2-1.

**Table 6.4.2-1: Attributes of the AssocStaNotification**

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "AssocStaNotification".
timeStamp	TimeStamp	0..1	Time stamp.
apId	ApIdentity	1	Identifier(s) to uniquely specify the Access Point to which the client stations are associated.
stald	StalIdentity	0..N	Identifier(s) to uniquely specify the client station(s) associated.

### 6.4.3 Type: StaDataRateNotification

This type represents a notification from WLAN Access Information service with regards to Data Rates of the subscribed client stations.

The attributes of the StaDataRateNotification shall follow the indications provided in Table 6.4.3-1.

**Table 6.4.3-1: Attributes of the StaDataRateNotification**

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "StaDataRateNotification".
timeStamp	TimeStamp	0..1	Time stamp.
staDataRate	StaDataRate	0..N	Data rates of a client station.

### 6.4.4 Type: ExpiryNotification

This type represents a notification from WLAN Access Information service with regards to expiry of an existing subscription.

The Notification is sent by the WLAN Access Information service to send information about expiry of a subscription.

**Table 6.4.4-1: Attributes of the ExpiryNotification**

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "ExpiryNotification".
_links	Structure (inlined)	1	Hyperlink related to the resource.
>subscription	LinkType	1	URI identifying the subscription which has expired.
expiryDeadline	TimeStamp	1	Time stamp.

## 6.4.5 Type: MeasurementReportNotification

This type represents a notification from WLAN Access Information service with regards to Measurement Reports of the subscribed client stations.

The attributes of the MeasurementReportNotification shall follow the indications provided in Table 6.4.5-1.

**Table 6.4.5-1: Attributes of the MeasurementReportNotification**

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "MeasurementReportNotification".
timeStamp	TimeStamp	0..1	Time stamp of the notification.
staStatistics	StaStatistics	0..N	STA Statistics Report as defined in IEEE 802.11-2016 [8].
beaconReport	BeaconReport	0..N	Beacon Report as defined in IEEE 802.11-2016 [8].
neighborReport	NeighborReport	0..N	Neighbor Report providing information about neighbor Access Points seen by the station as defined in IEEE 802.11-2016 [8].
channelLoad	ChannelLoad	0..N	Channel Load reports as seen by the station as defined in IEEE 802.11-2016 [8].
NOTE 1: Each MeasurementReportNotification shall include a single measurement report type (channelLoad, beaconRequest, etc.), based on the MeasurementReportSubscription.			
NOTE 2: A single MeasurementReportNotification may include multiple measurement reports (of the same type), if the MeasurementReportSubscription configured the measurement report for several client stations and multiple reports are available to WAIS.			

## 6.4.6 Type: TestNotification

This type represents a test notification from WLAN Access Information service to determine if the Websocket method is to be utilized for the WAIS to issue notifications for a subscription, as defined in clause 6.12a of ETSI GS MEC 009 [10].

**Table 6.4.6-1: Attributes of the TestNotification**

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "TestNotification".
_links	Structure (inlined)	1	Hyperlink related to the resource.
>subscription	LinkType	1	URI identifying the subscription for the test notification.

## 6.5 Referenced structured data types

### 6.5.1 Introduction

This clause defines data structures that are referenced from data structures defined in the previous clauses, but are neither resource representations nor bound to any pub/sub mechanism.

## 6.5.2 Type: TimeStamp

This type represents a time stamp.

**Table 6.5.2-1: Attributes of the TimeStamp**

Attribute name	Data type	Cardinality	Description
seconds	Uint32	1	The seconds part of the time. Time is defined as Unix-time since January 1, 1970, 00:00:00 UTC.
nanoSeconds	Uint32	1	The nanoseconds part of the time. Time is defined as Unix-time since January 1, 1970, 00:00:00 UTC.

## 6.5.3 Type: Aplidentity

This type represents identifiers determining a specific Access Point.

**Table 6.5.3-1: Attributes of the Aplidentity**

Attribute name	Data type	Cardinality	Description
bssid	String	1	Basic Service Set Identifier (BSSID) is a unique Identifier assigned to an Access Point (as network interface controller) for communications at the data link layer of a network segment. BSSID is typically set to an access point's MAC address.
ssid	String	0..N	Service Set Identifier (SSID) to identify logical WLAN networks available via the Access Point.
ipAddress	String	0..N	IPv4 or IPv6 address allocated for the Access Point.

## 6.5.4 Type: WlanCapabilities

This type represents the WLAN capabilities of the Access Point.

**Table 6.5.4-1: Attributes of the WlanCapabilities**

Attribute name	Data type	Cardinality	Description
ht	HtCapabilities	0..1	Information about Access Point HT capabilities as defined in IEEE 802.11-2016 [8].
vht	VhtCapabilities	0..1	Information about Access Point VHT capabilities as defined in IEEE 802.11-2016 [8].
he	HeCapabilities	0..1	Information about Access Point HE capabilities as defined in IEEE P802.11ax [i.10] section 9.2.2.248.
dmg	DmgCapabilities	0..1	Information about Access Point DMG capabilities as defined in IEEE 802.11-2016 [8].
edmg	EdmgCapabilities	0..1	Information about Access Point EDMG capabilities as defined in draft IEEE P802.11ay [i.11].

## 6.5.5 Void

## 6.5.6 Type: WanMetrics

This type represents the metrics related to the backhaul characteristics of an Access Point as defined for WAN metrics in Hotspot 2.0 Technical Specification v1.0.0 [i.3].

Table 6.5.6-1: Attributes of the WanMetrics

Attribute name	Data type	Cardinality	Description
wanInfo	Uint8	1	Info about WAN link status, link symmetricity and capacity currently used.
downlinkSpeed	Uint32	1	4-octet positive integer whose value is an estimate of the WAN Backhaul link current downlink speed in kilobits per second.
uplinkSpeed	UInt32	1	4-octet positive integer whose value is an estimate of the WAN Backhaul link's current uplink speed in kilobits per second.
downlinkLoad	UInt8	1	1-octet positive integer representing the current percentage loading of the downlink WAN connection, scaled linearly with 255 representing 100 %, as measured over an interval the duration of which is reported in Load Measurement Duration. In cases where the downlink load is unknown to the AP, the value is set to zero.
uplinkLoad	Uint8	1	1-octet positive integer representing the current percentage loading of the uplink WAN connection, scaled linearly with 255 representing 100 %, as measured over an interval, the duration of which is reported in Load Measurement Duration. In cases where the uplink load is unknown to the AP, the value is set to zero.
lmd	Uint16	1	The LMD (Load Measurement Duration) field is a 2-octet positive integer representing the duration over which the Downlink Load and Uplink Load have been measured, in tenths of a second. When the actual load measurement duration is greater than the maximum value, the maximum value will be reported. The value of the LMD field is set to 0 when neither the uplink nor downlink load can be computed. When the uplink and downlink loads are computed over different intervals, the maximum interval is reported.

### 6.5.7 Type: BssLoad

This type represents the load of a BSS as defined in section 9.4.2.28, BSS Load element, within IEEE 802.11-2016 [8].

Table 6.5.7-1: Attributes of the BssLoad

Attribute name	Data type	Cardinality	Description
staCount	UInt16	1	An unsigned integer that indicates the total number of STAs currently associated with this BSS.
channelUtilization	UInt8	1	The percentage of time, linearly scaled with 255 representing 100 %, that the AP sensed the medium was busy, as indicated by either the physical or virtual Carrier Sense (CS) mechanism.
availAdmCap	Uint16	1	Available Admission Capacity that specifies the remaining amount of medium time available via explicit admission control, in units of 32 $\mu$ s/s.

### 6.5.8 Type: ExtBssLoad

This type represents the Extended BSS Load information as defined in section 9.4.2.160, Extended BSS Load element, within IEEE 802.11-2016 [8].

Table 6.5.8-1: Attributes of the ExtBssLoad

Attribute name	Data type	Cardinality	Description
muMimoStaCount	UInt16	1	Indicates the total number of STAs currently associated with this BSS that have a 1 in the MU Beamformee Capable field of their VHT Capabilities element.
spatStreamUnderUtil	UInt8	1	The percentage of time, linearly scaled with 255 representing 100 %, that the AP has underutilized spatial domain resources for given busy time of the medium.
obsSec20MhzUtil	Uint8	1	Observable loading on each of the secondary 20 MHz channel.
obsSec40MhzUtil	Uint8	1	Observable loading on each of the secondary 40 MHz channel.
obsSec80MhzUtil	Uint8	1	Observable loading on each of the secondary 80 MHz channel.

## 6.5.9 Type: ApLocation

This type represents the location information of the Access Point as defined in IEEE 802.11-2016 [8].

**Table 6.5.9-1: Attributes of the ApLocation**

Attribute name	Data type	Cardinality	Description
geolocation	GeoLocation	0..1	Geospatial Location of the AP as defined in IEEE 802.11-2016 [8].
civicLocation	CivicLocation	0..1	Civic Location of the AP as described in IETF RFC 4776 [7].

## 6.5.10 Type: NeighborReport

This type represents the information about neighbor Access Points as defined in section 9.4.2.37, Neighbor Report element, within IEEE 802.11-2016 [8].

**Table 6.5.10-1: Attributes of the NeighborReport**

Attribute name	Data type	Cardinality	Description
stald	Stalidentity	0..1	Identifier to uniquely specify the station whose information is exposed within this report. If this report is contained within a data type that provides the station's identifier, this field may be omitted.
measurementId	String	1	Measurement ID of the Measurement configuration applied to this Neighbor Report.
bssid	String	1	BSSID (MAC address) of the Access Point that is being reported.
bssidInfo	BssidInfo	1	Additional information related to Access Point that is being reported such as AP reachability, security, key scope, Mobility Domain, HT/VHT capability and Fine Time Measurements, as defined in Figure 9-296, BSSID information field, within IEEE 802.11-2016 [8].
operatingClass	Uint8	1	Operating Class field indicates an operating class value as defined in Annex E within IEEE 802.11-2016 [8].
channel	Uint8	1	Channel field indicates a channel number, which is interpreted in the context of the indicated operating class. Channel numbers are defined in Annex E within IEEE 802.11-2016 [8].
phyType	Uint8	1	PHY type of the AP indicated by this BSSID. It is an integer value coded according to the value of the dot11PHYType, Annex C within IEEE 802.11-2016 [8]. <ul style="list-style-type: none"> <li>• 2 = dsss</li> <li>• 4 = ofdm</li> <li>• 5 = hrdsss</li> <li>• 6 = erp</li> <li>• 7 = ht</li> <li>• 8 = dmrg</li> <li>• 9 = vht</li> <li>• 10 = tvht</li> </ul>
bssTransitionCandidate Preference	Uint8	0..1	Relative value indicating the preferred ordering for this BSS as a transition candidate for roaming. 255 indicating the most preferred candidate and 1 indicating the least preferred candidate, as defined in Table 9-152 within IEEE 802.11-2016 [8].

### 6.5.11 Type: StalIdentity

This type represents identifiers determining a specific client station.

**Table 6.5.11-1: Attributes of the StalIdentity**

Attribute name	Data type	Cardinality	Description
macId	String	1	Unique identifier assigned to station (as network interface controller) for communications at the data link layer of a network segment.
ssid	String	0..N	Service Set Identifier(s) to identify logical networks.
aid	String	0..1	Number which identifies a particular association between a station and an Access Point
ipAddress	String	0..N	IPv4 or IPv6 address(es) allocated for the station.

### 6.5.12 Type: ApAssociated

This type represents information for the Access Point that the client station is associated to.

**Table 6.5.12-1: Attributes of the ApAssociated**

Attribute name	Data type	Cardinality	Description
bssid	String	1	Basic Service Set Identifier (BSSID) is a unique identifier assigned to the Access Point (as network interface controller) for communications at the data link layer of a network segment. BSSID is typically set to an access point's MAC address.
ssid	String	0..N	Service Set Identifier to identify logical networks.
assocId	String	0..1	Unique number which identifies a particular association between the station and Access Point.
ipAddress	String	0..N	IPv4 or IPv6 address allocated for the Access Point.

### 6.5.13 Type: StaStatistics

This type represents information statistics of the client station as defined in section 9.4.2.22.9, STA Statistics Report, within IEEE 802.11-2016 [8].

**Table 6.5.13-1: Attributes of the StaStatistics**

Attribute name	Data type	Cardinality	Description
stalId	StalIdentity	0..1	Identifier to uniquely specify the station whose information is exposed within this report. If this report is contained within a data type that provides the station's identifier, this field may be omitted.
measurementId	String	1	Measurement ID of the Measurement configuration applied to this STA Statistics Report.
measurementDuration	UInt16	1	Duration over which the Statistics Group Data was measured in time units of 1 024 $\mu$ s. Duration equal to zero indicates a report of current values.
groupIdentity	UInt8	1	Indicates the requested statistics group describing the Statistics Group Data according to Table 9-114 of IEEE 802.11-2016 [8]. Depending on group identity, one and only one of the STA Statistics Group Data will be present.
groupZeroData	StaStatisticsGroupZeroData	0..1	STA Statistics Data for Group Identity = 0
groupOneData	StaStatisticsGroupOneData	0..1	STA Statistics Data for Group Identity = 1
group2to9Data	StaStatisticsGroup2to9Data	0..1	STA Statistics Data for Group Identity = 2 through 9

## 6.5.14 Type: HtCapabilities

This type represents information on HT capabilities of an Access Point as defined in IEEE 802.11-2016 [8].

**Table 6.5.14-1: Attributes of the HtCapabilities**

Attribute name	Data type	Cardinality	Description
htCapabilityInfo	UInt16	1	HT Capability Information as defined in IEEE 802.11-2016 [8].
ampduParameters	UInt8	1	A-MPDU parameters as defined in IEEE 802.11-2016 [8].
supportedMcsSet	UInt128	1	Supported MCS set as defined in IEEE 802.11-2016 [8].
htExtendedCap	UInt16	1	Extended HT Capabilities as defined in IEEE 802.11-2016 [8].
txBeamFormCap	UInt32	1	Transmit Beamforming Capabilities as defined in IEEE 802.11-2016 [8].
aselCap	UInt8	1	ASEL capabilities as defined in IEEE 802.11-2016 [8].

## 6.5.15 Type: VhtCapabilities

This type represents information on VHT Capabilities of an Access Point as defined in IEEE 802.11-2016 [8].

**Table 6.5.15-1: Attributes of the VhtCapabilities**

Attribute name	Data type	Cardinality	Description
vhtCapInfo	UInt32	1	VHT capabilities Info as defined in IEEE 802.11-2016 [8].
vhtMcsNss	UInt64	1	Supported VHT-MCS and NSS Set as defined in IEEE 802.11-2016 [8].

## 6.5.16 Type: HeCapabilities

This type represents information on HE Capabilities of an Access Point as defined in draft IEEE P802.11ax [i.10].

**Table 6.5.16-1: Attributes of the HeCapabilities**

Attribute name	Data type	Cardinality	Description
heMacCapInfo	UInt8	1	MAC capabilities of an Access Point.
hePhyCapInfo	UInt8	1	PHY capabilities of an Access Point.
supportedHeMcsNssSet	UInt8	1	Supported MCS and NSS Set.

## 6.5.17 Type: DmgCapabilities

This type represents information on DMG Capabilities of an Access Point as defined in IEEE 802.11-2016 [8].

**Table 6.5.17-1: Attributes of the DmgCapabilities**

Attribute name	Data type	Cardinality	Description
dmgStaCapInfo	UInt64	1	DMG station capabilities information as defined in IEEE 802.11-2016 [8].
dmgApOrPcpCapInfo	UInt16	1	DMG AP or PCP capabilities information as defined in IEEE 802.11-2016 [8].
dmgStaBeamTrackTimeLimit	UInt16	1	DMG station beam tracking time limit as defined in IEEE 802.11-2016 [8].
ExtScMcsCap	UInt8	1	Extended SC MCS capabilities as defined in IEEE 802.11-2016 [8].
maxNrBasicAmsduSubframes	UInt8	1	Number of basic A-MSDU subframes in A-MSDU as defined in IEEE 802.11-2016 [8].
maxNrShortAmsduSubframes	UInt8	1	Number of short A-MSDU subframes in A-MSDU as defined in IEEE 802.11-2016 [8].

## 6.5.18 Type: EdmgCapabilities

This type represents information on EDMG Capabilities of an Access Point as defined in draft IEEE P802.11ay [i.11].

**Table 6.5.18-1: Attributes of the EdmgCapabilities**

Attribute name	Data type	Cardinality	Description
ampduParameters	Uint8	1	A-MPDU parameters as defined in draft IEEE P802.11ay [i.11]
trnParameters	Uint16	1	Training parameters as defined in draft IEEE P802.11ay [i.11]
supportedMcs	Uint32	1	Supported MCS as defined in draft IEEE P802.11ay [i.11]

## 6.5.19 Type: GeoLocation

This type represents information Geospatial Location of an Access Point as defined in IEEE 802.11-2016 [8] and in IETF RFC 6225 [6].

**Table 6.5.19-1: Attributes of the GeoLocation**

Attribute name	Data type	Cardinality	Description
latUncertainty	Uint8	1	The uncertainty for Latitude information as defined in IETF RFC 6225 [6]
lat	Uint64	1	The latitude value of location as defined in IETF RFC 6225 [6]
longUncertainty	Uint8	1	The uncertainty for Longitude information as defined in IETF RFC 6225 [6]
long	Uint64	1	The longitude value of location as defined in IETF RFC 6225 [6]
altitudeType	Uint8	0..1	The type description for altitude information e.g. floors or meters as defined in IETF RFC 6225 [6]
altitudeUncertainty	Uint8	0..1	The uncertainty for altitude information as defined in IETF RFC 6225 [6]
altitude	Uint32	0..1	The altitude value of location as defined in IETF RFC 6225 [6]
datum	Uint8	1	The datum value to express how coordinates are organized and related to real world as defined in IETF RFC 6225 [6]

## 6.5.20 Type: CivicLocation

This type represents information on Civic Location of an Access Point as defined in IETF RFC 4776 [7].

Table 6.5.20-1: Attributes of the CivicLocation

Attribute name	Data type	Cardinality	Description
country	String	1	The two-letter ISO 3166 [i.9] country code in capital ASCII letters, e.g. DE or US, as per ISO 3166 [i.9]
ca0	String	0..1	Language
ca1	String	0..1	National subdivisions (state, canton, region, province, prefecture)
ca2	String	0..1	County, parish, gun (JP), district (IN)
ca3	String	0..1	City, township, shi (JP)
ca4	String	0..1	City division, borough, city district, ward, chou (JP)
ca5	String	0..1	Neighborhood, block
ca6	String	0..1	Group of streets below the neighborhood level
ca16	String	0..1	Leading street direction
ca17	String	0..1	Trailing street suffix
ca18	String	0..1	Street suffix or type
ca19	String	0..1	House number
ca20	String	0..1	House number suffix
ca21	String	0..1	Landmark of vanity address
ca22	String	0..1	Additional location information
ca23	String	0..1	Name (residence and office occupant)
ca24	String	0..1	Postal/zip code
ca25	String	0..1	Building (structure)
ca26	String	0..1	Unit (apartment/suite)
ca27	String	0..1	Floor
ca28	String	0..1	Room
ca29	String	0..1	Type of place
ca30	String	0..1	Postal community name
ca31	String	0..1	Post office box
ca32	String	0..1	Additional code
ca33	String	0..1	Seat (desk.cubicle, workstation)
ca34	String	0..1	Primary road name
ca35	String	0..1	Road section
ca36	String	0..1	Branch road name
ca37	String	0..1	Sub-branch road name
ca38	String	0..1	Street name pre-modifier
ca39	String	0..1	Street name post-modifier
ca128	String	0..1	Script

## 6.5.21 Type: Rssi

This type represents information on the Received Signal Strength Indicator (RSSI) of a client station as defined in IEEE 802.11-2016 [8].

Table 6.5.21-1: Attributes of the Rssi

Attribute name	Data type	Cardinality	Description
rssi	Uint8	1	The Received Signal Strength Indicator from a station

## 6.5.22 Type: StaDataRate

This type represents the data rates of a client station as defined in Data Elements Specification v1.0 [i.8].

**Table 6.5.22-1: Attributes of the StaDataRate**

Attribute name	Data type	Cardinality	Description
stald	Stalidentity	0..1	Identifier to uniquely specify the station whose information is exposed within this report. If this report is contained within a data type that provides the station's identifier, this field may be omitted.
staLastDataDownlinkRate	Uint32	0..1	The data transmit rate in kbps that was most recently used for transmission of data PPDU from the access point to the station.
staLastDataUplinkRate	Uint32	0..1	The data transmit rate in Kbps that was most recently used for transmission of data PPDU from the associated station to the access point.

NOTE: StaDataRate shall include at least one instance of either downlink rate or uplink rate and may include both.

## 6.5.23 Type: LinkType

This type represents a type of link.

**Table 6.5.23-1: Attributes of the LinkType**

Attribute name	Data type	Cardinality	Description
href	Uri	1	URI referring to a resource

## 6.5.24 Type: ChannelLoadConfig

This configuration applies to Channel Load measurement as described in clause 6.5.39 in the present document. This configuration follows the format as described in section 9.4.2.21.5, Channel Load request, within IEEE 802.11-2016 [8].

**Table 6.5.24-1: Attributes of the ChannelLoadConfig**

Attribute name	Data type	Cardinality	Description
operatingClass	Uint8	1	Operating Class field indicates an operating class value as defined in Annex E within IEEE 802.11-2016 [8].
channel	Integer	1	Channel for which the channel load report is requested.
reportingCondition	Uint8	0..1	Reporting condition for the Beacon Report as per Table 9-153 of IEEE 802.11-2016 [8]: 0 = Report to be issued after each measurement. 1 = Report to be issued when Channel Load is greater than or equal to the threshold. 2 = Report to be issued when Channel Load is less than or equal to the threshold.  If this optional field is not provided, channel load report should be issued after each measurement (reportingCondition = 0).
threshold	Uint8	0..1	Channel Load reference value for threshold reporting. This field shall be provided for reportingCondition values 1 and 2.

## 6.5.25 Type: BeaconRequestConfig

This configuration applies to the BeaconReport as described in clause 6.5.27 of the present document. This configuration follows the format as described in section 9.4.2.21.7, Beacon request, within IEEE 802.11-2016 [8].

**Table 6.5.25-1: Attributes of the BeaconRequestConfig**

Attribute name	Data type	Cardinality	Description
operatingClass	Uint8	1	Operating Class field indicates an operating class value as defined in Annex E within IEEE 802.11-2016 [8].
channelId	Uint8	1	Channel number to scan. A Channel Number of 0 indicates a request to make iterative measurements for all supported channels in the Operating Class where the measurement is permitted on the channel and the channel is valid for the current regulatory domain. A Channel Number of 255 indicates a request to make iterative measurements for all supported channels in the current Operating Class listed in the latest AP Channel Report received from the serving AP.
measurementMode	Uint8	1	0 for passive. 1 for active. 2 for beacon table.
bssid	String	0..1	The BSSID field indicates the BSS for which a beacon report is requested. If absent, the requested beacon reports should include all BSSs on the channel.
ssid	String	0..1	The SSID subelement indicates the ESS or IBSS for which a beacon report is requested.
beaconReportingConf	BeaconReportingConfig	1	Beacon reporting configuration data field format as in Figure 9-157 in IEEE 802.11-2016 [8].

## 6.5.26 Type: StaStatisticsConfig

This configuration applies to the StaStatistics as described in clause 6.5.13 of the present document. This configuration references the format as described in section 9.4.2.21.9, STA Statistics request, within IEEE 802.11-2016 [8].

**Table 6.5.26-1: Attributes of the StaStatisticsConfig**

Attribute name	Data type	Cardinality	Description
groupIdentity	Uint8	1	As per Table 9-92 of IEEE 802.11-2016 [8].
triggeredReport	Boolean	1	True = triggered reporting, otherwise duration.
measurementCount	Uint32	0..1	Valid if triggeredReport = true. Specifies the number of MAC service data units or protocol data units to determine if the trigger conditions are met.
triggerTimeout	Uint16	0..1	Valid if triggeredReport = true. The Trigger Timeout field contains a value in units of 100 time-units of 1 024 $\mu$ s during which a measuring STA does not generate further triggered STA Statistics Reports after a trigger condition has been met.
triggerCondition	STACounterTriggerCondition	0..1	Valid if triggeredReport = true. As per Figure 9-161 of IEEE 802.11-2016 [8]. Defines what are the metrics returned by the STA Statistics Report.

## 6.5.27 Type: BeaconReport

This type represents information in a STA Beacon report as defined in section 9.4.2.22.7, Beacon Report, within IEEE 802.11-2016 [8].

Table 6.5.27-1: Attributes of the BeaconReport

Attribute name	Data type	Cardinality	Description
stald	StalIdentity	0..1	Identifier to uniquely specify the station whose information is exposed within this report. If this report is contained within a data type that provides the station's identifier, this field may be omitted.
measurementId	String	1	Measurement ID of the Measurement configuration applied to this Beacon Report.
operatingClass	Uint8	1	Operating Class field indicates an operating class value as defined in Annex E within IEEE 802.11-2016 [8].
channel	Uint8	1	Channel number where the beacon was received.
reportedFrameInfo	ReportedBeaconFrameInfo	1	Information about the reported beacon frame
bssid	String	1	Indicates the BSSID of the BSS for which a beacon report has been received.
ssid	String	0..1	The SSID subelement indicates the ESS or IBSS for which a beacon report is received.
rcpi	Uint8	0..1	RCPI indicates the received channel power of the Beacon, Measurement Pilot, or Probe Response frame, which is a logarithmic function of the received signal power, as defined in section 9.4.2.38 of IEEE 802.11-2016 [8].
rsni	Uint8	0..1	RSNI indicates the received signal-to-noise indication for the Beacon, Measurement Pilot, or Probe Response frame, as described in section 9.4.2.41 of IEEE 802.11-2016 [8].
antennaId	Uint8	0..1	The Antenna ID field contains the identifying number for the antenna(s) used for this measurement. Antenna ID is defined in section 9.4.2.40 of IEEE 802.11-2016 [8].
parentTsf	Uint32	0..1	The Parent TSF field contains the lower 4 octets of the measuring STA's TSF timer value at the start of reception of the first octet of the timestamp field of the reported Beacon, Measurement Pilot, or Probe Response frame at the time the Beacon, Measurement Pilot, or Probe Response frame being reported was received.

## 6.5.28 Type: BeaconReportingConfig

This type represents the Beacon Reporting Condition configuration as described in Figure 9-157 within IEEE 802.11-2016 [8].

**Table 6.5.28-1: Attributes of the BeaconReportingConfig**

Attribute name	Data type	Cardinality	Description
reportingCondition	Uint8	1	Reporting condition for the Beacon Report as per Table 9-89 of IEEE 802.11-2016 [8]: 0 = Report to be issued after each measurement. 1 = measured RCPI level is greater than the threshold. 2 = measured RCPI level is less than the threshold. 3 = measured RSNI level is greater than the threshold. 4 = measured RSNI level is less than the threshold. 5 = measured RCPI level is greater than a threshold defined by an offset from the serving AP's reference RCPI. 6 = measured RCPI level is less than a threshold defined by an offset from the serving AP's reference RCPI. 7 = measured RSNI level is greater than a threshold defined by an offset from the serving AP's reference RSNI. 8 = measured RSNI level is less than a threshold defined by an offset from the serving AP's reference RSNI. 9 = measured RCPI level is in a range bound by the serving AP's reference RCPI and an offset from the serving AP's reference RCPI. 10 = measured RSNI level is in a range bound by the serving AP's reference RSNI and an offset from the serving AP's reference RSNI.
threshold	Uint8	1	The threshold subfield contains either the threshold value or the offset value to be used for conditional reporting.  For reportingCondition subfield with values 1 and 2, the threshold value is a logarithmic function of the received signal power, as defined in section 9.4.2.38 of IEEE 802.11-2016 [8].  For reportingCondition subfield values 3 and 4, the threshold value is a logarithmic function of the signal-to-noise ratio, as described in section 9.4.2.41 of IEEE 802.11-2016 [8].  For reportingCondition subfield values 5 to 10, the offset value is an 8-bit 2s complement integer in units of 0,5 dBm. The indicated reporting condition applies individually to each measured Beacon, Measurement Pilot, or Probe Response frame.

## 6.5.29 Type: ReportedBeaconFrameInfo

This type represents the Beacon Reported Frame information as described in Figure 9-200 within IEEE 802.11-2016 [8].

**Table 6.5.29-1: Attributes of the ReportedBeaconFrameInfo**

Attribute name	Data type	Cardinality	Description
phyType	Uint8	1	Value between 0 and 127 coded according to dot11PHYType.
frameType	Uint8	1	A value of 0 indicates a Beacon or Probe Response. A value of 1 indicates a Measurement Pilot frame.

## 6.5.30 Type: BssidInfo

This type represents BSSID Information field within the Neighbor Report as described in Figure 9-296 of IEEE 802.11-2016 [8].

Table 6.5.30-1: Attributes of the BssidInfo

Attribute name	Data type	Cardinality	Description
apReachability	Uint8	1	The apReachability field indicates whether the AP identified by this BSSID is reachable by the STA that requested the neighbor report. Valid values: 0 = reserved 1 = not reachable 2 = unknown 3 = reachable.
security	Boolean	1	True indicates the AP identified by this BSSID supports the same security provisioning as used by the STA in its current association.  False indicates either that the AP does not support the same security provisioning or that the security information is not available at this time.
capabilities	BssCapabilities	1	Capability information for the AP indicated by this BSSID.
mobilityDomain	Boolean	1	True indicates the AP represented by this BSSID is including an MDE in its Beacon frames and that the contents of that MDE are identical to the MDE advertised by the AP sending the report.
highThroughput	Boolean	1	True indicates that the AP represented by this BSSID is an HT AP including the HT Capabilities element in its Beacons, and that the contents of that HT Capabilities element are identical to the HT Capabilities element advertised by the AP sending the report.
veryHighThroughput	Boolean	1	True indicates that the AP represented by this BSSID is a VHT AP and that the VHT Capabilities element, if included as a subelement in the report, is identical in content to the VHT Capabilities element included in the AP's Beacon.
ftm	Boolean	1	True indicates the AP represented by this BSSID is an AP that has set the Fine Timing Measurement Responder field of the Extended Capabilities element to 1.  False indicates either that the reporting AP has dot11FineTimingMsmtRespActivated equal to false, or the reported AP has not set the Fine Timing Measurement Responder field of the Extended Capabilities element to 1 or that the Fine Timing Measurement Responder field of the reported AP is not available to the reporting AP at this time.

### 6.5.31 Type: BssCapabilities

This type represents BSS Capabilities subfield within the Neighbor Report as described in Figure 9-297 of IEEE 802.11-2016 [8]. Boolean fields are set to true if the capability is advertised for the BSS.

Table 6.5.31-1: Attributes of the BssCapabilities

Attribute name	Data type	Cardinality	Description
spectrumManagement	Boolean	1	Spectrum Management required
qos	Boolean	1	QoS Option implemented
apsd	Boolean	1	APSD Option implemented
radioMeasurements	Boolean	1	Radio Measurement Activated
delayedBACK	Boolean	1	Delayed Block Ack Option implemented
immediateBACK	Boolean	1	Immediate Block Ack Option implemented

### 6.5.32 Type: NeighborReportConfig

This configuration applies to the Neighbor Report as described in clause 6.5.10 of the present document. This configuration follows the format as described in section 9.6.7.6, Neighbor Report Request, within IEEE 802.11-2016 [8].

**Table 6.5.32-1: Attributes of the NeighborReportConfig**

Attribute name	Data type	Cardinality	Description
ssid	String	0..1	The SSID field is optionally present. If present, it contains an SSID element. The presence of an SSID element in a Neighbor Report indicates a request for a neighbor list for the specified SSID in the SSID Element. The absence of an SSID element indicates neighbor report for the current ESS.
bssid	String	0..1	BSSID of the neighbor AP which information is intended to obtain. If no specific BSSID is given, the information will be provided for all APs matching the ssid criteria.
NOTE: Both SSID and BSSID are optional configuration parameters. BSSID is valid when a SSID setting is provided, otherwise ignored. If SSID is not included, the neighbor report will be generated for the SSID (i.e. current ESS) that the station is associated.			

### 6.5.33 Type: STACounterTriggerCondition

This type represents the STA Counter Trigger Condition configuration as described as described in Figure 9-160 and Figure 9-161 within IEEE 802.11-2016 [8]. If a threshold setting is provided, the associated report is to be included in the StaStatistics report, triggering when the requested threshold is exceeded.

**Table 6.5.33-1: Attributes of the STACounterTriggerCondition**

Attribute name	Data type	Cardinality	Description
failedCountThreshold	Uint32	0..1	Configure and set threshold for dot11FailedCount trigger
fcsErrorCountThreshold	Uint32	0..1	Configure and set threshold for dot11FCSErrorCount trigger
multipleRetryCountThreshold	Uint32	0..1	Configure and set threshold for dot11MultipleRetryCount trigger
frameDuplicateCountThreshold	Uint32	0..1	Configure and set threshold for dot11FrameDuplicateCount trigger
rtsFailureCountThreshold	Uint32	0..1	Configure and set threshold for dot11RTSFailureCount trigger
ackFailureCountThreshold	Uint32	0..1	Configure and set threshold for dot11AckFailureCount trigger
retryCountThreshold	Uint32	0..1	Configure and set threshold for dot11RetryCount trigger

### 6.5.34 Type: StaStatisticsGroupZeroData

This type represents STA Statistics Group Data for Group Identity = 0 as defined in Table 9-114 (Group Identity for a STA Statistics report) in IEEE 802.11-2016 [8].

**Table 6.5.34-1: Attributes of the StaStatisticsGroupZeroData**

Attribute name	Data type	Cardinality	Description
transmittedFragmentCount	Uint32	1	dot11TransmittedFragmentCount counter
groupTransmittedFrameCount	Uint32	1	dot11GroupTransmittedFrameCount counter
failedCount	Uint32	1	dot11FailedCount counter
receivedFragmentCount	Uint32	1	dot11ReceivedFragmentCount counter
groupReceivedFrameCount	Uint32	1	dot11GroupReceivedFrameCount counter
fcsErrorCount	Uint32	1	dot11FCSErrorCount counter
transmittedFrameCount	Uint32	1	dot11TransmittedFrameCount counter
reportingReasonStaCounters	ReportingReason StaCounters	0..1	Optionally reported reason for STA Statistics Group 0

### 6.5.35 Type: StaStatisticsGroupOneData

This type represents STA Statistics Group Data for Group Identity = 1 as defined in Table 9-114 (Group Identity for a STA Statistics report) in IEEE 802.11-2016 [8].

**Table 6.5.35-1: Attributes of the StaStatisticsGroupOneData**

Attribute name	Data type	Cardinality	Description
retryCount	UInt32	1	dot11RetryCount counter
multipleRetryCount	UInt32	1	dot11MultipleRetryCount counter
frameDuplicateCount	UInt32	1	dot11FrameDuplicateCount counter
rtsSuccessCount	UInt32	1	dot11RTSSuccessCount counter
rtsFailureCount	UInt32	1	dot11RTSFailureCount counter
ackFailureCount	UInt32	1	dot11AckFailureCount counter
reportingReasonStaCounters	ReportingReason StaCounters	0..1	Optionally reported reason for STA Statistics Group 1

### 6.5.36 Type: StaStatisticsGroup2to9Data

This type represents STA Statistics Group Data for Group Identity = 2 through 9 as defined in Table 9-114 (Group Identity for a STA Statistics report) in IEEE 802.11-2016 [8].

**Table 6.5.36-1: Attributes of the StaStatisticsGroup2to9Data**

Attribute name	Data type	Cardinality	Description
qosTransmittedFragmentCount	UInt32	1	dot11QosTransmittedFragmentCount counter
qosFailedCount	UInt32	1	dot11QosFailedCount counter
qosRetryCount	UInt32	1	dot11QosRetryCount counter
qosMultipleRetryCount	UInt32	1	dot11QosMultipleRetryCount counter
qosFrameDuplicateCount	UInt32	1	dot11QosFrameDuplicateCount counter
qosRTSSuccessCount	UInt32	1	dot11QosRTSSuccessCount counter
qosRTSFailureCount	UInt32	1	dot11QosRTSFailureCount counter
qosAckFailureCount	UInt32	1	dot11QosAckFailureCount counter
qosReceivedFragmentCount	UInt32	1	dot11QosReceivedFragmentCount counter
qosTransmittedFrameCount	UInt32	1	dot11QosTransmittedFrameCount counter
qosDiscardedFrameCount	UInt32	1	dot11QosDiscardedFrameCount counter
qosMPDUsReceivedCount	UInt32	1	dot11QosMPDUsReceivedCount counter
qosRetriesReceivedCount	UInt32	1	dot11QosRetriesReceivedCount counter
reportingReasonQoSCounters	ReportingReason QoSCounters	0..1	Optionally reported reason for STA Statistics Groups 2 to 9

### 6.5.37 Type: ReportingReasonStaCounters

This type represents optionally reported reason for STA Statistics Group Identities 0 or 1 (STA Counters) in the STA Statistics Optional subelements as described in Table 9-115 (Optional subelement IDs for STA Statistics report) within IEEE 802.11-2016 [8].

**Table 6.5.37-1: Attributes of the ReportingReasonStaCounters**

Attribute name	Data type	Cardinality	Description
failed	Boolean	1	dot11Failed
fcsError	Boolean	1	dot11FCSError
multipleRetry	Boolean	1	dot11MultipleRetry
frameDuplicate	Boolean	1	dot11FrameDuplicate
rtsFailure	Boolean	1	dot11RTSFailure
ackFailure	Boolean	1	dot11AckFailure
retry	Boolean	1	dot11Retry

## 6.5.38 Type: ReportingReasonQoSCounters

This type represents optionally reported reason for STA Statistics Group Identities 2 to 9 (QoS STA Counters) in the STA Statistics Optional subelements as described in Table 9-115 (Optional subelement IDs for STA Statistics report) within IEEE 802.11-2016 [8].

**Table 6.5.38-1: Attributes of the ReportingReasonQoSCounters**

Attribute name	Data type	Cardinality	Description
qosFailed	Boolean	1	dot11QoSFailed
qosRetry	Boolean	1	dot11QoSRetry
qosMultipleRetry	Boolean	1	dot11QoSMultipleRetry
qosFrameDuplicate	Boolean	1	dot11QoSFrameDuplicate
qosRtsFailure	Boolean	1	dot11QoSRTSFailure
qosAckFailure	Boolean	1	dot11QoSAckFailure
qosDiscarded	Boolean	1	dot11QoSDiscarded

## 6.5.39 Type: ChannelLoad

This type represents a Channel Load report from a station as defined in section 9.4.2.22.5 within IEEE 802.11-2016 [8].

**Table 6.5.39-1: Attributes of the ChannelLoad**

Attribute name	Data type	Cardinality	Description
stald	Stalidentity	0..1	Identifier to uniquely specify the station whose information is exposed within this report. If this report is contained within a data type that provides the station's identifier, this field may be omitted.
measurementId	String	1	Measurement ID of the Measurement configuration applied to this Channel Load Report.
operatingClass	Uint8	1	Operating Class field indicates an operating class value as defined in Annex E within IEEE 802.11-2016 [8].
channel	Uint8	1	Channel number indicates the channel number for which the measurement report applies.
measurementDuration	Uint8	1	Duration over which the Channel Load report was measured, in units of TUs of 1 024 $\mu$ s.
channelLoad	Uint8	1	Proportion of measurement duration for which the measuring STA determined the channel to be busy, as a percentage of time, linearly scaled with 255 representing 100 %.

## 6.5.40 Type: OBssLoad

This type represents the load of a Overlapping BSS as defined in section 9.2.123.1 within IEEE 802.11-2016 [8].

**Table 6.5.40-1: Attributes of the OBssLoad**

Attribute name	Data type	Cardinality	Description
allocatedTrafficSelfMean	Uint16	1	Mean of allocated traffic from this AP (BSS) in units of 32 $\mu$ s per second.
allocatedTrafficSelfStdDev	Uint16	0..1	Standard deviation from the mean of allocation traffic from this BSS in units of 32 $\mu$ s per second.
allocatedTrafficShareMean	Uint16	1	Mean of the sum of allocated traffic from other APs on the overlapping channel in unit of 32 $\mu$ s per second.
allocatedTrafficShareStdDev	Uint16	0..1	Standard deviation from the mean of the sum of allocated traffic from other APs on the overlapping channel in unit of 32 $\mu$ s per second.
overlap	Uint8	0..1	Indicates the number of other APs that are sharing the same channel as the reporting AP.

## 6.5.41 Type: MeasurementInfo

This type represents the information required to define client station measurements available from the WLAN Access Information Service.

The attributes of the MeasurementInfo shall follow the notations provided in Table 6.5.41-1.

**Table 6.5.41-1: Attributes for MeasurementInfo**

Attribute name	Data type	Cardinality	Description
measurementDuration	Uint16	0..1	Duration of the measurement in Time Units (TUs) of 1 024 $\mu$ s, as defined in section 11.11.4 of IEEE 802.11-2016 [8]. If not provided, the underlying system may utilize a default configuration that will be indicated in resulting measurement reports.
randomInterval	Uint16	0..1	Random interval to be used for starting the measurement in TUs of 1 024 $\mu$ s, as specified in section 11.11.3 of IEEE 802.11-2016 [8]. If not provided, the underlying system may utilize a default configuration that will be indicated in resulting measurement reports.
channelLoadConf	ChannelLoadConfig	0..1	Configuration related to the Channel Load.
beaconRequestConf	BeaconRequestConfig	0..1	Configuration related to Beacon Request.
staStatisticsConf	StaStatisticsConfig	0..1	Configuration related to the statistics provided by STAs.
neighborReportConf	NeighborReportConfig	0..1	Configuration related to Neighbor Reports.
NOTE 1: Only one of channelLoadConf, beaconRequestConf, staStatisticsConf or neighborReportConf is allowed in a MeasurementInfo instance.			
NOTE 2: As per IEEE 802.11-2016 [8], measurementDuration and randomInterval apply to channelLoad, beaconRequest, staStatistics, and neighborReport configurations.			

## 6.6 Referenced simple data types and enumerations

Referenced simple data types and enumerations are not used in the present document.

---

## 7 API definition

### 7.1 Introduction

This clause defines the resources and operations of the WLAN Access Information API (WAI API).

### 7.2 Global definitions and resource structure

All resource URLs of this API shall have the following root:

**{apiRoot}/{apiName}/{apiVersion}/**

The "apiRoot" is discovered using the service registry. The "apiName" shall be set to "wai" and "apiVersion" shall be set to "v1" for the present document. It includes the scheme ("http" or "https"), host and optional port, and an optional prefix string. The API shall support HTTP over TLS (also known as HTTPS defined in IETF RFC 2818 [2]). TLS version 1.2 as defined by IETF RFC 5246 [3] shall be supported. HTTP without TLS is not recommended. All resource URIs in the clauses below are defined relative to the above root URI.

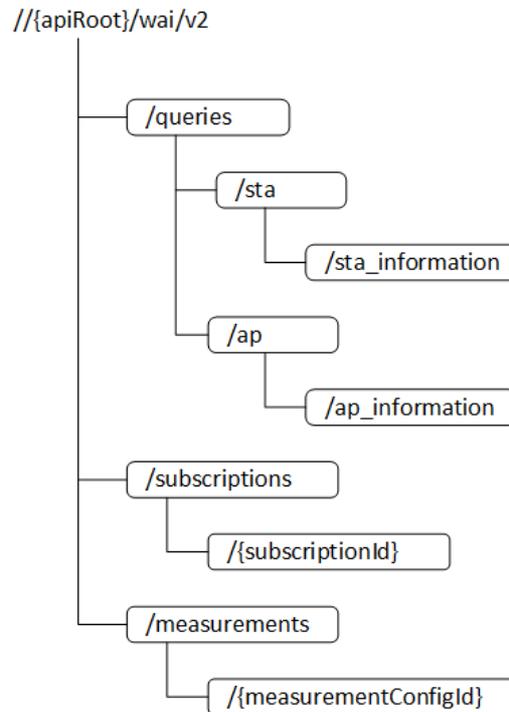
The content format of JSON shall be supported.

The JSON format is signalled by the content type "application/json".

This API shall require the use of the OAuth 2.0 client credentials grant type according to IETF RFC 6749 [4] with bearer tokens according to IETF RFC 6750 [5]. See clause 7.16 of ETSI GS MEC 009 [10] for more information. The token endpoint can be discovered as part of the service availability query procedure defined in ETSI GS MEC 011 [i.1]. How the client credentials are provisioned into the MEC application is out of scope of the present document.

This API supports additional application-related error information to be provided in the HTTP response when an error occurs. See clause 7.15 of ETSI GS MEC 009 [10] for more information.

Figure 7.2-1 illustrates the resource URI structure of this API.



**Figure 7.2-1: Resource URI structure of the WLAN Access Information API**

Table 7.2-1 provides an overview of the resources defined by the present document for the WAI API, and the applicable HTTP methods.

**Table 7.2-1: Resources and methods overview**

Resource name	Resource URI	HTTP method	Meaning
Access Point information	/queries/ap/ap_information	GET	Retrieve current status of Access Point information
Station information	/queries/sta/sta_information	GET	Retrieve current status of Station information
All subscriptions for a subscriber	/subscriptions	GET	Retrieve a list of active subscriptions for this subscriber
		POST	Create a new subscription
Existing subscription	/subscriptions/{subscriptionId}	GET	Retrieve information on current specific subscription
		PUT	Modify existing subscription by sending a new data structure
		DELETE	Cancel an existing subscription
Notification callback	Client provided callback reference	POST	Send a notification
All measurement configurations for a subscriber	/measurements	GET	Retrieve a list of configured measurements for this subscriber
		POST	Create a new measurement configuration
Existing measurement configuration	/measurements/{measurementConfigId}	GET	Retrieve information on an existing measurement configuration
		PUT	Modify an existing measurement configuration by sending a new data structure
		DELETE	Cancel an existing measurement configuration

## 7.3 Resource: ap\_information

### 7.3.1 Description

This resource is queried to retrieve information on WLAN access points.

### 7.3.2 Resource definition

Resource URI: {apiRoot}/wai/v1/queries/ap/ap\_information

This resource shall support the resource URI variables defined in Table 7.3.2-1.

**Table 7.3.2-1: Resource URI Variables for resource "ap\_information"**

Name	Definition
apiRoot	See clause 7.2

### 7.3.3 Resource Methods

#### 7.3.3.1 GET

The GET method is used to query information about the WLAN Access Points.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.3.3.1-1 and 7.3.3.1-2.

Table 7.3.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
filter		0..1	Attribute-based filtering expression according to clause 6.19 of ETSI GS MEC 009 [10]. The WLAN Access Information API shall support receiving this parameter as part of the URI query string. All attribute names that appear in the ap_information and in data types referenced from it shall be supported by the WLAN Access Information API in the filter expression.
all_fields		0..1	Include all complex attributes in the response. See clause 6.18 of ETSI GS MEC 009 [10] for details. The WLAN Access Information API shall support this parameter.
fields		0..1	Complex attributes to be included into the response. See clause 6.18 of ETSI GS MEC 009 [10] for details. The WLAN Access Information API shall support this parameter.
exclude_fields		0..1	Complex attributes to be excluded from the response. See clause 6.18 of ETSI GS MEC 009 [10] for details. The WLAN Access Information API shall support this parameter.
exclude_default		0..1	Indicates to exclude the following complex attributes from the response. See clause 6.18 of ETSI GS MEC 009 [10] for details. The WLAN Access Information API shall support this parameter. The following attributes shall be excluded from the ap_information structure in the response body if this parameter is provided, or none of the parameters "all_fields", "fields", "exclude_fields", "exclude_default" are provided: <ul style="list-style-type: none"> <li>Not applicable.</li> </ul>

Table 7.3.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	Data type	Cardinality	Response Codes	Remarks
	ApInfo	0..N	200 OK	Shall be returned when information about zero or more Access Points has been queried successfully.  The response body shall contain in an array the representations of zero or more Access Points, as defined in clause 6.2.2.  If the "filter" URI parameter or one of the "all_fields", "fields" (if supported), "exclude_fields" (if supported) or "exclude_default" URI parameters was supplied in the request, the data in the response body shall have been transformed according to the rules specified in clauses 6.19 and 6.18 of ETSI GS MEC 009 [10], respectively.
	ProblemDetails	1	400 Bad Request	Shall be returned upon the following error: Invalid attribute-based filtering expression.  The response body shall contain a ProblemDetails structure, in which the "detail" attribute should convey more information about the error.
	ProblemDetails	See annex E of [10]	4xx/5xx	In addition to the response codes defined above, any common error response code as defined in annex E of ETSI GS MEC 009 [10] may be returned.

## 7.3.3.2 PUT

Not applicable.

### 7.3.3.3 PATCH

Not applicable.

### 7.3.3.4 POST

Not applicable.

### 7.3.3.5 DELETE

Not applicable.

## 7.4 Resource: sta\_information

### 7.4.1 Description

This resource is queried to retrieve information on WLAN stations.

### 7.4.2 Resource definition

Resource URI: **{apiRoot}/wai/v1/queries/sta/sta\_information**

This resource shall support the resource URI variables defined in Table 7.4.2-1.

**Table 7.4.2-1: Resource URI Variables for resource "sta\_information"**

Name	Definition
apiRoot	See clause 7.2

### 7.4.3 Resource Methods

#### 7.4.3.1 GET

The GET method is used to query information about the WLAN stations.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.4.3.1-1 and 7.4.3.1-2.

Table 7.4.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
filter		0..1	Attribute-based filtering expression according to clause 6.19 of ETSI GS MEC 009 [10]. The WLAN Access Information API shall support receiving this parameter as part of the URI query string.  All attribute names that appear in the ap_information and in data types referenced from it shall be supported by the WLAN Access Information API in the filter expression.
all_fields		0..1	Include all complex attributes in the response. See clause 6.18 of ETSI GS MEC 009 [10] for details. The WLAN Access Information API shall support this parameter.
fields		0..1	Complex attributes to be included into the response. See clause 6.18 of ETSI GS MEC 009 [10] for details. The WLAN Access Information API shall support this parameter.
exclude_fields		0..1	Complex attributes to be excluded from the response. See clause 6.18 of ETSI GS MEC 009 [10] for details. The WLAN Access Information API shall support this parameter.
exclude_default		0..1	Indicates to exclude the following complex attributes from the response. See clause 6.18 of ETSI GS MEC 009 [10] for details. The WLAN Access Information API shall support this parameter. The following attributes shall be excluded from the ap_information structure in the response body if this parameter is provided, or none of the parameters "all_fields", "fields", "exclude_fields", "exclude_default" are provided: <ul style="list-style-type: none"> <li>Not applicable.</li> </ul>

Table 7.4.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	Data type	Cardinality	Response Codes	Remarks
	StaInfo	0..N	200 OK	Shall be returned when information about zero or more WLAN stations has been queried successfully.  The response body shall contain in an array the representations of zero or more WLAN stations, as defined in clause 6.2.2.  If the "filter" URI parameter or one of the "all_fields", "fields" (if supported), "exclude_fields" (if supported) or "exclude_default" URI parameters was supplied in the request, the data in the response body shall have been transformed according to the rules specified in clauses 6.19 and 6.18 of ETSI GS MEC 009 [10], respectively.
	ProblemDetails	1	400 Bad Request	Shall be returned upon the following error: Invalid attribute-based filtering expression.  The response body shall contain a ProblemDetails structure, in which the "detail" attribute should convey more information about the error.
	ProblemDetails	See annex E of [10]	4xx/5xx	In addition to the response codes defined above, any common error response code as defined in annex E of ETSI GS MEC 009 [10] may be returned.

## 7.3.3.2 PUT

Not applicable.

### 7.3.3.3 PATCH

Not applicable.

### 7.3.3.4 POST

Not applicable.

### 7.3.3.5 DELETE

Not applicable.

## 7.5 Resource: subscriptions

### 7.5.1 Description

This resource contains various resources related to subscriptions for notifications.

### 7.5.2 Resource definition

Resource URI: {apiRoot}/wai/v1/subscriptions/

This resource shall support the resource URI variables defined in Table 7.5.2-1.

**Table 7.5.2-1: Resource URI variables for resource "subscriptions"**

Name	Definition
apiRoot	See clause 7.2

### 7.5.3 Resource methods

#### 7.5.3.1 GET

The GET method is used to request information about the subscriptions for this requestor. Upon success, the response contains entity body with the list of links to the subscriptions that are present for the requestor.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.5.3.1-1 and 7.5.3.1-2.

**Table 7.5.3.1-1: URI query parameters supported by the GET method on this resource**

Name	Data type	Cardinality	Remarks
subscription_type	String	0..1	Query parameter to filter on a specific subscription type. Permitted values: <ul style="list-style-type: none"> <li>• assoc_sta Associated Stations</li> <li>• sta_data_rate Station Data Rate</li> <li>• measure_report Measurement Report</li> </ul>

Table 7.5.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	Data type	Cardinality	Response Codes	Remarks
	SubscriptionLink List	1	200 OK	Upon success, a response body containing the list of links to requestor's subscriptions is returned.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide the any of the content formats supported by the client.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

### 7.5.3.2 PUT

Not applicable.

### 7.5.3.3 PATCH

Not applicable.

### 7.5.3.4 POST

The POST method is used to create a new subscription to WLAN notifications. Upon success, the response contains entity body describing the created subscription.

This method shall support the request and response data structures, and response codes, as specified in Table 7.5.3.4-1.

Table 7.5.3.4-1: Data structures supported by the POST request/response on this resource

Request body	Data type	Cardinality	Remarks	
	{NotificationSubscription}	1	The entity body in the request contains data type of the specific WLAN event subscription that is to be created, where the data type options are listed below and defined in clauses 6.3.2, 6.3.3, and 6.3.5: <ul style="list-style-type: none"> <li>• AssocStaSubscription.</li> <li>• StaDataRateSubscription.</li> <li>• MeasurementReportSubscription.</li> </ul>	
Response body	Data type	Cardinality	Response Codes	Remarks
	{NotificationSubscription}	1	201 Created	Indicates successful resource creation, where the resource URI shall be returned in the HTTP Location header field.  In the returned NotificationSubscription structure, the created subscription is described using the appropriate data type from the list below and as defined in clauses 6.3.2, 6.3.3, and 6.3.5: <ul style="list-style-type: none"> <li>• AssocStaSubscription.</li> <li>• StaDataRateSubscription.</li> <li>• MeasurementReportSubscription.</li> </ul>
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide the any of the content formats supported by the client.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	415 Unsupported Media Type	It is used to indicate that the server or the client does not support the content type of the entity body.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

	Data type	Cardinality	Response Codes	Remarks
<b>Response body</b>	ProblemDetails	0..1	422 Unprocessable Entity	It is used to indicate that the server understands the content type of the request entity and that the syntax of the request entity is correct but that the server is unable to process the contained instructions. This error condition can occur if an JSON request body is syntactically correct but semantically incorrect, for example if the target area for the request is considered too large. This error condition can also occur if the capabilities required by the request are not supported.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

### 7.5.3.5 DELETE

Not applicable.

## 7.6 Resource: existing subscription

### 7.6.1 Description

This resource represents a subscription that the client has created to receive WLAN event notifications.

### 7.6.2 Resource definition

Resource URI: {apiRoot}/wai/v1/subscriptions/{subscriptionId}

This resource shall support the resource URI variables defined in Table 7.6.2-1.

**Table 7.6.2-1: Resource URI variables for resource "existing subscription"**

Name	Definition
apiRoot	See clause 7.2.
subscriptionId	Refers to created subscription, where the WLAN Access Information API allocates a unique resource name for this subscription. The resource name can be also used to identify the resource.

### 7.6.3 Resource methods

#### 7.6.3.1 GET

The GET method is used to retrieve information about this subscription. Upon success, the response contains entity body with the data type describing the subscription.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.6.3.1-1 and 7.6.3.1-2.

Table 7.6.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.6.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	Data type	Cardinality	Response Codes	Remarks
	{NotificationSubscription}	1	200 OK	Upon success, a response body containing data type describing the specific WLAN event subscription is returned. The allowed data types for subscriptions are defined in clauses 6.3.2, 6.3.3, and 6.3.5: <ul style="list-style-type: none"> <li>• AssocStaSubscription.</li> <li>• StaDataRateSubscription.</li> <li>• MeasurementReportSubscription.</li> </ul>
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide the any of the content formats supported by the client.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

### 7.6.3.2 PUT

The PUT method is used to update the existing subscription. PUT method in this case has "replace" semantics. Upon successful operation, the target resource is updated with new Data Type received within the message body of the PUT request.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.6.3.2-1 and 7.6.3.2-2.

Table 7.6.3.2-1: URI query parameters supported by the PUT method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.6.3.2-2: Data structures supported by the PUT request/response on this resource

Request body	Data type	Cardinality	Remarks	
	{NotificationSubscription}	1	New NotificationSubscription is included as entity body of the request. The allowed data types for subscriptions are defined in clauses 6.3.2, 6.3.3, and 6.3.5: <ul style="list-style-type: none"> <li>• AssocStaSubscription.</li> <li>• StaDataRateSubscription.</li> <li>• MeasurementReportSubscription.</li> </ul>	
Response body	Data type	Cardinality	Response Codes	Remarks
	{NotificationSubscription}	1	200 OK	Upon success, a response body containing data type describing the updated subscription is returned. The allowed data types for subscriptions are defined in clauses 6.3.2, 6.3.3, and 6.3.5: <ul style="list-style-type: none"> <li>• AssocStaSubscription.</li> <li>• StaDataRateSubscription.</li> <li>• MeasurementReportSubscription.</li> </ul>
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide the any of the content formats supported by the client.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	412 Precondition Failed	It is used when a condition has failed during conditional requests, e.g. when using ETags to avoid write conflicts when using PUT.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

	Data type	Cardinality	Response Codes	Remarks
Response body	ProblemDetails	0..1	422 Unprocessable Entity	It is used to indicate that the server understands the content type of the request entity and that the syntax of the request entity is correct but that the server is unable to process the contained instructions. This error condition can occur if an JSON request body is syntactically correct but semantically incorrect, for example if the target area for the request is considered too large. This error condition can also occur if the capabilities required by the request are not supported.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

### 7.6.3.3 PATCH

Not applicable.

### 7.6.3.4 POST

Not applicable.

### 7.6.3.5 DELETE

The DELETE method is used to cancel the existing subscription. Cancellation can be made by deleting the resource that represents existing subscription.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.6.3.5-1 and 7.6.3.5-2.

**Table 7.6.3.5-1: URI query parameters supported by the DELETE method on this resource**

Name	Data type	Cardinality	Remarks
n/a			

Table 7.6.3.5-2: Data structures supported by the DELETE request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	Data type	Cardinality	Response Codes	Remarks
	n/a		204 No Content	Upon success, a response 204 No Content without any response body is returned.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

## 7.7 Resource: measurements

### 7.7.1 Description

This resource contains various resources related to WAI measurement configurations. A WAI service consumer uses measurement configurations to tailor measurement information returned from the ap\_information and sta\_information queries.

### 7.7.2 Resource definition

Resource URI: {apiRoot}/wai/v2/measurements/

This resource shall support the resource URI variables defined in Table 7.7.2-1.

Table 7.7.2-1: Resource URI variables for resource "measurements"

Name	Definition
apiRoot	See clause 7.2

### 7.7.3 Resource methods

#### 7.7.3.1 GET

The GET method is used to request information about the measurement configurations for this requestor. Upon success, the response contains an entity body with the list of links to the measurement configurations that are present for the requestor.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.7.3.1-1 and 7.7.3.1-2.

Table 7.7.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
Na			

Table 7.7.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	Data type	Cardinality	Response Codes	Remarks
	MeasurementConfig LinkList	1	200 OK	Upon success, a response body containing the list of links to requestor's measurement configurations is returned.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide the any of the content formats supported by the client.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

### 7.5.3.2 PUT

Not applicable.

### 7.5.3.3 PATCH

Not applicable.

### 7.7.3.4 POST

The POST method is used to create a new WLAN measurement configuration. Upon success, the response contains an entity body describing the created measurement configuration.

This method shall support the request and response data structures, and response codes, as specified in Table 7.7.3.4-1.

**Table 7.7.3.4-1: Data structures supported by the POST request/response on this resource**

Request body	Data type	Cardinality	Remarks	
	MeasurementConfig	1	Entity body in the request contains the measurement configuration as defined in clause 6.2.4.	
Response body	Data type	Cardinality	Response Codes	Remarks
	MeasurementConfig	1	201 Created	Indicates successful resource creation, where the resource URI shall be returned in the HTTP Location header field.  The returned MeasurementConfig structure describes the created configuration resource.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide the any of the content formats supported by the client.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	415 Unsupported Media Type	It is used to indicate that the server or the client does not support the content type of the entity body.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	422 Unprocessable Entity	It is used to indicate that the server understands the content type of the request entity and that the syntax of the request entity is correct but that the server is unable to process the contained instructions. This error condition can occur if an JSON request body is syntactically correct but semantically incorrect, for example if the target area for the request is considered too large. This error condition can also occur if the capabilities required by the request are not supported.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

Response body	Data type	Cardinality	Response Codes	Remarks
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

### 7.7.3.5 DELETE

Not applicable.

## 7.8 Resource: existing measurements

### 7.8.1 Description

This resource represents a measurement configuration that a WAI service consumer has created to configure measurement information available via the `ap_information` and `sta_information` queries.

### 7.8.2 Resource definition

Resource URI: `{apiRoot}/wai/v2/measurements/{measurementConfigId}`

This resource shall support the resource URI variables defined in Table 7.8.2-1.

**Table 7.8.2-1: Resource URI variables for resource "existing measurement configuration"**

Name	Definition
<code>apiRoot</code>	See clause 7.2.
<code>measurementConfigId</code>	Refers to a created measurement configuration, where the WLAN Access Information API allocates a unique resource name for this configuration. The resource name can be also used to identify the resource.

### 7.8.3 Resource methods

#### 7.8.3.1 GET

The GET method is used to retrieve information about this measurement configuration. Upon success, the response contains entity body with the data type describing the measurement configuration.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.8.3.1-1 and 7.8.3.1-2.

**Table 7.8.3.1-1: URI query parameters supported by the GET method on this resource**

Name	Data type	Cardinality	Remarks
n/a			

Table 7.8.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
n/a				
Response body	Data type	Cardinality	Response Codes	Remarks
	MeasurementConfig	1	200 OK	Upon success, the response body contains the measurement configuration for the queried measurementConfigId.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide the any of the content formats supported by the client.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

### 7.8.3.2 PUT

The PUT method is used to update the existing WAI measurement configuration. PUT method in this case has "replace" semantics. Upon successful operation, the target resource is updated with new Data Type received within the message body of the PUT request.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.8.3.2-1 and 7.8.3.2-2.

Table 7.8.3.2-1: URI query parameters supported by the PUT method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.8.3.2-2: Data structures supported by the PUT request/response on this resource

Request body	Data type	Cardinality	Remarks	
	MeasurementConfig	1	Entity body in the request contains the measurement configuration as defined in clause 6.2.4.	
Response body	Data type	Cardinality	Response Codes	Remarks
	MeasurementConfig	1	200 OK	Indicates successful resource update. The returned MeasurementConfig structure describes the updated measurement configuration resource.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide the any of the content formats supported by the client.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	412 Precondition Failed	It is used when a condition has failed during conditional requests, e.g. when using ETags to avoid write conflicts when using PUT.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	422 Unprocessable Entity	It is used to indicate that the server understands the content type of the request entity and that the syntax of the request entity is correct but that the server is unable to process the contained instructions. This error condition can occur if an JSON request body is syntactically correct but semantically incorrect, for example if the target area for the request is considered too large. This error condition can also occur if the capabilities required by the request are not supported.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

	Data type	Cardinality	Response Codes	Remarks
Response body	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

### 7.8.3.3 PATCH

Not applicable.

### 7.8.3.4 POST

Not applicable.

### 7.8.3.5 DELETE

The DELETE method is used to remove an existing measurement configuration. After the measurement configuration is removed, the configuration's associated measurementId will no longer be available in the ap\_information and sta\_information queries.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in Tables 7.8.3.5-1 and 7.8.3.5-2.

**Table 7.8.3.5-1: URI query parameters supported by the DELETE method on this resource**

Name	Data type	Cardinality	Remarks
n/a			

**Table 7.8.3.5-2: Data structures supported by the DELETE request/response on this resource**

Request body	Data type	Cardinality	Remarks	
	Data type	Cardinality	Response Codes	Remarks
Response body	n/a			
	n/a		204 No Content	Upon success, a response 204 No Content without any response body is returned.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource.  More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.  In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

---

## Annex A (informative): Complementary material for API utilization

To complement the definitions for each method and resource defined in the interface clauses of the present document, ETSI MEC ISG is providing for the WLAN Access Information (WAI) API a supplementary description file compliant to the OpenAPI Specification [i.2].

In case of discrepancies between the supplementary description file and the related data structure definitions in the present document, the data structure definitions take precedence.

The supplementary files, relating to the present document, are located at <https://forge.etsi.org/rep/mec/g028-wai-api>.

---

## History

<b>Document history</b>		
V2.1.1	June 2020	Publication
V2.2.1	July 2021	Publication