

ETSI GS MEC 013 V2.2.1 (2022-01)



Multi-access Edge Computing (MEC); Location 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-0013v221LocationApi

Keywords

API, location, MEC, service

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.

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 2022.

All rights reserved.

Contents

Intellectual Property Rights	5
Foreword.....	5
Modal verbs terminology.....	5
1 Scope	6
2 References	6
2.1 Normative references	6
2.2 Informative references.....	7
3 Definition of terms, symbols and abbreviations.....	7
3.1 Terms.....	7
3.2 Symbols.....	7
3.3 Abbreviations	8
4 Overview	8
5 Description of the service (informative).....	8
5.1 Introduction	8
5.2 Relation with OMA APIs	9
5.2.1 Relation with OMA API for Zonal Presence	9
5.2.2 Relation with OMA API for Terminal Location.....	9
5.3 Sequence diagrams	9
5.3.1 Introduction.....	9
5.3.2 UE Location Lookup	10
5.3.3 UE Information Lookup.....	10
5.3.4 UE Location Subscribe	11
5.3.5 UE Information Subscribe	11
5.3.6 Subscribe Cancellation	12
5.3.7 Radio Node Location Lookup.....	12
5.3.8 UE Tracking Subscribe	13
5.3.9 UE Distance Lookup.....	13
5.3.10 UE Distance Subscribe	14
5.3.11 UE Area Subscribe.....	15
6 Data Model	15
6.1 Introduction	15
6.2 Resource data types	15
6.2.1 Zonal presence data types	15
6.2.1A Terminal location data types	16
6.2.2 Type: UserInfo.....	16
6.3 Subscription data types.....	17
6.3.1 Zonal presence data types	17
6.3.2 Terminal Location data types.....	17
6.3.3 Type: NotificationSubscriptionList	17
6.4 Notifications data types	18
6.4.1 Zonal presence data types	18
6.4.2 Terminal location data types	18
6.5 Referenced structured data types.....	18
6.5.1 Introduction.....	18
6.5.2 Type: TimeStamp	18
6.5.3 Type: LocationInfo	19
6.6 Mapping of Identifiers.....	20
6.6.1 accessPointId	20
6.6.2 userId	20
7 API definition.....	20
7.1 Introduction	20
7.2 Global definitions and relation with OMA resources.....	20

7.3	Mapping of services to API functions	22
7.3.1	Introduction.....	22
7.3.2	UE Location Lookup	22
7.3.2.1	Definition	22
7.3.2.2	Example: Two steps method to retrieve the geographical coordinates of an access point a specific user is connected to (informative).....	23
7.3.3	UE Information Lookup.....	24
7.3.3.1	Definition	24
7.3.3.2	Example: Retrieve UE Information for all the users (informative).....	24
7.3.3.3	Example: Retrieve UE Information for the users in a zone (informative).....	25
7.3.3.4	Example: Retrieve UE Information for the users in an access point of a zone (informative)	26
7.3.4	UE Location Subscribe	26
7.3.4.1	Definition	26
7.3.4.2	Example: Create a UE Location subscription (informative)	27
7.3.4.3	Example: Client notification about UE Location change (informative).....	27
7.3.5	UE Information Subscribe	28
7.3.5.1	Definition	28
7.3.5.2	Example: Create a UE Information subscription (informative)	28
7.3.5.3	Example: Client notification about UE Information change (informative).....	29
7.3.6	Subscribe Cancellation	29
7.3.6.1	Definition	29
7.3.6.2	Example: Cancelling a UE Location subscription (informative)	30
7.3.6.3	Example: Cancelling a UE Information subscription (informative).....	30
7.3.6.4	Example: Cancelling a UE Tracking subscription (informative)	30
7.3.6.5	Example: Cancelling a UE Distance subscription (informative).....	31
7.3.6.6	Example: Cancelling a UE Area subscription (informative).....	31
7.3.7	Radio Node Location Lookup.....	31
7.3.7.1	Definition	31
7.3.7.2	Example: Retrieve a list of radio nodes associated to a zone (informative).....	31
7.3.8	UE Tracking Subscribe	32
7.3.8.1	Definition	32
7.3.8.2	Example: Create a UE Tracking subscription (informative).....	33
7.3.8.3	Example: Client notification about UE Tracking change (informative)	33
7.3.9	UE Distance Lookup.....	34
7.3.9.1	Definition	34
7.3.9.2	Example: Distance between a terminal and a location (informative).....	34
7.3.9.3	Example: Distance between two terminals (informative)	35
7.3.10	UE Distance Subscribe	35
7.3.10.1	Definition	35
7.3.10.2	Example: Create a UE Distance subscription (informative)	35
7.3.10.3	Example: Client notification about UE Tracking change (informative)	36
7.3.11	UE Area Subscribe.....	37
7.3.11.1	Definition	37
7.3.11.2	Example: Create a UE Area subscription (informative).....	37
7.3.11.3	Example: Client notification about UE Area change (informative)	38
Annex A (informative):	Complementary material for API utilisation	39
History		40

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 MEC Location Service. It describes the related application policy information including authorization and access control, information flows, required information and service aggregation patterns. The present document specifies the necessary API with the data model and data format.

It is to be noted that the actual data model and data format which is functional for the present API re-uses the definitions in "RESTful Network API for Zonal Presence" [5] and "RESTful Network API for Terminal Location" [6] published by the Open Mobile Alliance.

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] ETSI GS MEC 001: "Multi-access Edge Computing (MEC); Terminology".
- [2] ETSI GS MEC 002: "Multi-access Edge Computing (MEC); Phase 2: Use Cases and Requirements".
- [3] ETSI GS MEC 003: "Multi-access Edge Computing (MEC); Framework and Reference Architecture".
- [4] ETSI GS MEC 009 (V2.1.1): "Multi-access Edge Computing (MEC); General principles for MEC Service APIs".
- [5] OMA-TS-REST-NetAPI-ZonalPresence-V1-0-20160308-C: "RESTful Network API for Zonal Presence".
- [6] OMA-TS-REST-NetAPI-TerminalLocation-V1-0-1-20151029-A: "RESTful Network API for Terminal Location".
- [7] OMA-TS-REST-NetAPI-ACR-V1-0-20151201-C: "RESTful Network API for Anonymous Customer Reference Management".
- [8] ETSI TS 129 171: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; Location Services (LCS); LCS Application Protocol (LCS-AP) between the Mobile Management Entity (MME) and Evolved Serving Mobile Location Centre (E-SMLC); SLs interface (3GPP TS 29.171)".
- [9] Void.
- [10] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

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

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

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

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

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

[13] National Geospatial-Intelligence Agency (NGA) Standardization Document: "World Geodetic System 1984, Its Definition and Relationships with Local Geodetic Systems", NGA.STND.0036-1.0.0-WGS84, 2014-07-08, Version 1.0.0.

NOTE: Available at <https://earth-info.nga.mil/php/download.php?file=coord-wgs84>.

[14] ETSI TS 123 032: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); Universal Geographical Area Description (GAD) (3GPP TS 23.032)".

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

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

[16] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3".

NOTE: Available at <https://tools.ietf.org/html/rfc8446>.

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] SCF 084.07.01: "Small cell zone services - RESTful bindings".

[i.2] SCF 152.07.01: "Small cell services API".

[i.3] OpenAPI™ Specification.

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

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 [1] and the following apply:

Anonymous Customer Reference (ACR): Uniform Resource Identifier (URI) scheme describing an anonymous reference that can be mapped to a resource or user/user group

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GS MEC 001 [1] and the following apply:

ACR	Anonymous Customer Reference
API	Application Programming Interface
E-SMLC	Evolved Serving Mobile Location Centre
ID	Identifier
LS	Location Service
OMA	Open Mobile Alliance
REST	REpresentational State Transfer
SCF	Small Cell Forum
UE	User Equipment
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
WGS	World Geodetic System

4 Overview

The present document specifies the Location Service API to support the requirements defined for Multi-access Edge Computing in ETSI GS MEC 002 [2]. The Location Service (LS) defined in the present document leverages the Zonal Presence service described in SCF 084.07.01 [i.1] and in SCF 152.07.01 [i.2]. The Location Service is accessible through the API defined in the Open Mobile Alliance (OMA) specification "RESTful Network API for Zonal Presence" [5]. In addition, the LS offers capabilities of the API defined by the OMA specification "RESTful Network API for Terminal Location" [6].

In addition to referencing the OMA APIs [5] and [6], the present document contains application instructions on how to apply these OMA APIs in the MEC environment, and application instructions on how the Anonymous Customer Reference (ACR) defined in [7] can be applied to fulfil the ETSI MEC requirement [Location-04] on addressing user categories defined in ETSI GS MEC 002 [2], and application instructions on how the 3GPP Cell Identifiers defined in ETSI TS 129 171 [8] can be mapped to the Access Point identifier of the OMA API [5].

5 Description of the service (informative)

5.1 Introduction

Location Service is a service to provide the location related information to the MEC platform or authorized applications. With location related information, the MEC platform or applications perform the active device location tracking, location-based service recommendation, etc., see ETSI GS MEC 002 [2]. The Location Service is registered and discovered over the Mp1 reference point defined in ETSI GS MEC 003 [3].

The Location Service supports the location retrieval mechanism, i.e. the location is reported only once for each location information request.

The Location Service supports the location subscribe mechanism, i.e. the location is able to be reported multiple times for each location request, periodically or based on specific events, such as location change.

The Location Service supports the anonymous location report, i.e. without the related UE ID information, e.g. for the statistics collection.

The Location Service supports the following location information:

- the location information of specific UEs currently served by the radio node(s) associated with the MEC host;
- the location information of all UEs currently served by the radio node(s) associated with the MEC host;
- the distance between specified UEs currently served by the radio node(s) associated with the MEC host;

- the distance between a specified location and a UE currently served by the radio node(s) associated with the MEC host;
- the location information of a certain category of UEs currently served by the radio node(s) associated with the MEC host;
- a list of UEs in a particular location area;
- the specific UEs which move in or out of a particular location area;
- information about the location of all radio nodes currently associated with the MEC host;
- subscriptions to location information are also offered, including periodic location information updates, updates on changes in distance and location updates relating to UEs in a particular location area.

The Location Service supports both geolocation, such as a geographical coordinates, and logical location, such as a Cell ID.

5.2 Relation with OMA APIs

5.2.1 Relation with OMA API for Zonal Presence

The Zonal Presence service utilizes the concept of "zone", defined in OMA API [5]. According to the definition therein, a zone lends itself to be used to group all the radio nodes that are associated to a MEC host, or a subset of them, according to the desired deployment.

The OMA Zonal Presence API provides means for an application to retrieve information about a zone, the access points associated to the zones and the users that are connected to the access points. In addition, the OMA Zonal Presence API allows authorized application to subscribe to a notification mechanism, reporting about user activities within a zone.

In the present document, clauses 6 and 7 define how the OMA Zonal Presence API is used in providing the MEC Location Service.

5.2.2 Relation with OMA API for Terminal Location

The OMA Terminal Location API [6] provides means for an application to retrieve information about the location of a terminal or group of terminals, as well as the distance between terminals, or distance of a terminal from a given location. In addition, the OMA Terminal Location API allows authorized application to subscribe to a notification mechanism, reporting about terminal location activities.

In the present document, clauses 6 and 7 define how the OMA Terminal Location API is used in providing the MEC Location Service.

5.3 Sequence diagrams

5.3.1 Introduction

Clauses 5.3.2 to 5.3.11 describe how the Service Consumers interact with the Location Service over LS API to obtain location information of a UE, a group of UEs or the radio nodes currently associated with the MEC host. The sequence diagrams that are relevant for the Location Service are presented.

The Service Consumers communicate with the Location Service over the LS API. Both the MEC application and MEC platform can be Service Consumers. Location information can be provided by both MEC platform and MEC application.

The Location Service API supports both queries and subscriptions (pub/sub mechanism) that can be used over RESTful API as per the mechanism in OMA APIs [5] and [6].

5.3.2 UE Location Lookup

The UE Location Lookup is the procedure for applications acquiring the current location information of a specific UE or a group of UEs. In this procedure, the MEC Application will query the UE information firstly and the procedure may stop if the UE information in response message contains the location information. Otherwise, the MEC Application will query the location of the access point by the zone ID and access point ID which are contained in the UE information.

The UE Location Lookup procedure is illustrated in figure 5.3.2-1.

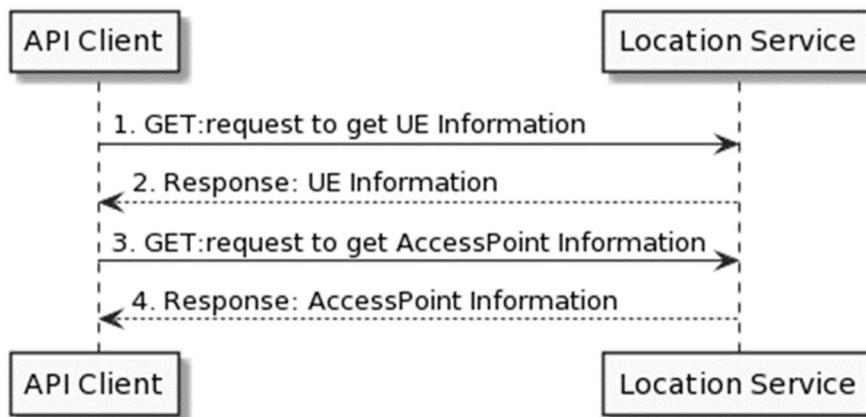


Figure 5.3.2-1: Flow of UE Location Lookup

Steps 1) and 2) are defined in clause 5.3.3 of the present document. If the UserInfo element in the response contains any of the locationInfo or contextLocationInfo sub-elements, then the procedure may stop, otherwise steps 3) and 4) may be issued.

- 3) The MEC application looks up access point information by sending a request to the resource AccessPointInfo. The URI of the resource consists of zone ID and access point ID which are obtained in the message received in step 2).
- 4) The Location Service returns a response with a message body including the access point information in which the location information is contained.

5.3.3 UE Information Lookup

The UE Information Lookup is the procedure for applications acquiring information pertaining to one or more UEs in a particular location. In this procedure, the Location Service will report the lookup result once on each request.

The UE Information Lookup procedure is illustrated in figure 5.3.3-1.

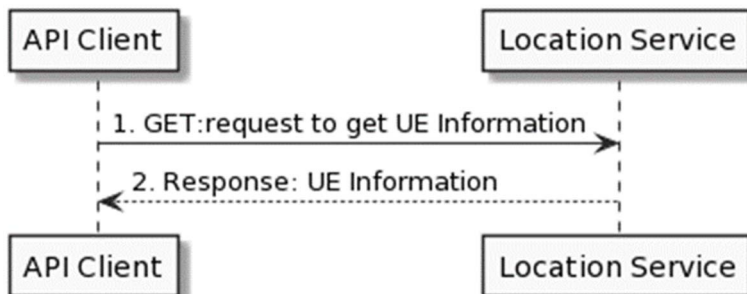


Figure 5.3.3-1: Flow of UE Information Lookup

- 1) The MEC application looks up UE information in a particular area by sending a request to the resource representing the UE Information, which includes location area information. The request may optionally include one or more query parameters specifying the sub-region of interest, the Access Point identifier and requirements on reporting timeliness and accuracy.
- 2) The Location Service returns a response with a message body including the UE information of the UE(s) according to the query parameters in the location area, if the UE information lookup is accepted.

5.3.4 UE Location Subscribe

The UE Location Subscribe is the procedure for applications to request to receive notifications about location information changes of a specific UE or a group of UEs, which for instance may help the applications to regularly track the UE(s) location. In this procedure, the Location Service will continue to report the subscribed information until the subscription is cancelled, or an optional specified time limit.

The UE Location Subscribe procedure is illustrated in figure 5.3.4-1.

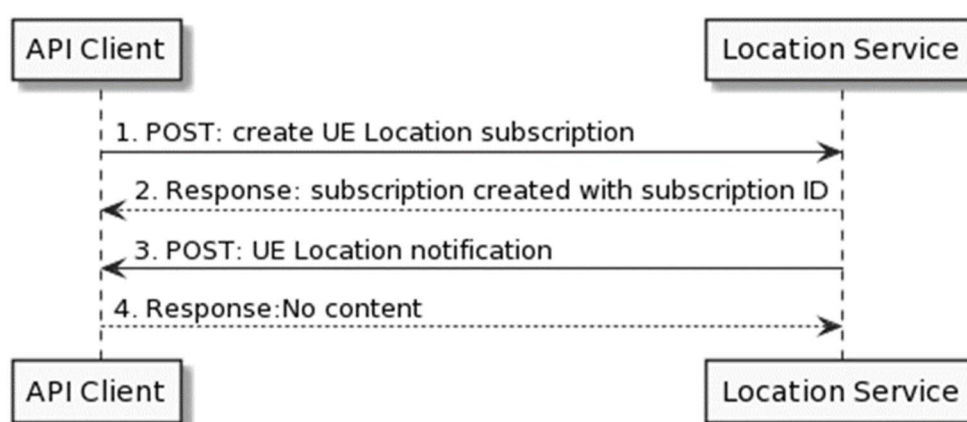


Figure 5.3.4-1: Flow of UE Location Subscribe

- 1) The MEC application subscribes to UE location notification by requesting the creation of a resource containing the subscription details, which includes UE(s) identifier, e.g. UE IP address, and a callbackURL for receiving the UE location.
- 2) The Location Service returns a response with resource URI containing the subscriptionId.
- 3) The Location Service reports the up-to-date subscribed information to the MEC application by sending a message with the message body containing the UE Location notification to the callbackURL, which includes location information.
- 4) The MEC application returns a response with the code 204.

5.3.5 UE Information Subscribe

The UE Information Subscribe is the procedure for applications to receive notifications of UE Information updates for the list of UEs in a particular location. Notifications can be to update status changes of UE information, e.g. UE entering the specified area. In this procedure, the Location Service will continue to report the subscribed information until the subscription is cancelled.

The UE Information Subscribe procedure is illustrated in figure 5.3.5-1.

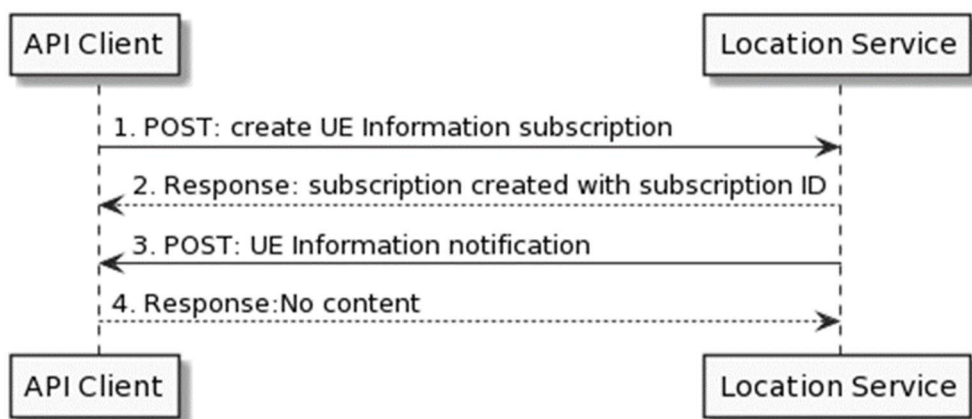


Figure 5.3.5-1: Flow of UE Information Subscribe

- 1) The MEC application subscribes to the UE information notification for a particular area by requesting the creation of a resource containing all subscription details, which includes location area information and a callbackURL for receiving the UE information.
- 2) The Location Service returns a response with resource URI containing the subscriptionId.
- 3) The Location Service reports the subscribed information to the MEC application by sending a message with the request body containing the UE Information notification to the callbackURL, which includes the UE information for each UE in the location area.
- 4) The MEC application returns a response with the code 204.

5.3.6 Subscribe Cancellation

The Subscribe Cancellation is the procedure for applications to cancel the subscription, with which the Location Service stop reporting the subscribed information to the application.

The Subscribe Cancellation procedure is illustrated in figure 5.3.6-1.

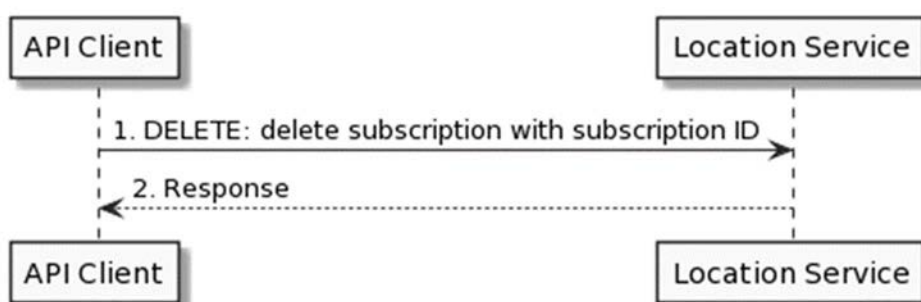


Figure 5.3.6-1: Flow of Location Subscribe Cancellation

- 1) The MEC application unsubscribes to notifications by sending a request to delete the resource URI containing the subscriptionId.
- 2) The Location Service returns a successful response if the subscription cancellation is accepted.

5.3.7 Radio Node Location Lookup

The Radio Node Location Lookup procedure enables an application to make a location enquiry about the radio nodes currently associated with the MEC host. The Radio Node Location Lookup procedure is illustrated in figure 5.3.7-1.

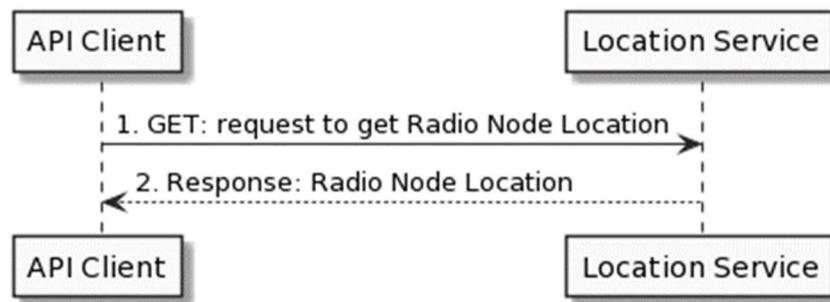


Figure 5.3.7-1: Radio Node Location Lookup

- 1) The MEC application makes an enquiry about the radio nodes currently associated with the MEC host by sending a request to the resource representing radio node information, e.g. E-CGI as per ETSI TS 129 171 [8].
- 2) The Location Service returns a response with message body including the list of radio nodes currently associated with the MEC host and the location of each radio node.

5.3.8 UE Tracking Subscribe

The UE Tracking Subscribe is the procedure for applications to receive notifications of UE information updates for a specified UE. The UE information updates can be triggered by status changes, e.g. the UE handing over between cells. In this procedure, the Location Service will continue to report the subscribed information until the subscription is cancelled.

The UE Tracking Subscribe procedure is illustrated in figure 5.3.8-1.

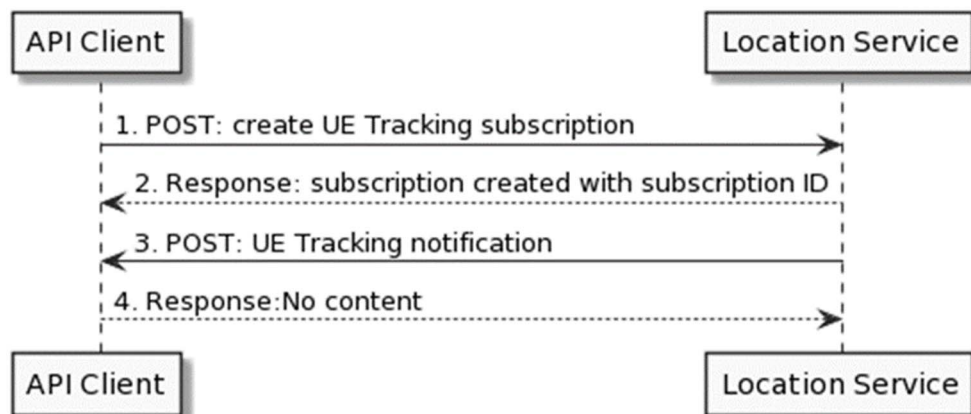


Figure 5.3.8-1: Flow of UE Tracking Subscribe

- 1) The MEC application subscribes to the UE Tracking notification by requesting the creation of a resource containing all subscription details, which includes the UE identifier, e.g. UE IP address, and a callbackURL for receiving the UE location.
- 2) The Location Service returns a response with resource URI containing the subscriptionId.
- 3) The Location Service reports the subscribed information to the MEC application by sending a message with the request body containing the UE Tracking notification to the callbackURL, which includes the UE location information.
- 4) The MEC application returns a response with the code 204.

5.3.9 UE Distance Lookup

The UE Distance Lookup is the procedure for applications acquiring the current distance of a specific UE to a geographical location, or another UE.

The UE Distance Lookup procedure is illustrated in figure 5.3.9-1.

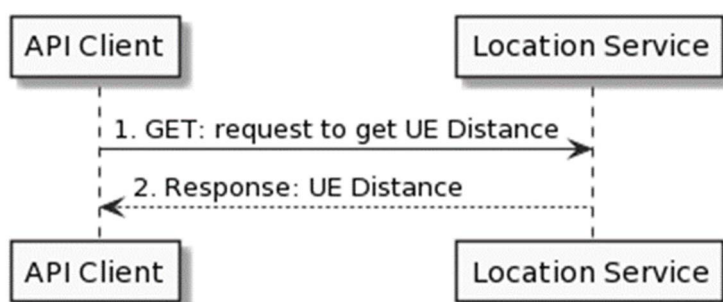


Figure 5.3.9-1: Flow of UE Distance Lookup

- 1) The MEC application looks up an UE distance by sending a request to the resource representing the UE distance, which includes the two UE identities, e.g. UE IP address, or a single UE identifier plus the coordinates of the location to measure from.
- 2) The Location Service returns a response with a message body including the distance information if the UE location lookup is accepted.

5.3.10 UE Distance Subscribe

The UE Distance Subscribe is the procedure for applications acquiring up-to-data distance of a specific UE to a geographical location, or another UE. In this procedure, the Location Service will continue to report the subscribed information until the subscription is cancelled.

The UE Distance Subscribe procedure is illustrated in figure 5.3.10-1.

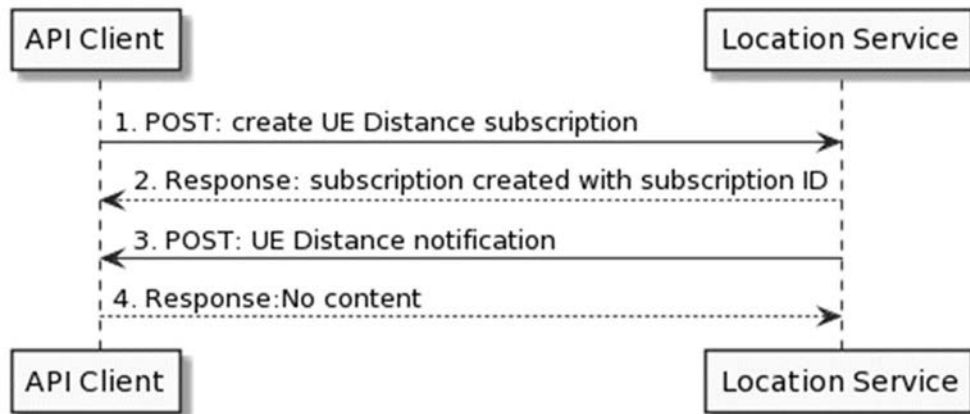


Figure 5.3.10-1: Flow of UE Distance Subscribe

- 1) The MEC application subscribes to UE distance notification by requesting the creation of a resource containing the subscription details, with a callbackURL for receiving the UE distance. It also includes the UE identities, e.g. UE IP address, between which to calculate the distances, the required accuracy and the minimum interval between notifications.
- 2) The Location Service returns a response with resource URI containing the subscriptionId.
- 3) The Location Service reports the up-to-data subscribed information to the MEC application by sending a message with the message body containing the UE Distance notification to the callbackURL, which includes distance information.
- 4) The MEC application returns a response with the code 204.

5.3.11 UE Area Subscribe

The UE Area Subscribe is the procedure for applications acquiring UE movement notifications in relation to a geographic area. In this procedure, the Location Service will continue to report the subscribed information until the subscription is cancelled.

The UE Area Subscribe procedure is illustrated in figure 5.3.11-1.

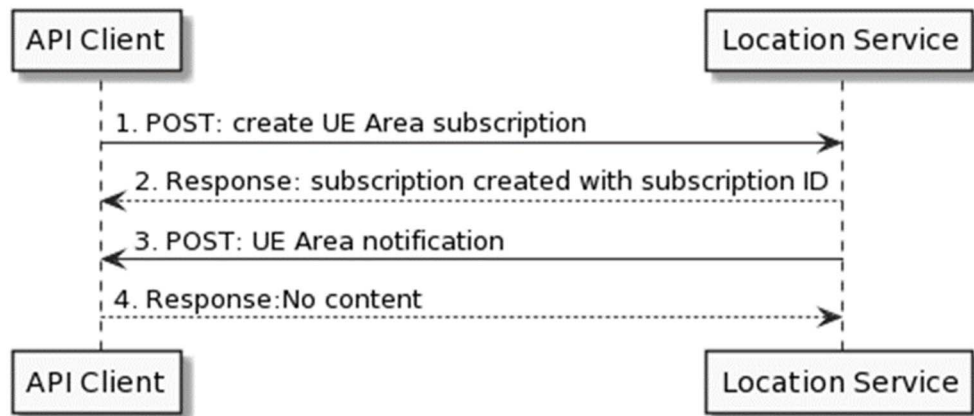


Figure 5.3.11-1: Flow of UE Area Subscribe

- 1) The MEC application subscribes to UE Area notifications by requesting the creation of a resource containing the subscription details, which includes the UE identities, e.g. UE IP address, and a callbackURL for receiving the UE Area notification. Also included is the area definition, the required accuracy, the minimum interval between notifications and whether notifications should be generated on entering, or leaving the area, or both.
- 2) The Location Service returns a response with resource URI containing the subscriptionId.
- 3) The Location Service reports the up-to-date subscribed information to the MEC application by sending a message with the message body containing the UE Area notification to the callbackURL, which includes area information.
- 4) The MEC application returns a response with the code 204.

6 Data Model

6.1 Introduction

Data types from the APIs defined in [5] and in [6] are reused. The following application instructions describe the mapping between the terminology used in the OMA specifications and the terminologies used in ETSI MEC and ETSI TS 129 171 [8].

The elements specified in OMA APIs' data models in [5] and [6] shall be represented as a JSON object. Moreover, the elements therein with upperbound cardinality greater than 1, e.g. "0.. N", "1.. N" etc., shall be represented as a JSON array.

6.2 Resource data types

6.2.1 Zonal presence data types

The data structures to be used in resource representations for the Location Service API consist in the following data types defined in [5], clause 5:

- ZoneList
A type containing list of zones' information;

- ZoneInfo
A type containing zone information;
- AccessPointList
A type containing list of access points' information;
- AccessPointInfo
A type containing access point information;
- UserList
A type containing list of users' information;
- UserInfo
A type containing user information, expanded by the present document to optionally include location information.

6.2.1A Terminal location data types

In addition, further data types as defined in [6], clause 5, are offered:

- TerminalLocation
A type containing device address, retrieval status and location information;
- TerminalDistance
A type containing information about the distance from a terminal to a location or between two terminals.

6.2.2 Type: UserInfo

This type represents the information related to a user attached to an access point associated to the MEC host, such access point is in scope of the Location Service instance.

The UserInfo data type extends the UserInfo data type defined in [5], and its elements shall follow the notations provided in table 6.2.2-1.

Table 6.2.2-1: Definition of type UserInfo

Attribute name	Data type	Cardinality	Description
address	AnyURI	1	Address of user (e.g. 'sip' URI, 'tel' URI, 'acr' URI) currently on the access point, see note 1.
accessPointId	String	1	The identity of the access point the user is currently on, see note 1.
zoneId	String	1	The identity of the zone the user is currently within, see note 1.
resourceURL	AnyURI	1	Self-referring URL, see note 1.
timeStamp	TimeStamp	1	Date and time that location was collected.
locationInfo	LocationInfo	0..1	The geographical coordinates where the user is, see note 2.
contextLocationInfo	String	0..1	Contextual information of a user location (e.g. aisle, floor, room number, etc.).
ancillaryInfo	String	0..1	Reserved for future use.
NOTE 1: As specified in [5], clause 5.2.2.7.			
NOTE 2: As specified in [5], clause 5.2.2.5.			

The address sub-element supports the representation of the UE's IP address, as specified in clause 6.6.2.

6.3 Subscription data types

6.3.1 Zonal presence data types

The data structures to be used in subscriptions for the Location Service API consist in the following data types defined in [5], clause 5:

- **NotificationSubscriptionList**
A type containing list of zonal traffic, user tracking, and zone status subscription(s), see clause 6.3.3;
- **ZonalTrafficSubscription**
A type containing zonal traffic subscription. A root element named notificationSubscriptionList of type NotificationSubscriptionList is allowed in response bodies;
- **UserTrackingSubscription**
A type containing user traffic subscription. A root element named notificationSubscriptionList of type NotificationSubscriptionList is allowed in response bodies;
- **ZoneStatusSubscription**
A type containing zonal status subscription. A root element named notificationSubscriptionList of type NotificationSubscriptionList is allowed in response bodies.

6.3.2 Terminal Location data types

In addition, further data types as defined in [6], clause 5, are supported:

- **NotificationSubscriptionList**
A type containing list of circle, periodic and distance subscription(s), see clause 6.3.3;
- **CircleNotificationSubscription**
A type containing a terminal location within a circle subscription;
- **PeriodicNotificationSubscription**
A type containing a periodic update of terminal location subscription;
- **DistanceNotificationSubscription**
A type containing a geographical relationship between terminals subscription.

6.3.3 Type: NotificationSubscriptionList

This type contains a list of subscriptions. The NotificationSubscriptionList data type merges the NotificationSubscriptionList data types defined in [5], clause 5.2.2.8 and in [6], clause 5.2.7 and its elements shall follow the notations provided in table 6.3.3-1.

Table 6.3.3-1: Definition of type NotificationSubscriptionList

Attribute name	Data type	Cardinality	Description
zonalTrafficSubscription	ZonalTrafficSubscription	0.. N	Collection of ZonalTrafficSubscription elements, see note 1.
userTrackingSubscription	UserTrackingSubscription	0.. N	Collection of UserTrackingSubscription elements, see note 1.
zoneStatusSubscription	ZoneStatusSubscription	0.. N	Collection of ZoneStatusSubscription elements, see note 1.
circleNotificationSubscription	CircleNotificationSubscription	0.. N	Collection of CircleNotificationSubscription elements, see note 2.
periodicNotificationSubscription	PeriodicNotificationSubscription	0.. N	Collection of PeriodicNotificationSubscription elements, see note 2.
distanceNotificationSubscription	DistanceNotificationSubscription	0.. N	Collection of DistanceNotificationSubscription elements, see note 2.
resourceURL	AnyURI	1	Self-referring URL, see note 1.
NOTE 1: As specified in [5], clause 5.2.2.8.			
NOTE 2: As specified in [6], clause 5.2.2.7.			

6.4 Notifications data types

6.4.1 Zonal presence data types

The data structures that define notifications for the Location Service API consist in the following data types defined in [5], clause 5:

- ZonalPresenceNotification
A type containing zonal presence notification, applicable to both zonal traffic and user tracking subscriptions;
- ZoneStatusNotification
A type containing zone status notification, applicable to zonal status subscriptions.

6.4.2 Terminal location data types

In addition, further data types as defined in [6], clause 5, are supported:

- SubscriptionNotification
A type containing a terminal location notification, applicable to circle, periodic and distance subscriptions.

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 can neither be 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 type 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: LocationInfo

This data type extends the LocationInfo data type defined in [5], clause 5.2.2.5, and its elements shall follow the notations provided in table 6.5.3-1. The World Geodetic System 1984 (WGS 84) [13] shall be used as the base coordinate reference system for the coding of location information. The listed attributes are aligned to those defined for the Universal Geographical Area Description (GAD) specified in ETSI TS 123 032 [14]. However, attributes are not limited to the reporting of location information obtained from a 3GPP based access network. For instance, location information from WLAN may also include uncertainty information, IETF RFC 6225 [15]. In addition, the source of the location information is not limited to GNSS reports from the end user device and may be provided an entity associated with the underlying access network, or even by a specific MEC application.

Table 6.5.3-1: Definition of type LocationInfo

Attribute name	Data type	Cardinality	Description
latitude	Float	1..N	Location latitude, expressed in the range -90° to +90°. Cardinality greater than one only if "shape" equals 7
longitude	Float	1..N	Location longitude, expressed in the range -180° to +180°. Cardinality greater than one only if "shape" equals 7
altitude	Float	0..1	Location altitude relative to the WGS84 ellipsoid surface.
accuracy	UnsignedInt	0..1	Horizontal accuracy / (semi-major) uncertainty of location provided in meters, as defined in ETSI TS 123 032 [14]. Present only if "shape" equals 4, 5 or 6
accuracySemiMinor	UnsignedInt	0..1	Horizontal accuracy / (semi-major) uncertainty of location provided in meters, as defined in ETSI TS 123 032 [14]. Present only if "shape" equals 4, 5 or 6
accuracyAltitude	UnsignedInt	0..1	Altitude accuracy / uncertainty of location provided in meters, as defined in ETSI TS 123 032 [14]. Present only if "shape" equals 3 or 4
orientationMajorAxis	UnsignedInt	0..1	Angle of orientation of the major axis, expressed in the range 0° to 180°, as defined in ETSI TS 123 032 [14]. Present only if "shape" equals 4 or 6
confidence	UnsignedInt	0..1	Confidence by which the position of a target entity is known to be within the shape description, expressed as a percentage and defined in ETSI TS 123 032 [14]. Present only if "shape" equals 1, 4 or 6
innerRadius	UnsignedInt	0..1	Present only if "shape" equals 6
uncertaintyRadius	UnsignedInt	0..1	Present only if "shape" equals 6
offsetAngle	UnsignedInt	0..1	Present only if "shape" equals 6
includedAngle	UnsignedInt	0..1	Present only if "shape" equals 6
shape	Enum (inlined)	1	Shape information, as detailed in ETSI TS 123 032 [14], associated with the reported location coordinate: 1 = ELLIPSOID_ARC 2 = ELLIPSOID_POINT 3 = ELLIPSOID_POINT_ALTITUDE 4 = ELLIPSOID_POINT_ALTITUDE_UNCERT_ELLIPSOID 5 = ELLIPSOID_POINT_UNCERT_CIRCLE 6 = ELLIPSOID_POINT_UNCERT_ELLIPSE 7 = POLYGON
velocity	Structure (inlined)	0..1	Structure with attributes relating to the target entity's velocity, as defined in ETSI TS 123 032 [14]
>velocityType	Enum (inlined)	1	Velocity information, as detailed in ETSI TS 123 032 [14], associated with the reported location coordinate: 1 = HORIZONTAL 2 = HORIZONTAL_VERTICAL 3 = HORIZONTAL_UNCERT 4 = HORIZONTAL_VERTICAL_UNCERT

Attribute name	Data type	Cardinality	Description
>bearing	UnsignedInt	1	Bearing, expressed in the range 0° to 360°, as defined in ETSI TS 123 032 [14]
>horizontalSpeed	UnsignedInt	1	Horizontal speed, expressed in km/h and defined in ETSI TS 123 032 [14]
>uncertainty	UnsignedInt	0..1	Horizontal uncertainty, as defined in ETSI TS 123 032 [14]. Present only if "velocityType" equals 3 or 4
>verticalSpeed	Int	0..1	Vertical speed, expressed in km/h and defined in ETSI TS 123 032 [14]. Present only if "velocityType" equals 2 or 4
>verticalUncertainty	UnsignedInt	0..1	Vertical uncertainty, as defined in ETSI TS 123 032 [14]. Present only if "velocityType" equals 4

6.6 Mapping of Identifiers

6.6.1 accessPointId

With respect to identifiers assigned to access points in a cellular deployment, the accessPointId shall equal to an E-CGI concatenated with a Cell Portion ID as defined in ETSI TS 129 171 [8], i.e.:

- accessPointId = <E-CGI><Cell Portion ID>,

where the Cell Portion ID may not exist, if the cell does not contain portions.

6.6.2 userId

The Location Service allows using the UE's IP address as a user identifier. The OMA API [5] does not explicitly list IP address as a user identifier option (i.e. as part of the "address" Element of UserInfo). Therefore to satisfy the requirements of the Location Service, if a user identifier of type anyURI is in the form of an IP address, the "acr" scheme shall be used, followed by the UE's IP address, i.e.:

- address = acr:<UE's IP address>,

where the format of the UE's IP address shall be an IP literal encapsulated within square brackets or an IPv4 address in dotted-decimal form.

NOTE: The "acr" scheme can also be used to convey user identifiers other than an IP address, e.g. an alias.

7 API definition

7.1 Introduction

This clause defines the resources and operations of the Location Service API.

The resources and the operations defined by the Open Mobile Alliance Zonal Presence API [5] and Terminal Location API [6] are reused. The relation with the resources defined in [5] and [6] and those used in this API is described in clause 7.2 of the present document. The application instructions in clause 7.3 describe the mapping between the service descriptions of the MEC Location Service API (refer to clause 5) and the operations of the OMA Zonal Presence API [5] and of the OMA Terminal Location API [6].

7.2 Global definitions and relation with OMA resources

All the resources defined in [5], clause 6 and in [6], clause 6 are used in this API except the resource of individual user information in [5], clause 6.6 and the resource of terminal location in [6], clause 6.1.

The "URL Base URL" defined in [5] as:

http://{serverRoot}/zonalPresence/{apiVersion}/{zoneOwnerId}

and defined in [6] as:

http://{serverRoot}/location/{apiVersion}

shall be replaced by the following URI root:

{apiRoot}/location/{apiVersion}/

"apiRoot" is discovered using the service registry, it includes the scheme ("http" or "https"), host and optional port, and an optional prefix string. The "apiVersion" shall be set to "v2" for the current version of the specification. It includes the scheme ("http" or "https"), host and optional port, and an optional prefix string.

This API shall support HTTP over TLS (also known as HTTPS) using TLS version 1.2 as defined by IETF RFC 5246 [10]. TLS 1.3 (including the new specific requirements for TLS 1.2 implementations) defined by IETF RFC 8446 [16] should be supported. HTTP without TLS shall not be used. Versions of TLS earlier than 1.2 shall neither be supported nor used.

All resource URIs defined in [5], clause 6 and in [6], clause 6 are re-defined for this API relative to the above root URI, with the addition of "/queries" following "{apiVersion}" for the query related resources defined in [5]. Therefore, the URI variable "serverRoot" defined for the resources in [5] and [6] are not used in this API, and shall be discarded. In addition, "zoneOwnerId" defined for the resources in [5] is not used in this API, and shall also be discarded.

The content format of JSON shall be supported and signalled by setting the content type header to "application/json".

This API supports additional application-related error information to be provided in the HTTP response when an error occurs. See clause 6.15 of ETSI GS MEC 009 [4] for more information.

This API shall require the use of the OAuth 2.0 client credentials grant type according to IETF RFC 6749 [11] with bearer tokens according to IETF RFC 6750 [12]. See ETSI GS MEC 009 [4], clause 6.16 for more information. How the token endpoint and client credentials are provisioned into the MEC applications is out of scope of the present document.

The resource URI structure presented in [5], clause 5.1 and in [6] clause 5.1 and the summary tables therein shall apply for this API as well, except for the URI root, which shall be replaced by the one defined above and the addition of "/queries" where applicable. The resulting resource tree is illustrated in figure 7.2-1.

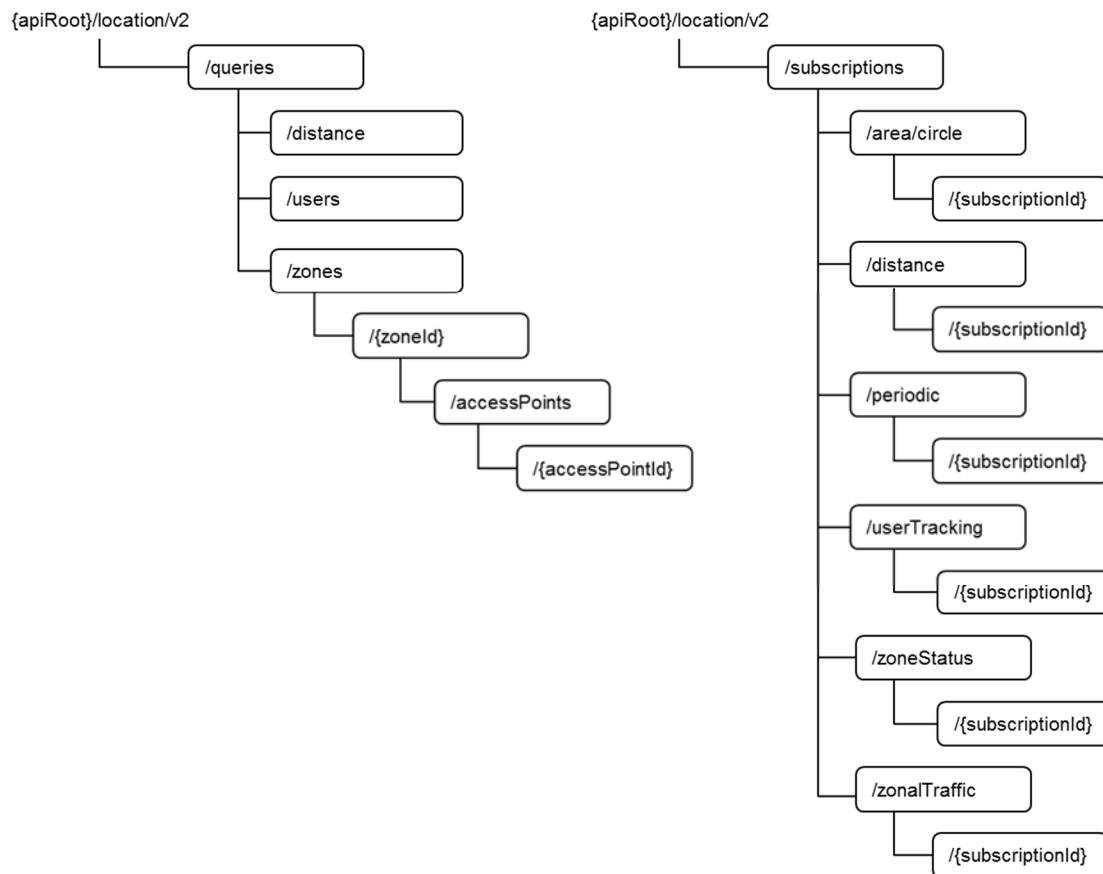


Figure 7.2-1: Resource URI structure of the Location API

7.3 Mapping of services to API functions

7.3.1 Introduction

This clause describes how the resources and operations defined by the OMA Zonal Presence API [5] and by the OMA Terminal Location API [6] shall be used in order to realize the operations of the present API.

7.3.2 UE Location Lookup

7.3.2.1 Definition

The UE Location Lookup (clause 5.3.2) of a specific UE or a group of UEs shall be realized by requesting the resource "User list" defined in [5], clause 6.5 and applying the "GET" method defined in [5], clause 6.5.3.

The request URI shall be:

{apiRoot}/location/v2/queries/users

where {apiRoot} is defined in clause 7.2.

The "GET" method defined in [5], clause 6.5.3 shall support the URI query parameters indicated in table 7.3.2.1-1, which is obtained by extending the table specified in [5], clause 6.5.3.

Table 7.3.2.1-1: URI query parameters supported by the GET method on the "User List" resource

Name	Data type	Cardinality	Remarks
zoneId	String	0..N	Identifier of zone (e.g. zone001)
accessPointId	String	0..N	Identifier of access point (e.g. 00101000000000000000000000000001)
address	AnyUri	0..N	Address of users (e.g. 'sip' URI, 'tel' URI, 'acr' URI). See note
NOTE: In order to perform the UE Location Lookup for a specific UE or a group of specific UEs, the filtering parameter "address" shall be used.			

The response shall contain an instance of the UserInfo data type, associated to the user that matches with the address provided in the request URI. If no locationInfo or contextLocationInfo sub-elements are included in the returned UserInfo instance, then the location may be inferred from the resource "Individual access point information" (refer to [5], clause 6.4), requested using the accessPointId sub-element included in UserInfo as variable in the request URI of the "Individual access point information" resource.

7.3.2.2 Example: Two steps method to retrieve the geographical coordinates of an access point a specific user is connected to (informative)

In this example, the client issues a request to fetch the user location of a certain user identified by its IP address. If the userInfo element in the response contains any of the locationInfo or contextLocationInfo sub-elements, then the procedure may stop, otherwise an additional request for the access point information may be issued.

Request 1

```
GET /exampleAPI/location/v2/queries/users?address=acr%3A10.0.0.1 HTTP/1.1
Host: example.com
Accept: application/json
```

Response 1

```
HTTP/1.1 200 OK
Date: Sun, 01 Jan 2017 00:38:59 GMT
Content-Type: application/json
Content-Length: nnnn

{"userList": {
  "user": [
    {
      "address": "acr:10.0.0.1",
      "accessPointId": "00101000000000000000000000000001",
      "zoneId": "zone01",
      "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/users?address=acr%3A10.0.0.1",
      "timestamp": {
        "seconds": 1483231138,
        "nanoSeconds": 0
      }
    }
  ],
  "resourceURL": "http://example.com/exampleAPI/location/v2/queries/users"
}}
```

Request 2

```
GET /exampleAPI/location/v2/queries/zones/zone01/accessPoints/00101000000000000000000000000001
HTTP/1.1
Host: example.com
Accept: application/json
```

Response 2

```
HTTP/1.1 200 OK
Date: Sun, 01 Jan 2017 00:38:59 GMT
Content-Type: application/json
Content-Length: nnnn

{"accessPointInfo": {
  "accessPointId": "00101000000000000000000000000001",
  "locationInfo": {
    "latitude": "80.123",
```

```

        "longitude": "70.123",
        "altitude": "10.0",
        "accuracy": "0"
    },
    "connectionType": "Macro",
    "operationStatus": "Serviceable",
    "numberOfUsers": "15",
    "interestRealm": "LA",
    "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/zones/zone01/accessPoints/ap001"
}}

```

7.3.3 UE Information Lookup

7.3.3.1 Definition

The UE Information Lookup (clause 5.3.3) shall be realized by requesting the resource "User list" defined in [5], clause 6.5 and applying the "GET" method defined in [5], clause 6.5.3.

The request URI shall be:

{apiRoot}/location/v2/queries/users

where {apiRoot} is defined in clause 7.2.

The request may be further refined by filtering the request URI by using the "zoneId" and/or the "accessPointId" query parameters, as defined in table 7.3.2.1-1.

The response shall contain an instance of the UserList data type, which in turn includes all the UserInfo instances that match the filtering criteria specified in the request.

7.3.3.2 Example: Retrieve UE Information for all the users (informative)

In this example, the client issues a request to retrieve a list of all the users connected to all the access points associated to all the zones.

Request

```

GET /exampleAPI/location/v2/queries/users HTTP/1.1
Host: example.com
Accept: application/json

```

Response

```

HTTP/1.1 200 OK
Date: Sun, 01 Jan 2017 00:38:59 GMT
Content-Type: application/json
Content-Length: nnnn

{"userList": {
  "user": [
    {
      "address": "acr:10.0.0.1",
      "accessPointId": "00101000000000000000000000000001",
      "zoneId": "zone01",
      "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/users?address=acr%3A10.0.0.1",
      "timestamp": {
        "seconds": 1483231138,
        "nanoSeconds": 0
      }
    },
    {
      "address": "acr:10.0.0.2",
      "accessPointId": "00101000000000000000000000000001",
      "zoneId": "zone01",
      "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/users?address=acr%3A10.0.0.2",
      "timestamp": {
        "seconds": 1483231138,
        "nanoSeconds": 0
      }
    }
  ]
},

```



```

    {
      "address": "acr:10.0.0.3",
      "accessPointId": "00101000000000000000000000000002",
      "zoneId": "zone01",
      "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/users?address=acr%3A10.0.0.3",
      "timestamp": {
        "seconds": 1483231138,
        "nanoSeconds": 0
      }
    },
    {
      "address": "acr:10.0.0.4",
      "accessPointId": "00101000000000000000000000000004",
      "zoneId": "zone02",
      "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/users?address=acr%3A10.0.0.4",
      "timestamp": {
        "seconds": 1483231138,
        "nanoSeconds": 0
      }
    },
    {
      "address": "acr:10.0.0.5",
      "accessPointId": "00101000000000000000000000000005",
      "zoneId": "zone02",
      "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/users?address=acr%3A10.0.0.5",
      "timestamp": {
        "seconds": 1483231138,
        "nanoSeconds": 0
      }
    }
  ],
  "resourceURL": "http://example.com/exampleAPI/location/v2/queries/users"
}

```

7.3.3.3 Example: Retrieve UE Information for the users in a zone (informative)

In this example, the client issues a request to retrieve a list of all the users connected to all the access points associated to the zone specified in the request.

Request

```

GET /exampleAPI/location/v2/queries/users?zoneId=zone01 HTTP/1.1
Host: example.com
Accept: application/json

```

Response

```

HTTP/1.1 200 OK
Date: Sun, 01 Jan 2017 00:38:59 GMT
Content-Type: application/json
Content-Length: nnnn

```

```

{"userList": {
  "user": [
    {
      "address": "acr:10.0.0.1",
      "accessPointId": "00101000000000000000000000000001",
      "zoneId": "zone01",
      "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/users?address=acr%3A10.0.0.1",
      "timestamp": {
        "seconds": 1483231138,
        "nanoSeconds": 0
      }
    },
    {
      "address": "acr:10.0.0.2",
      "accessPointId": "00101000000000000000000000000001",
      "zoneId": "zone01",
      "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/users?address=acr%3A10.0.0.2",
      "timestamp": {
        "seconds": 1483231138,
        "nanoSeconds": 0
      }
    }
  ]
}

```


{apiRoot}/location/v2/subscriptions/userTracking

where {apiRoot} is defined in clause 7.2.

The request shall contain an instance of the UserTrackingSubscription data type, which includes the address of the user to monitor, and optionally the events intended for tracking (e.g. zone enter, zone exit, zone transfer). The response shall contain an instance of the UserTrackingSubscription data type and the location of the created resource.

NOTE: The subscription may be updated as per the method defined in [5], clause 5.3.3.

The notification messages sent to a client subscribed to UE Location Subscribe shall follow the notification procedure defined in [5], clause 6.13, by applying the method "POST" therein, see [5], clause 6.13.5. The body of the notification message shall contain an instance of the ZonalPresenceNotification data type.

7.3.4.2 Example: Create a UE Location subscription (informative)

In this example, the client issues a request to create a UE Location subscription for a certain user identified by its IP address. The subscription is meant to receive a notification about the "transferring" user event, i.e. when the requested UE transfers from one access point to another within the zone. If no user event is included in the request, then the subscription is created for all the events (zone enter, zone exit and zone transfer).

Request

```
POST /exampleAPI/location/v2/subscriptions/userTracking HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: example.com

{"userTrackingSubscription": {
  "clientCorrelator": "0123",
  "callbackReference": {"notifyURL": "http://clientApp.example.com/location_notifications/123456"},
  "address": "acr:10.0.0.1",
  "userEventCriteria" : ["Transferring"]}
}
```

Response

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: http://example.com/exampleAPI/location/v2/subscriptions/userTracking/subscription123
Content-Length: nnnn
Date: Sun, 01 Jan 2017 00:38:59 GMT

{"userTrackingSubscription": {
  "clientCorrelator": "0123",
  "resourceURL":
    "http://example.com/exampleAPI/location/v2/subscriptions/userTracking/subscription123",
  "callbackReference": {"notifyURL": "http://clientApp.example.com/location_notifications/123456"},
  "address": "acr:10.0.0.1",
  "userEventCriteria" : ["Transferring"]}
}
```

7.3.4.3 Example: Client notification about UE Location change (informative)

In this example, the client receives a notification of a location change of the requested UE, identified by its IP address. The notification relates to the "transferring" user event, i.e. when the requested UE transfers from one access point to another within the zone.

Request

```
POST /location_notifications/123456 HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: clientApp.example.com

{"zonalPresenceNotification": {
  "clientCorrelator": "0123",
```

```

"zoneId": "zone01",
"address": "acr:10.0.0.1",
"interestRealm": "LA",
"userEventType": "Transferring"
"currentAccessPointId": "00101000000000000000000000000002",
"previousAccessPointId": "00101000000000000000000000000001",
"timestamp": {
  "seconds": 1483231138,
  "nanoSeconds": 0},
"link": {
  "rel": "UserTrackingSubscription",
  "href":
    "http://example.com/exampleAPI/location/v2/subscriptions/userTracking/subscription123"
}}}

```

Response

```

HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT

```

7.3.5 UE Information Subscribe

7.3.5.1 Definition

The UE Information Subscribe (clause 5.3.5) shall follow the "Zonal traffic notification" procedure defined in [5], clause 5.3.2. In order to subscribe, the client shall request the creation of the resource "Zonal traffic change notification subscriptions" defined in [5], clause 6.7, applying the "POST" method as in [5], clause 6.7.5. The request URI shall be:

{apiRoot}/location/v2/subscriptions/zonalTraffic

where {apiRoot} is defined in clause 7.2.

The request shall contain an instance of the zonalTrafficSubscription data type, which shall include the identifier of the zone to monitor, and optionally the interest realm of the access points within the zone and the user events intended for tracking (e.g. zone enter, zone exit, zone transfer). The response shall contain an instance of the zonalTrafficSubscription data type and the location of the created resource. Successfully created subscriptions may be updated by applying the method "PUT" defined in [5], clause 5.3.2.

The notification messages sent to a client subscribed to UE Information Subscribe shall follow the notification procedure defined in [5], clause 6.13, by applying the method "POST" therein, see [5], clause 6.13.5. The body of the notification message shall contain an instance of the ZonalPresenceNotification data type.

7.3.5.2 Example: Create a UE Information subscription (informative)

In this example, the client issues a request to create a UE Information subscription for a certain zone. The subscription is meant to receive a notification about the "transferring" user event, i.e. when any UE transfers from one access point to another within the zone. If no user event is included in the request, then the subscription is created for all the events (zone enter, zone exit and zone transfer).

Request

```

POST /exampleAPI/location/v2/subscriptions/zonalTraffic HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: example.com

{"zonalTrafficSubscription": {
  "clientCorrelator": "0123",
  "callbackReference": {"notifyURL": "http://clientApp.example.com/location_notifications/123456"},
  "zoneId": "zone01",
  "interestRealm": ["LA"],
  "userEventCriteria": ["Transferring"]}
}

```

Response

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: http://example.com/exampleAPI/location/v2/subscriptions/zonalTraffic/subscription123
Content-Length: nnnn
Date: Sun, 01 Jan 2017 00:38:59 GMT

{"zonalTrafficSubscription": {
  "clientCorrelator": "0123",
  "resourceURL":
    "http://example.com/exampleAPI/location/v2/subscriptions/zonalTraffic/subscription123",
  "callbackReference": {"notifyURL": "http://clientApp.example.com/location_notifications/123456"},
  "zoneId": "zone01",
  "interestRealm": ["LA"],
  "userEventCriteria": ["Transferring"]}
}}
```

7.3.5.3 Example: Client notification about UE Information change (informative)

In this example, the client receives a notification of a location change of a UE, identified by its IP address, within the zone. The notification relates to the "transferring" user event, i.e. when the UE transfers from one access point to another within the zone.

Request

```
POST /location_notifications/123456 HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: clientApp.example.com

{"zonalPresenceNotification": {
  "clientCorrelator": "0123",
  "zoneId": "zone01",
  "address": "acr:10.0.0.2",
  "interestRealm": "LA",
  "userEventType": "Transferring",
  "currentAccessPointId": "00101000000000000000000000000002",
  "previousAccessPointId": "00101000000000000000000000000001",
  "timestamp": {
    "seconds": 1483231138,
    "nanoSeconds": 0},
  "link": {
    "rel": "UserTrackingSubscription",
    "href":
      "http://example.com/exampleAPI/location/v2/subscriptions/userTracking/subscription123"
  }}
}}
```

Response

```
HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT
```

7.3.6 Subscribe Cancellation

7.3.6.1 Definition

The Subscribe Cancellation (clause 5.3.6) shall follow the cancellation methods defined in the "User tracking notification" procedure defined in [5], clause 5.3.3, in the "Zonal traffic notification" procedure defined in [5], clause 5.3.2, in the "Periodic location notification" procedure defined in [6], clause 5.3.4, in the "Distance location notification" procedure defined in [6], clause 5.3.6 and in "Area (circle) location notification" procedure defined in [6], clause 5.3.5, respectively to cancel the UE Location subscription, to cancel the UE Information subscription, to cancel the UE Tracking subscription, to cancel the UE Distance subscription and to cancel the UE area subscription.

It shall be realized by requesting the deletion of the corresponding subscription resource, represented by the subscription's identifier, and shall be returned by the server to the client as a {subscriptionId} variable upon the subscription creation. The request URI to delete a UE Location subscription shall be:

{apiRoot}/location/v2/subscriptions/userTracking/{subscriptionId}

and the client shall apply the method "DELETE" to the resource, as specified in [5], clause 6.10.6. The request URI to delete a UE Information subscription shall be:

{apiRoot}/location/v2/subscriptions/zonalTraffic/{subscriptionId}

and the client shall apply the method "DELETE" to the resource, as specified in [5], clause 6.8.6. In both the URIs above, {apiRoot} is defined in clause 7.2 and {subscriptionId} shall be the subscription's identifier. The request URI to delete a UE Tracking subscription shall be:

{apiRoot}/location/v2/subscriptions/periodic/{subscriptionId}

and the client shall apply the method "DELETE" to the resource, as specified in [6], clause 6.4.6. The request URI to delete a UE Distance subscription shall be:

{apiRoot}/location/v2/subscriptions/distance/{subscriptionId}

and the client shall apply the method "DELETE" to the resource, as specified in [6], clause 6.8.6. The request URI to delete a UE Area subscription shall be:

{apiRoot}/location/v2/subscriptions/area/circle/{subscriptionId}

and the client shall apply the method "DELETE" to the resource, as specified in [6], clause 6.6.6.

7.3.6.2 Example: Cancelling a UE Location subscription (informative)

In this example, the client issues a request to cancel a UE Location subscription.

Request

```
DELETE /exampleAPI/location/v2/subscriptions/userTracking/subscription123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response

```
HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT
```

7.3.6.3 Example: Cancelling a UE Information subscription (informative)

In this example, the client issues a request to cancel a UE Information subscription.

Request

```
DELETE /exampleAPI/location/v2/subscriptions/zonalTraffic/subscription123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response

```
HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT
```

7.3.6.4 Example: Cancelling a UE Tracking subscription (informative)

In this example, the client issues a request to cancel a UE Tracking subscription.

Request

```
DELETE /exampleAPI/location/v2/subscriptions/periodic/subscription123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response

HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT

7.3.6.5 Example: Cancelling a UE Distance subscription (informative)

In this example, the client issues a request to cancel a UE Distance subscription.

Request

```
DELETE /exampleAPI/location/v2/subscriptions/distance/subscription123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response

HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT

7.3.6.6 Example: Cancelling a UE Area subscription (informative)

In this example, the client issues a request to cancel a UE Area subscription.

Request

```
DELETE /exampleAPI/location/v2/subscriptions/area/circle/subscription123 HTTP/1.1
Accept: application/json
Host: example.com
```

Response

HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT

7.3.7 Radio Node Location Lookup

7.3.7.1 Definition

The Radio Node Location Lookup (clause 5.3.7) shall be realized by requesting the resource "Access point list" defined in [5], clause 6.3 and applying the method "GET" defined in [5], clause 6.3.3.

The request URI shall be:

{apiRoot}/location/v2/queries/zones/{zoneId}/accessPoints

where {apiRoot} is defined in clause 7.2 and {zoneId} is the identifier of the zone which the access point is associated to (e.g. zone01).

The request may be further refined by filtering the request URI by using the "interestRealm" query parameter, as defined in [5], clause 6.3.3.

The response shall contain an instance of the AccessPointList data type, which in turn shall include all the AccessPointInfo instances for all the access points associated to the zone specified in the request.

7.3.7.2 Example: Retrieve a list of radio nodes associated to a zone (informative)

In this example, the client issues a request to retrieve a list of all the access points associated to the zone specified in the request.

Request

```
GET /exampleAPI/location/v2/queries/zones/zone01/accessPoints HTTP/1.1
Host: example.com
Accept: application/json
```

Response

```
HTTP/1.1 200 OK
Date: Sun, 01 Jan 2017 00:38:59 GMT
Content-Type: application/json
Content-Length: nnnn
```

```
{
  "accessPointList": {
    "zoneId": "zone01",
    "accessPoint": [
      {
        "accessPointId": "00101000000000000000000000000001",
        "locationInfo": {
          "latitude": "80.123",
          "longitude": "70.123",
          "altitude": "10.0",
          "accuracy": "0"
        },
        "connectionType": "Macro",
        "operationStatus": "Serviceable",
        "numberOfUsers": "15",
        "interestRealm": "LA",
        "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/zones/zone01/accessPoints/ap001"
      },
      {
        "accessPointId": "00101000000000000000000000000002",
        "locationInfo": {
          "latitude": "82.123",
          "longitude": "72.123",
          "altitude": "12.0",
          "accuracy": "0"
        },
        "connectionType": "Femto",
        "operationStatus": "Unserviceable",
        "numberOfUsers": "5",
        "interestRealm": "NY",
        "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/zones/zone01/accessPoints/ap002"
      },
      {
        "accessPointId": "00101000000000000000000000000003",
        "locationInfo": {
          "latitude": "83.123",
          "longitude": "73.123",
          "altitude": "13.0",
          "accuracy": "0"
        },
        "connectionType": "Femto",
        "operationStatus": "Serviceable",
        "numberOfUsers": "25",
        "interestRealm": "DC",
        "resourceURL":
"http://example.com/exampleAPI/location/v2/queries/zones/zone01/accessPoints/ap003"
      }
    ],
    "resourceURL": "http://example.com/exampleAPI/location/v2/queries/zones/zone01/accessPoints"
  }
}
```

7.3.8 UE Tracking Subscribe

7.3.8.1 Definition

The UE Tracking Subscribe (clause 5.3.8) shall follow the "Periodic location notification" procedure defined in [6], clause 5.3.4. In order to subscribe, the client shall request the creation of the resource "Periodic location notification subscriptions" defined in [6], clause 6.3, applying the "POST" method as in [6], clause 6.3.5. The request URI shall be:

{apiRoot}/location/v2/subscriptions/periodic

where {apiRoot} is defined in clause 7.2.

The request shall contain an instance of the PeriodicNotificationSubscription data type, which includes the address of the user to monitor, the requested accuracy, the frequency of the notifications, and, optionally, the duration of the subscription. The response shall contain an instance of the PeriodicNotificationSubscription data type and the location of the created resource. Successfully created subscriptions may be updated by applying the method "PUT" defined in [6], clause 5.3.4.

The notification messages sent to a client subscribed to UE Tracking Subscribe shall follow the notification procedure defined in [6], clause 6.9, by applying the method "POST" therein, see [6], clause 6.9.5. The body of the notification message shall contain an instance of the SubscriptionNotification data type.

7.3.8.2 Example: Create a UE Tracking subscription (informative)

In this example, the client issues a request to create a UE Tracking subscription for a certain UE. The client expects to receive periodic notifications about the location of the specified UE at the time interval specified in the request.

Request

```
POST /exampleAPI/location/v2/subscriptions/periodic HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: example.com

{"periodicNotificationSubscription": {
  "address": "acr:10.0.0.1",
  "callbackReference": {
    "callbackData": "0123",[6]
    "notifyURL": "http://clientApp.example.com/location_notifications/123456"
  },
  "clientCorrelator": "0123",
  "frequency": "10",
  "requestedAccuracy": "10"
}}
```

Response

```
HTTP/1.1 201 Created
Content-Type: application/json
Location: http://example.com/exampleAPI/location/v2/subscriptions/periodic/subscription123
Content-Length: nnnn
Date: Sun, 01 Jan 2017 00:38:59 GMT

{"periodicNotificationSubscription": {
  "address": "acr:10.0.0.1",
  "callbackReference": {
    "callbackData": "0123",
    "notifyURL": "http://clientApp.example.com/location_notifications/123456"
  },
  "clientCorrelator": "0123",
  "frequency": "10",
  "requestedAccuracy": "10",
  "resourceURL": "http://example.com/exampleAPI/location/v2/subscriptions/periodic/subscription123"
}}
```

7.3.8.3 Example: Client notification about UE Tracking change (informative)

In this example, the client receives a notification of the location of the requested UE, identified by its IP address.

Request

```
POST /location_notifications/123456 HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: clientApp.example.com
```

```

{"subscriptionNotification": {
  "isFinalNotification": "false",
  "link": {
    "href": "http://example.com/exampleAPI/location/v2/subscriptions/periodic/sub123",
    "rel": "PerodicNotificationSubscription"
  },
  "terminalLocation": {
    "address": "acr:10.0.0.1",
    "currentLocation": {
      "accuracy": "100",
      "altitude": "1001.0",
      "latitude": "-80.86302",
      "longitude": "41.277306",
      "timestamp": {
        "seconds": 1483231138,
        "nanoSeconds": 0
      }
    }
  },
  "locationRetrievalStatus": "Retrieved"
}
}
}

```

Response

```

HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT

```

7.3.9 UE Distance Lookup

7.3.9.1 Definition

The UE Distance Lookup (clause 5.3.9) relating to a single UE shall be realized by requesting the resource "Terminal distance" defined in [6], clause 6.2, and applying the "GET" method defined in [6], clause 6.2.3 as well as the query parameters therein.

The request URI shall be:

{apiRoot}/location/v2/queries/distance

where {apiRoot} is defined in clause 7.2

The response shall contain an instance of the TerminalDistance data type, associated to the user that matches with the address, i.e. the identifier of the user (e.g. tel:+19585550199, acr:alias1234, acr:10.0.0.1), provided as a query parameter.

7.3.9.2 Example: Distance between a terminal and a location (informative)

In this example, the client issues a request to fetch the distance of a certain user, identified by its IP address, to a location.

Request

```

GET /exampleAPI/location/v2/queries/
  ?address=acr%3A10.0.0.1&latitude=43.6169128&longitude=7.0528493 HTTP/1.1
Host: example.com
Accept: application/json

```

Response

```

HTTP/1.1 200 OK
Date: Sun, 01 Jan 2017 00:38:59 GMT
Content-Type: application/json
Content-Length: nnnn

```

```

{" terminalDistance": {
  "distance": "123"
}}

```

7.3.9.3 Example: Distance between two terminals (informative)

In this example, the client issues a request to fetch the distance between two users.

Request

```
GET /exampleAPI/location/v2/queries/distance?address=acr%3A10.0.0.1&address=acr%3A10.0.0.2 HTTP/1.1
Host: example.com
Accept: application/json
```

Response

```
HTTP/1.1 200 OK
Date: Sun, 01 Jan 2017 00:38:59 GMT
Content-Type: application/json
Content-Length: nnnn
```

```
{"terminalDistance": {
  "distance": "123"
}}
```

7.3.10 UE Distance Subscribe

7.3.10.1 Definition

The UE Distance Subscribe (clause 5.3.10) shall follow the "Distance location notification" procedure defined in [6], clause 5.3.6. In order to subscribe, the client shall request the creation of the resource "Distance notification subscriptions" defined in [6], clause 6.7, applying the method "POST" as in [6], clause 6.7.5. The request URI shall be:

{apiRoot}/location/v2/subscriptions/distance

where {apiRoot} is defined in clause 7.2.

The request shall contain an instance of the DistanceNotificationSubscription data type, which includes the address of the user(s) to monitor. The response shall contain an instance of the DistanceNotificationSubscription data type and the location of the created resource. Successfully created subscriptions may be updated by applying the method "PUT" defined in [6], clause 5.3.6.

The notification messages sent to a client subscribed to UE Distance Subscribe shall follow the notification procedure defined in [6], clause 6.9, by applying the method "POST" therein, see [6], clause 6.9.5. The body of the notification message shall contain an instance of the SubscriptionNotification data type.

7.3.10.2 Example: Create a UE Distance subscription (informative)

In this example, the client issues a request to create a UE Distance subscription for a certain user identified by its IP address. The client expects to receive a notification when (all) the requested UE(s) is (are) within the specified distance.

Request

```
POST /exampleAPI/location/v2/subscriptions/distance HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: example.com
```

```
{"distanceNotificationSubscription": {
  "callbackReference": {
    "callbackData": "1234",
    "notifyURL": "http://application.example.com/notifications/distance_notifications/123456"
  },
  "checkImmediate": "true",
  "clientCorrelator": "0123",
  "criteria": "AllWithinDistance",
  "distance": "100",
  "frequency": "10",
  "monitoredAddress": [
    "acr:10.0.0.1",
```

```

    "acr:10.0.0.2"
  ],
  "referenceAddress": "acr:10.0.0.3",
  "trackingAccuracy": "10"
}}

```

Response

```

HTTP/1.1 201 Created
Content-Type: application/json
Location: http://example.com/exampleAPI/location/v2/subscriptions/distance/subscription123
Content-Length: nnnn
Date: Sun, 01 Jan 2017 00:38:59 GMT

{"distanceNotificationSubscription": {
  "callbackReference": {
    "callbackData": "1234",
    "notifyURL": "http://application.example.com/notifications/distance_notifications/123456"
  },
  "checkImmediate": "true",
  "clientCorrelator": "0123",
  "criteria": "AllWithinDistance",
  "distance": "100",
  "frequency": "10",
  "monitoredAddress": [
    "acr:10.0.0.1",
    "acr:10.0.0.2"
  ],
  "referenceAddress": "acr:10.0.0.3",
  "trackingAccuracy": "10",
  "resourceURL": "http://example.com/exampleAPI/location/v2/subscriptions/distance/subscription123"
}}

```

7.3.10.3 Example: Client notification about UE Tracking change (informative)

In this example, the client receives a notification of the location of the requested UEs, identified by their IP addresses.

Request

```

POST /location_notifications/123456 HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: clientApp.example.com

{"subscriptionNotification": {
  "distanceCriteria": "AllWithinDistance",
  "isFinalNotification": "false",
  "link": {
    "href": "http://example.com/exampleAPI/location/v2/subscriptions/periodic/subscription123",
    "rel": "DistanceNotificationSubscription"
  },
  "terminalLocation": [
    {
      "address": "acr:10.0.0.1",
      "currentLocation": {
        "accuracy": "100",
        "altitude": "1001.0",
        "latitude": "-80.86302",
        "longitude": "41.277306",
        "timestamp": {
          "seconds": 1483231138,
          "nanoSeconds": 0
        }
      }
    },
    {
      "address": "acr:10.0.0.2",
      "currentLocation": {
        "accuracy": "100",
        "altitude": "1001.0",
        "latitude": "-80.86301",
        "longitude": "41.277306",
        "timestamp": {
          "seconds": 1483231138,

```

```

        "nanoSeconds": 0
      }},
      "locationRetrievalStatus": "Retrieved"
    ]
  }
}

```

Response

```

HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT

```

7.3.11 UE Area Subscribe

7.3.11.1 Definition

The UE Area Subscribe (clause 5.3.11) shall follow the "Area (circle) location notification" procedure defined in [6], clause 5.3.5. In order to subscribe, the client shall request the creation of the resource "Area (circle) notification subscriptions" defined in [6], clause 6.5, applying the method "POST" as in [6], clause 6.5.5. The request URI shall be:

{apiRoot}/location/v2/subscriptions/area/circle

where {apiRoot} is defined in clause 7.2.

The request shall contain an instance of the CircleNotificationSubscription data type, which includes the address of the user(s) to monitor as well as the location (coordinates of the centre) and size (radius) of the circle. The response shall contain an instance of the CircleNotificationSubscription data type and the location of the created resource. Successfully created subscriptions may be updated by applying the method "PUT" defined in [6], clause 5.3.5.

The notification messages sent to a client subscribed to UE Area Subscribe shall follow the notification procedure defined in [6], clause 6.5, by applying the method "POST" therein, see [6], clause 6.5.5. The body of the notification message shall contain an instance of the SubscriptionNotification data type.

7.3.11.2 Example: Create a UE Area subscription (informative)

In this example, the client issues a request to create a UE Area subscription for a certain user identified by its IP address. The client expects to receive a notification when the requested UE enters the specified circle.

Request

```

POST /exampleAPI/location/v2/subscriptions/area/circle HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: example.com

```

```

{"circleNotificationSubscription": {
  "address": "acr:10.0.0.1",
  "callbackReference": {
    "callbackData": "1234",
    "notifyURL": "http://application.example.com/notifications/distance_notifications/123456"
  },
  "checkImmediate": "true",
  "clientCorrelator": "0123",
  "enteringLeavingCriteria": "Entering",
  "frequency": "10",
  "latitude": "-80.86302",
  "longitude": "41.277306",
  "radius": "500",
  "trackingAccuracy": "10"
}}

```

Response

```

HTTP/1.1 201 Created
Content-Type: application/json
Location: http://example.com/exampleAPI/location/v2/subscriptions/distance/subscription123
Content-Length: nnnn
Date: Sun, 01 Jan 2017 00:38:59 GMT

```

```

{"circleNotificationSubscription": {
  "address": "acr:10.0.0.1",
  "callbackReference": {
    "callbackData": "1234",
    "notifyURL": "http://application.example.com/notifications/distance_notifications/123456"
  },
  "checkImmediate": "true",
  "clientCorrelator": "0123",
  "enteringLeavingCriteria": "Entering",
  "frequency": "10",
  "latitude": "-80.86302",
  "longitude": "41.277306",
  "radius": "500",
  "trackingAccuracy": "10",
  "resourceURL":
    "http://example.com/exampleAPI/location/v2/subscriptions/area/circle/subscription123"
}}

```

7.3.11.3 Example: Client notification about UE Area change (informative)

In this example, the client receives a notification of the location of the requested UE, identified by its IP address.

Request

```

POST /location_notifications/123456 HTTP/1.1
Content-Type: application/json
Content-Length: nnnn
Accept: application/json
Host: clientApp.example.com

```

```

{"subscriptionNotification": {
  "enteringLeavingCriteria": "Entering",
  "isFinalNotification": "false",
  "link": {
    "href":
      "http://example.com/exampleAPI/location/v2/subscriptions/area/circle/subscription123",
    "rel": "CircleNotificationSubscription"
  },
},
"terminalLocation": {
  "address": "acr:10.0.0.1",
  "currentLocation": {
    "accuracy": "100",
    "altitude": "1001.0",
    "latitude": "-80.86302",
    "longitude": "41.277306",
    "timestamp": {
      "seconds": 1483231138,
      "nanoSeconds": 0
    }
  },
  "locationRetrievalStatus": "Retrieved"
}
}

```

Response

```

HTTP/1.1 204 No Content
Date: Sun, 01 Jan 2017 00:38:59 GMT

```

Annex A (informative): Complementary material for API utilisation

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 Location API a supplementary description file compliant to the OpenAPI Specification [i.3].

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 description file, relating to the present document, is located at https://forge.etsi.org/rep/gitweb.cgi/MEC.GS_013.git.

History

Document history		
V1.1.1	July 2017	Publication
V2.1.1	September 2019	Publication
V2.2.1	January 2022	Publication