ENVELOPE

Evaluation and validation of connected mobility in real open systems beyond 5GS

Devices in Area APIs

HORIZON JU Innovation Actions | 101139048 |
ENVELOPE - HORIZON-JU-SNS-2023

Co-funded by the European Union

6GSNS

Table of contents

1 Introduction

The present document focuses on a Devices in Area Service that facilitates the provision of information about devices located within a given geographical area. This document specifies the necessary API with the data model and data format.

The specification follows the same approach of CAMARA Device Location APIs sharing the main data model structures.

The objective of the Devices in Area APIs is to provide location information about the UEs to CAM applications running on the network side.

2 API Overview

API name	Description	Modules Involved	Use Cases
Name of the API	Short summary of the API	List of 5GS or other modules interacting with this API	List of Use Cases using the API
Devices in Area	Information (i.e., event-based notifications or on-demand query) about the devices in a certain geographical area	Edge Location Service	It-UC1, It-UC2

3 Sequence Diagrams

3.1.1 Devices in Area Query

The Devices in Area Query is the procedure for applications acquiring the devices that are in a geographical area.

The Devices in Area Query procedure is illustrated in Figure 1.

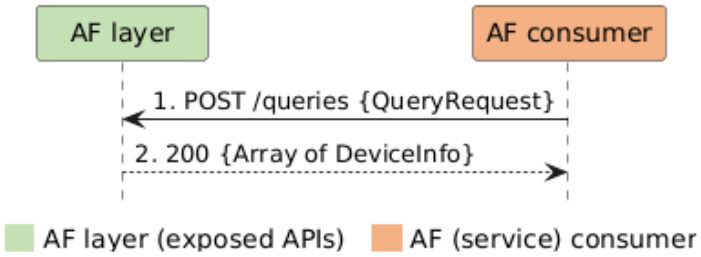

Figure 1 – Flow of Devices in Area Query

1. The AF consumer looks up information about devices in a geographical area by sending a request to the resource representing the devices in area information. The request must include the geographical area of interest.

2. The AF layer Service returns a response with a message body including the information of the devices in the area according to the request.

3.1.2 Devices in Area Subscription

The Devices in Area Subscription is the procedure for applications to request to receive notifications about devices that enter or exit a geographical area. Such notifications help the application to be updated about which devices are in a given geographical area. In this procedure, the Devices in Area Service will continue to report the subscribed information until the subscription is cancelled, or an optional specified time limit is reached.

The Devices in Area Subscription procedure is illustrated in Figure 2.

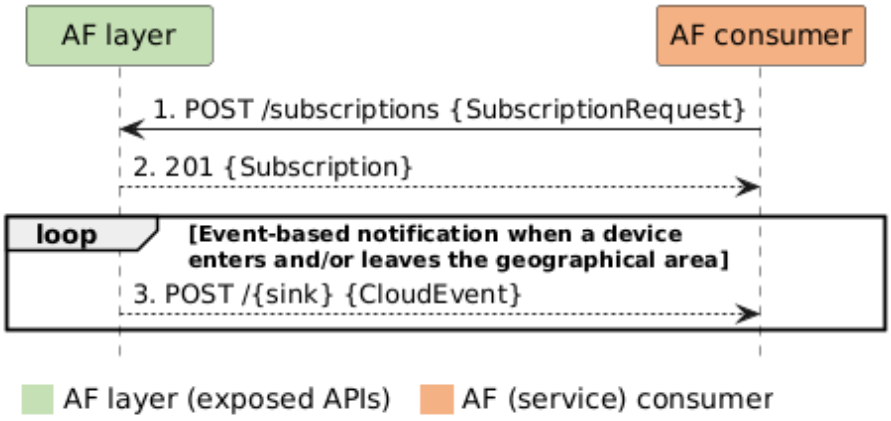

Figure 2 – Flow of Devices in Area Subscription

1. The AF consumer application subscribes to Devices in Area notifications by requesting the creation of a resource containing the subscription details, which include the geographical area, and a callbackURL for receiving the device information.
2. The Devices in Area Service returns a response with a resource URI containing the subscriptionId.
3. The Devices in Area Service reports the up-to-data subscribed information to the AF consumer application by sending a message with the message body containing the CloudEvent notification to the callbackURL, which includes the event about a device (i.e., information about a device entering or leaving an area).

4 API definition

4.1 Resource structure

4.1.1 Resource URIs

The root of the resource URIs of this API shall be:

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

where *apiName* shall be set to *devices-in-area*, and *apiVersion* shall be set to *v1* for the current version of the present document.

The resource URI structure of this API is introduced in Figure 3.

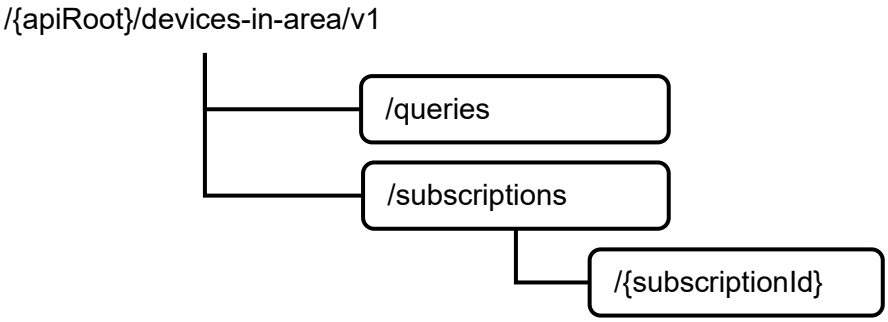

Figure 3 - Resource URI structure for the devices-in-area API

Table 1 introduces an overview of the resources and the methods.

Table 1. Resources and methods overview.

Resource name	Resource URI	HTTP method	Description
Devices in Area information	/queries	POST	Retrieve information about the devices in the geographical area.
All devices in area subscriptions for a subscriber	/subscriptions	GET	Retrieve a list of active devices in area subscriptions for this subscriber.
		POST	Create a new devices in area discovery subscription.
Existing subscription for devices in area	/subscriptions/{subscriptionId}	GET	Retrieve information on current specific devices in area subscription.
		DELETE	Cancel the existing devices in area subscription.

4.1.2 Request Headers

All API requests must include the *X-Camara-User-Id* header. Each third party should generate its own unique identifier using **UUIDv4** format and include it in every request.

This header is used for resource ownership management, allowing the API to filter and return only the queries and subscriptions that belong to the specified user.

Example:

X-Camara-User-Id: *550e8400-e29b-41d4-a716-446655440000*

4.2 Data model

4.2.1 QueryRequest data type

Table 2. QueryRequest data type

Field	Data Type	Cardinality	Description
area	AreaInfo	1	Definition of the geographical area of interest
fromTime	date-time	0..1	Start of a time window. If specified, the API returns devices known to be in the area at or after this timestamp.
toTime	integer	0..1	End of a time window. If specified, the API returns devices known to be in the area up to this timestamp. (Must be >= fromTime if both set.)
maxAge	integer	0..1	*(Optional)* Maximum age (in seconds) for device location data. If present, only devices with location data not older than this threshold are returned.

4.2.2 SubscriptionRequest data type

Table 3. SubscriptionRequest data type

Field	Data Type	Cardinality	Description
protocol	string	1	Only “HTTP” is allowed in the current implementation
sink	string	1	The HTTP address to which events shall be delivered to.
sinkCredential	structure (inlined)	0..1	Sink credential
→ credentialType	string	1	The type of the credential: only “ACCESSTOKEN” is supported
config	structure (inlined)	1	Implementation-specific configuration parameters needed by the subscription manager for acquiring events
→ subscriptionDetail	structure (inlined)	1	The detail of the requested event subscription
→→ area	areaInfo	1	Definition of the geofencing geographical area for which events are generated
→ subscriptionExpireTime	date-time	0..1	The subscription expiration time requested by the API consumer.

→ subscriptionMaxEvents	integer	0..1	Identifies the maximum number of event reports to be generated (>=1) requested by the API consumer - Once this number is reached, the subscription ends.
→ initialEvent	boolean	0..1	Set to true by API consumer if consumer wants to get an event as soon as the subscription is created and current situation reflects event request. Example: Consumer request area entered event. If consumer sets initialEvent to true and device is already in the geofence, an event is triggered

4.2.3 Subscription data type

Table 4. Subscription data type

Field	Data Type	Cardinality	Description
protocol	string	1	Only “HTTP” is allowed in the current implementation
sink	string	1	The HTTP address to which events shall be delivered to.
sinkCredential	Structure (inlined)	0..1	Sink credential
→ credentialType	string	1	The type of the credential: only “ACCESSTOKEN” is supported
config	Structure (inlined)	1	Implementation-specific configuration parameters needed by the subscription manager for acquiring events
→ subscriptionDetail	Structure (inlined)	1	The detail of the requested event subscription
→→ area	areaInfo	1	Definition of the geofencing geographical area for which events are generated
→ subscriptionExpireTime	date-time	0..1	The subscription expiration time requested by the API consumer.
→ subscriptionMaxEvents	integer	0..1	Identifies the maximum number of event reports to be generated (>=1) requested by the API consumer - Once this number is reached, the subscription ends.
→ initialEvent	boolean	0..1	Set to true by API consumer if consumer wants to get an event as soon as the subscription is created and current situation reflects event request. Example: Consumer request area entered event. If consumer sets initialEvent to true and device is

			already in the geofence, an event is triggered
id	string	1	The unique identifier of the subscription
startsAt	date-time	1	Date and time when the event subscription will begin/began
expiresAt	date-time	0..1	Date and time when the event subscription will expire
status	string	0..1	Status of the subscription: ACTIVATION_REQUESTED: Subscription creation (POST) is triggered but subscription creation process is not finished yet. ACTIVE: Subscription creation process is completed. Subscription is fully operative. DEACTIVE: Subscription is temporarily inactive, but its workflow logic is not deleted. EXPIRED: Subscription is ended (no longer active). This status applies when subscription is ended due to SUBSCRIPTION_EXPIRED or ACCESS_TOKEN_EXPIRED event. DELETED: Subscription is ended as deleted (no longer active). This status applies when subscription information is kept (i.e. subscription workflow is no longer active, but its meta-information is kept).

4.2.4 CloudEvent data type

Table 5. CloudEvent data type.

Field	Data Type	Cardinality	Description
id	string	1	The unique identifier of this event
source	string	1	Identifies the context in which an event happened – defined as a non-empty URI-reference, for example: URI with a DNS authority: https://github.com/cloudevents

			Universally unique URN with a UUID: urn:uuid:6e8bc430-9c3a-11d9-9669-0800200c9a66 Application-specific identifier:/cloudevents/spec/pull/123
type	string	1	Camara Event type corresponding to the event notified Possible values: "org.camaraproject.geofencing-subscriptions.v0.area-entered", or "org.camaraproject.geofencing-subscriptions.v0.area-left"
specversion	string	1	Version of the specification to which this event conforms (must be 1.0 if it conforms to cloudevents 1.0.2 version)
datacontenttype	string	0..1	Media-type that describes the event payload encoding, must be "application/json" for CAMARA APIs
time	date-time	1	Timestamp of when the occurrence happened. Must adhere to RFC 3339.
data	structure (inlined)	1	Event details payload described in each CAMARA API and referenced by its type
→ subscriptionId	string	1	The unique identifier of the subscription
→ device	deviceInfo	1	The developer can choose to provide the below specified device identifiers: * ipv4Address (currently the only supported)
→ area	areaInfo	1	Definition of the geofencing geographical area for which events are generated

4.2.5 DeviceInfo data type

Table 6. deviceInfo data type.

Field	Data Type	Cardinality	Description
ipv4Address	Structure (inlined)	1	In the current implementation, both public and private addresses should be set to the IPv4 address of the UE (device) as assigned by the B5G system
→ publicAddress	string	1	A single IPv4 address with no subnet mask

→ privateAddress	string	1	A single IPv4 address with no subnet mask
→ publicPort	integer	1	TCP or UDP port number

4.2.6 AreaInfo data type

Table 7. areaInfo data type.

Field	Data Type	Cardinality	Description
areaType	string	1	Type of this area. Only “CIRCLE” is supported
radius	integer	1	Expected accuracy (radius) for the subscription event of device location, in meters from `center`. Allowed value range: [1, 200000]
center	Structure (inlined)	1	Coordinates (latitude, longitude) defining the center point of the subscribed geofencing region
→ latitude	Number (double)	1	Latitude component of the center point. Allowed value range: [-90, 90]
→ longitude	Number (double)	1	Longitude component of the center point. Allowed value range: [-180, 180]

4.2.7 ErrorInfo data type

Table 8. ErrorInfo data type.

Field	Data Type	Cardinality	Description
status	integer	1	HTTP status code returned along with this error response
code	string	1	Code given to this error
message	string	1	Detailed error description

4.3 API operations

4.3.1 Resource: devices_queries

The resource URI is: {apiRoot}/devices-in-area/v1/queries

The content of the methods requests and response is JSON, and it is indicated by setting the content type "application/json".

4.3.1.1 Resource methods

4.3.1.1.1 POST

The POST method is used to query the devices in the area.

Parameters: no parameters

Table 9. Data structures supported by the POST request on this resource

Request body	Data type	Cardinality	Notes
	QueryRequest	1	

Table 10. Data structures supported by the GET response on this resource

Response body	Data type	Cardinality	Response Codes	Notes
	Array of DeviceInfo(s)	1	200	Upon success, a response body containing the available devices is returned. One object for each device found in the specified area. The array can have zero elements if no devices match the query.
	ErrorInfo	0..1	400	Problem with the client request
	ErrorInfo	0..1	401	Authentication problem with the client request
	ErrorInfo	0..1	403	Client does not have sufficient permission
	ErrorInfo	0..1	404	Not found
	ErrorInfo	0..1	406	Not acceptable
	ErrorInfo	0..1	429	Too many requests
	ErrorInfo	0..1	500	Internal server error
	ErrorInfo	0..1	503	Service unavailable

4.3.2 Resource: devices_subscriptions

The resource URI is: {apiRoot}/devices-in-area/v1/subscriptions

The content of the methods requests and response is JSON, and it is indicated by setting the content type "application/json".

4.3.2.1 Resource methods

4.3.2.1.1 GET

Retrieve a list of devices in area event subscription(s).

Parameters: no parameters

Table 11. Data structures supported by the GET request on this resource

Request body	Data type	Cardinality	Notes
			No request body

Table 12. Data structures supported by the GET response on this resource

Response body	Data type	Cardinality	Response Codes	Notes
	Array of Subscription(s)	1	200	List of subscriptions retrieved
	ErrorInfo	0..1	400	Problem with the client request
	ErrorInfo	0..1	401	Authentication problem with the client request
	ErrorInfo	0..1	403	Client does not have sufficient permission
	ErrorInfo	0..1	404	Not found
	ErrorInfo	0..1	406	Not acceptable
	ErrorInfo	0..1	429	Too many requests
	ErrorInfo	0..1	500	Internal server error
	ErrorInfo	0..1	503	Service unavailable

4.3.2.1.2 POST

Create a subscription to receive notifications about devices that enter or exit a specified geographical area.

Parameters: no parameters

Table 13. Data structures supported by the POST request on this resource

Request body	Data type	Cardinality	Notes
	SubscriptionRequest	1	

Table 14. Data structures supported by the POST response on this resource

	Data type	Cardinality	Response Codes	Notes
Response body	Subscription	1	201	Subscription created
	ErrorInfo	0..1	400	Problem with the client request
	ErrorInfo	0..1	401	Authentication problem with the client request
	ErrorInfo	0..1	403	Client does not have sufficient permission
	ErrorInfo	0..1	409	Conflict
	ErrorInfo	0..1	415	Unsupported Media Type
	ErrorInfo	0..1	422	Unprocessable Entity
	ErrorInfo	0..1	429	Too many requests
	ErrorInfo	0..1	500	Internal server error
	ErrorInfo	0..1	503	Service unavailable

4.3.2.1.2.1 Callback

Important: this endpoint is to be implemented by the API consumer. The Device Location (Geofencing) server will call this endpoint (given during subscription creation by the 'sink field) whenever a new subscription event is available.

Parameters: no parameters

Table 15. Data structures supported by the callback of this resource

	Data type	Cardinality	Notes
Request body	CloudEvent	1	

4.3.3 Resource: individual devices_subscriptions

The resource URI is: {apiRoot}/devices-in-area/v1/subscriptions/{subscriptionId}

4.3.3.1 Resource methods

4.3.3.1.1 GET

Retrieve a Devices in Area subscription.

Parameters: no parameters

Table 16. Data structures supported by the GET response on this resource

	Data type	Cardinality	Response Codes	Notes
Response body	Subscription	1	200	Successful Response
	ErrorInfo	0..1	400	Problem with the client request
	ErrorInfo	0..1	401	Authentication problem with the client request
	ErrorInfo	0..1	403	Client does not have sufficient permission
	ErrorInfo	0..1	404	Resource Not Found
	ErrorInfo	0..1	406	Not acceptable
	ErrorInfo	0..1	422	Validation Error
	ErrorInfo	0..1	500	Server error
	ErrorInfo	0..1	503	Service unavailable. Typically the server is down.

4.3.3.1.2 DELETE

Delete a Devices in Area subscription.

Parameters: no parameters

Table 17. Data structures supported by the DELETE response on this resource

	Data type	Cardinality	Response Codes	Notes
Response body		1	204	Subscription deleted, no response body
	ErrorInfo	0..1	400	Problem with the client request
	ErrorInfo	0..1	401	Authentication problem with the client request
	ErrorInfo	0..1	403	Client does not have sufficient permission
	ErrorInfo	0..1	404	Resource Not Found
	ErrorInfo	0..1	406	Not acceptable
	ErrorInfo	0..1	422	Validation Error
	ErrorInfo	0..1	500	Server error
	ErrorInfo	0..1	503	Service unavailable. Typically the server is down.