



## MSE API Object Descriptions

---

This chapter contains the following sections:

- [maps](#), page 4-2
- [mapscount](#), page 4-2
- [mapsinfo](#), page 4-3
- [mapsimage](#), page 4-3
- [mapsimagesource](#), page 4-4
- [locationclients](#), page 4-4
- [locationclientscount](#), page 4-5
- [locationtags](#), page 4-5
- [locationtagscount](#), page 4-6
- [locationrogueaps](#), page 4-6
- [locationrogueapscount](#), page 4-7
- [locationrogueclients](#), page 4-7
- [locationrogueclientscount](#), page 4-8
- [locationinterferers](#), page 4-8
- [locationinterfererscount](#), page 4-9
- [locationhistoryclients](#), page 4-9
- [locationhistoryclientscount](#), page 4-10
- [locationhistorytags](#), page 4-10
- [locationhistorytagscount](#), page 4-11
- [locationhistoryrogueaps](#), page 4-11
- [locationhistoryrogueapscount](#), page 4-12
- [locationhistoryrogueclients](#), page 4-12
- [locationhistoryrogueclientscount](#), page 4-13
- [locationhistoryinterferers](#), page 4-13
- [locationhistoryinterfererscount](#), page 4-14
- [notifications](#), page 4-14

## maps

The maps object returns detailed map information about campuses, buildings, floors, access points, map dimensions, regions, zones, GPS marker, image information, etc.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

None

**Response Types**

JSON, XML

**Return Codes**

- 200—Successful response
- 404—MSE does not have any Map information
- 500—Server Error

## mapscount

The mapscount returns maps count specifying the number of Campuses, Buildings, and Floors known to MSE.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

None

**Response Types**

JSON, XML

**Return Codes**

- 200—Successful response
- 500—Server Error

# mapsinfo

The mapsinfo object returns all the floor information associated with the campusName -> buildingName -> floorName. This includes floor dimension, Access Points and their information, GPS Markers etc.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- floorname—Name of the required floor
- buildingname—Name of the required building
- campusname—Name of the required campus

**Response Types**

JSON, XML

**Return Codes**

- 200—Successful response
- 404—No floor with the specified campusName -> buildingName -> floorName
- 500—Server Error

# mapsimage

The mapsimage object returns the floor image data associated with the particular campusName -> buildingName -> floorName.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- floorname—Name of the required floor
- buildingname—Name of the required building
- campusname—Name of the required campus

**Response Types**

image/jpeg

**Return Codes**

- 200—Successful response
- 404—No image with the specified campusName -> buildingName -> floorName
- 500—Server Error

## mapsimagesource

The mapsimagesource object returns the image associated with the specified image name.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- imageName—Name of the required image.

**Response Types**

image/jpeg

**Return Codes**

- 200—Successful response
- 404—No image with the specified imagename
- 500—Server Error

## locationclients

The location clients object returns the current location of the wireless client for the specified device ID. The ID can be MAC address, IP address, or Username.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- id—Mac address, IP Address, or Username of the wireless client.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there is no wireless client for the specified ID.No image with the specified imagename
- 500—Server Error

## locationclientscount

The locationclientscount object returns count or location of wireless clients on the MSE. Results are filtered based on the specified query param conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the wireless client.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 500—Server Error

## locationtags

Returns a list of Location of Tags for the specified query conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the tag.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there are no tags for the specified query conditions.
- 500—Server Error

## locationtagscount

The location tags count object returns a count of Tags on MSE based on the specified Query Param conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the tag.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 500—Server Error

## locationrogueaps

Returns a list of Location of Rogue APs for the specified query conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the tag.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there are no Rogue APs for the specified query conditions.
- 500—Server Error

## locationrogueapscount

Returns a count of Rogue APs on MSE based on the specified Query Param conditions.

Returns a list of Location of Rogue APs for the specified query conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the rogue AP.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 500—Server Error

## locationrogueclients

Returns the Location of Rogue Client for the specified id.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- id—MAC address of the rogue client.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there is no rogue client for the specified id.
- 500—Server Error

## locationrogueclientscount

Returns a count of Rogue Clients on MSE based on the specified Query Param conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the rogue client.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 500—Server Error

## locationinterferers

Returns the Location of Interferer for the specified id.

Returns a count of Rogue Clients on MSE based on the specified Query Param conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- id—MAC address of the interferer.

**Response Types**

XML and Json



**Return Codes**

- 200—Successful response
- 404—If there is no interferer for the specified ID.
- 500—Server Error

## locationinterfererscount

Returns a count of Interferers on MSE based on the specified Query Param conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the interferers.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 500—Server Error

## locationhistoryclients

Returns a list of historical Location records of Wireless Client for the specified id and query conditions

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- id—MAC address, IP address, or username of the wireless client.
- []—Defines query conditions for the wireless client

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there are no historical records of wireless client for the specified ID and query conditions
- 500—Server Error

## locationhistoryclientscount

Returns a count of historical Location records of Wireless Clients on MSE based on the specified Query Param conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the wireless client.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 500—Server Error

## locationhistorytags

Returns a list of the historical Location records of Tag for the specified id and query conditions

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- id—MAC address of the tag
- []—Defines query conditions for the tag.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there are no historical records of tag for the specified ID and query conditions.
- 500—Server Error

## locationhistorytagscount

Returns a count of historical location records of Tags on MSE based on the specified Query Param conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the interferers.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 500—Server Error

## locationhistoryrogueaps

Returns a list of historical records of Location of Rogue APs for the specified query conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the rogue AP.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there are no historical records of Rogue APs for the specified query conditions.
- 500—Server Error

## locationhistoryrogueapscount

Returns a count of historical Location records of Rogue APs based on the specified Query Param conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the rogue AP.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 500—Server Error

## locationhistoryrogueclients

Returns a list of historical Location records of Rogue Clients for the specified query conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the rogue client.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there are no historical records of rogue clients for the specified query conditions.
- 500—Server Error

## locationhistoryrogueclientscount

Returns a count of Historical Location records of Rogue Clients on MSE based on the specified Query Param conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the rogue client.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 500—Server Error

## locationhistoryinterferers

Returns a list of historical Location records of Interferers for the specified query conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the interferers.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there are no Historical Location records of Interferer for the specified query conditions.
- 500—Server Error

## locationhistoryinterfererscount

Returns a count of historical Location records of Interferers for the specified id and query conditions.

**HTTP Method**

GET

**Accepts**

- application/json
- application/xml

**Parameters**

- []—Defines query conditions for the interferers.

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404— If there are no historical records of Interferer for the specified id and query conditions
- 500—Server Error

## notifications

**HTTP Method**

GET—Returns all the Notifications Events subscribed by the requesting user.

**Accepts**

- application/json
- application/xml

**Response Types**

XML and Json

**Return Codes**

- 200—Successful response
- 404—If there are no Notification Events subscribed by the requesting user.
- 500—Server Error

**HTTP Method**

PUT—Creates the Notification Event associated with the specified name and subscribed by the requesting user.

**Accepts**

- application/json
- application/xml

**Response Types**

XML and Json

**Return Codes**

- 200—Notification Event was successfully created.
- 404—If there is no Notification Event associated with the specified name or the requesting user did not create the notification event.
- 500—Server Error

