BlueUp S.r.l.
Loc. Belvedere, Ingresso 2, 99
53034 Colle di Val d'Elsa (SI)
Ph. +39 344 2030929 / +39 0577 043101
E-mail: info@blueupbeacons.com
Web: www.blueupbeacons.com
P.IVA/CF/VAT N. IT01373980521

Technical Documentation

MeshCube

API HTTP User Guide

Colle Val d'Elsa, June 28th, 2023

Summary

List of tables

References

[1] MeshCube - BPE - User Manual, June 2023

[2] MeshCube - DemoKit - Smart Industry, January 2024

[3] MeshCube - DemoKit - Smart Building, January 2024

[4] MeshCube_AnchorsInstallation, September 2022

[5] BPE Installer - User Guide, March 2023

[6] BPE- MQTT Broker Installation Guide, March 2023

[7] MeshCube - BPE APIs - MQTT - User Manual, July 2022

1. HTTP

HTTP is a request-response communication protocol.

In this type of communication the standard request is http://<URL>[:port]/<request_type>.

Unlike the MQTT protocol ([7]), the HTTP response is received only after a request is sent by the client.

There are two request methods, that are Get and Post. The Post request supports also parameters that can be stated in the Body of the request to obtain specific response. The Post method is used only if the request supports the parameters statement.

The responses are in JSON format and the commands are case sensitive.

2. Requests

2.1. http://<URL>[:port]/bpe

The response contains basic information about the system.

Method: Get / Post

Response:

```
{
    "is_bpe_instance": true,
    "version": "4.6.11.31141",
    "system": {
        "code": 1,
        "description": "<system_description>"
    },
    "license": {
        "type": "timerelated",
        "remote_check": true,
        "last_check": 1618407849,
        "status": "active",
        "valid_from": 1587115813
    },
    "started_on": 1618224215,
    "uptime": {
        "days": 2,
        "hours": 3,
        "minutes": 2,
        "seconds": 43
    },
    "pressure": {
        "ram": 316,
        "pram": 4,
        "pcpu": 80.7
    }
}
```

Tab. 1: bpe

Name	Type	Description
is_bpe_instance	bool	Whether the system is a BPE instance
version	string	Version of the system
system	object	System numeric code and description
code	number	System type code
description	string	System type Description
license	object	License information
type	string	Type of the License
remote_check	bool	Whether the remote check is enabled
last_check	number	Date of the last check of the license, UNIX timestamp
status	string	License Status

valid_from	number	Date from when the license is valid, UNIX timestamp
started_on	number	UNIX timestamp of the date when the system started
uptime	object	System uptime specified in days, hours, minutes and seconds
pressure	object	RAM and CPU resources currently used by the system
ram	number	Used RAM in mb
pram	number	Used RAM in %
pcpu	number	Used Processor in %

2.2. http://<URL>[:port]/bpe/getSystemInfo

The response contains all the information about the project.

Method: Get / Post

Response:

```
"data": {
  "description": "<project_description>",
  "maps": [
    {
      "id": 1,
      "description": "<map-description>",
      "pixel_per_mt": 47,
      "z": 0,
      "gps_coordinates": false,
      "image": {
        "type": "image/png",
        "size": 270220,
        "data": "<image-data>",
        "width": 883,
        "height": 649
      },
      "offset": { "x": 0, "y": 0, "z": 0  },
      "area": {
        "vertexes": [ ~ ]
      },
      "zones": {
        "zone1": {
          "id": "zone1",
          "description": "zone1_description",
          "color": "#f0008a",
          "z": 0,
```

```
            "vertexes": [ ~ ]
          },
          "zone2": { ~ }
        },
        "locators": {},
        "mesh_nodes": {
          "Node1": {
            "from_cache": false,
            "id": "Node1",
            "address": <node_address>,
            "network_id": <network_id>,
            "version": "3.0",
            "node_class": "F9",
            "operation_mode": "opportunistic_anchor",
            "alarm": "none",
            "state": "default",
            "role": "router_node",
            "low_latency": false,
            "autorole": false,
            "max_buffer_usage": 0,
            "avg_buffer_usage": 0,
            "routers_in_neighborhood": 2,
            "scans": 13,
            "boot_count": 0,
            "battery": 3036,
            "routing": {
              "advertised_cost": 0.01,
              "sink_address": 2,
              "next_hop": {
                "address": 2,
                "quality": 100,
                "rssi": -51,
                "power": -12
              }
            },
            "configuration": {
              "sensing_interval": 300,
              "alerts_enabled": true,
              "beaconing": "disabled"
            },
```

```
            "map": {
              "id": 1,
              "description": "<map_description>"
            },
            "position": {"x": 9.93, "y": 2.19, "z": 0 },
            "zone": {
              "id": "zone1",
              "description": "<zone1_description>",
              "color": "#f0008a",
              "z": 0,
              "vertexes": [ ~ ]
            },
            "anchor": {
              "id": "ANCHOR1",
              "positioning_enabled": true,
              "calibrated_rssi": -50,
              "rssi_offset": 0,
              "n": 1
            },
            "sensor": {
              "id": "SENSOR1",
              "temperature": {
                "unit": "celsius",
                "value": 20.2
              },
              "humidity": 44,
              "pressure": 100189,
              "co2": {
                "ppm": 961,
                "classification": "good"
              }
            },
            "last_update": 1635426135,
            "last_update_elapsed": { },
            "status": "online"
          },
          "Node2": { ~ }
        }
      }
    ]
```

```
    }
  }
}
```

Tab. 2: getSystemInfo

Name	Type	Description
id	string	Response ID, same as specified in the request
response	object	Object containing all the requested information
status	bool	Status of the request
description	string	Project Description
maps	array	Array containing information about all the maps of the project
id	number	ID Map number
description	string	Map name
pixel_per_mq	number	Number of pixel per meter
z	number	Z coordinate of the map
gps_coordinates	bool	Whether the GPS coordinates of the map are enabled
image	object	Map Image information
type	string	Image mime type
size	number	Image size in bytes
data	string	Image data URL
width	number	Image width in pixel
height	number	Image height in pixel
offset	object	Offset of the origin of the Map
area	object	Map Area information, containing the coordinates of the covered area
zones	object	Map Zones information
id	string	Zone Id
description	string	Zone description
color	string	Zone Color
z	number	Used in case of overlying zones. The positioning considers the zone with highest z value
vertexes	array	Coordinates of the vertexes of the zone
locators	object	Map BLE Locators information
mesh_nodes	object	Map Mesh Nodes information

Tab. 3: getSystemInfo - "locators" object

Name	Type	Description
id	string	Locator Id
map	object	Information about the Map where the Locator is positioned. It contains information about the Map Id number and the Map Description
position	object	Coordinates of the position of the Locator on the Map
positioning_enabled	bool	Whether the Locator contributes to the Tags Positioning
max_rssi_attenuation	number	Max RSSI Attenuation
n	number	Environmental Factor
proximity_enabled	bool	Whether the Locator contributes to the Proximity
proximity_immediate_threshold	number	Threshold value to consider the distance Immediate in meters
proximity_near_threshold	number	Threshold value to consider the distance Near in meters

Tab. 4: getSystemInfo - "mesh_nodes" object

Name	Type	Description
from_cache	bool	Whether the information are recovered from the previous save
id	string	Mesh Node Id
address	number	Mesh Node Address
network_id	number	Mesh Node Network Id
version	string	FW version of the Mesh Node
node_class	string	Node Class
operation_mode	string	Node Operation Mode
alarm	string	Type of the alarm activated on the device
state	string	Current Node state
role	string	Mesh Node Role
low_latency	bool	Whether the Node is Low Latency
autorole	bool	Whether the Node is in Autorole Mode
max_buffer_usage	number	Maximum amount of used buffers
avg_buffer_usage	number	Average amount of used buffers
mem_alloc_fails	number	Number of Memory Allocations Fails
routers_in_neighborhood	number	Number of Routers in Neighborhood
scans	number	Number of Scans performed by the Node
boot_count	number	Number of reboots
battery	number	Current battery level of the device
routing	object	Provides information about the routing of a Node in the Mesh network
advertised_cost	number	Price of the route
sink_address	number	Sink node to which the Locator refers
next_hop	object	Information about the next hop of the routing
address	number	Next Hop Node address
quality	number	Quality of the next hop
rssi	number	Average RSSI of the next hop
power	number	Radio power used to hop
configuration	object	Configuration of the parameters of the Node
sensing_interval	number	Sensing Interval of the device

alerts_enabled	bool	Whether the alerts are enabled
beaconing	string	Whether the beaconing is enabled
map	object	Information about the Map where the Mesh Locator is positioned. It contains information about the Map Id number and the Map Description
position	object	Coordinates of the position of the Locator on the Map
zone	object	Information about the zone where the Node is placed
anchor	object	This object is present if the Node is also an Anchor
id	string	Anchor Id
positioning_enabled	bool	Whether the Locator contributes to the Tags Positioning
calibrated_rssi	number	Value of the Calibrated RSSI at 1 meter
rssi_offset	number	Offset set to adjust the received RSSI if it's higher or lower than the calibrated one
n	number	Environmental Factor
sensor		This object is present if the Node is also a Sensor
id	string	Sensor Id
temperature	object	Object containing the information about the current sensed temperature with the value of the temperature and the measuring unit
humidity	number	Humidity value
pressure	number	Pressure value
co2	object	Object containing the information about the current sensed CO2 with the value of the CO2 in ppm and the quality of the air based on the CO2 value
last_update	number	UNIX Timestamp of the last update of the Node
last_update_elapsed	object	Time elapsed from the last update
status	string	Current status of the Node

2.3. http://<URL>[:port]/bpe/getTagInfo

Response contains the list of all detected tags.

Method: Get / Post

```
{
  "id": "<identifier>",
  "response": {
    "status": true,
    "data": {
      "count": 4,
      "tags": [
        {
          "id": "TAG1",
          "node_id": "<node_id>",
          "node_address": <node_address>,
          "node_network_id": <network_id>,
          "node_role": "non_router_node",
          "type": "mesh",
          "from_cache": false,
          "version": "3.0",
          "node_class": "FA",
          "operation_mode": "autoscan_tag",
          "alarm": "none",
          "state": "sleep",
          "last_update_elapsed": { },
          "battery": 3078,
          "anchors": 3,
          "anchor": [
            {
              "id": "ANCHOR1",
              "address": <anchor-address>,
              "last_update": 1635414607,
              "rssi": -54,
              "detection_radius": 2.51,
              "position": {"x": 14.61, "y": 4.61, "z": 0}
            }
          ],
          "localization_mode": "trilateration",
          "location": {
```

```
            "map": 1,
            "map_description": "<map_description>",
            "position": { "x": 12.33, "y": 4.01, "z": 0},
            "zone": {
              "id": "zone1",
              "description": "<zone1_description>"
            },
            "accuracy": 3.92,
            "anchors": [
              {
                "id": "ANCHOR1",
                "address": <anchor_address>,
                "last_update": 1635414607,
                "rssi": -54,
                "detection_radius": 2.51,
                "position": { "x": 14.61, "y": 4.61, "z": 0}
              ]
            },
            "status": "online"
          }
        ]
      }
    }
}
```


Tab. 5: getTagInfo - mesh tag

Name	Type	Description
id	string	Tag ID
node_id	string	Node Id
node_address	number	Node Address
node_network_id	number	Network ID
node_role	string	Specifies the role of the node
type	string	Tag Type. Possible values are: • Pager • Tracer • Beacon • Mesh Tag
from_cache	bool	Whether the information are recovered from the previous save
version	string	Firmware Version

node_class	string	Node Class, where FA means a Tag
operation_mode	string	Mesh Tag Operating Mode. Possible values are: • NRLS • Autoscan
alarm	string	What type of alarm was sent
state	string	Node State. Possible values are: • Default • Alarm • Motion • Sleep
last_update_elapsed	object	Time elapsed from the last detection, in days, hours, minutes and seconds
battery	number	Battery voltage of the Tag in mV
anchors	number	Number of Anchors that detect the Tag
anchors	number	Number of Anchors that detect the Tag
anchor	array	List of the Anchors that detect the Tag
id	string	Anchor ID
address		Anchor address
last_update	number	UNIX timestamp of the last detection date
rssi	number	Instant RSSI value of the signal exchanged between the Tag and the Locator
detection_radius	number	Location Distance Factor in meters
position	object	Coordinates of the position of the Tag on the Map
localization_mode	string	Type of localization of the tag
location	object	Object containing the information about the map, the position and the zone where the Tag is
accuracy	number	Accuracy in meters of the Tag position
anchors		List of anchors that contribute to the Tag Localization
id	string	Anchor ID, i.e. Locators Hostname
address	number	Node Address
last_update	number	UNIX timestamp of the last detection date
rssi	number	Instant RSSI value of the signal exchanged between the Tag and the Locator
detection_radius	number	Location Distance Factor in meters
position	object	Coordinates of the position of the Tag on the Map
status	string	Status of the Tag

2.3.1 GetTagInfo with Parameters

This API is used to filter the list of tag that are retrieved as a response.

Each filter is optional.

Method**:** Post, with parameters specified in body

Request:

```
{
    "id": "<tag_id>"
    "network_id": <network_id>,
    "address": <network_address>,
    "from_cache": false,
    "located": true,
    "map": <map_number>,

    "include_anchors": false,
    "include_position": true,
    "include_positioning_anchors": true
}
```

Tab. 6: GetTagInfo parameters

Name	Type	Optional	Default	Description
id	string	Yes	*N/A*	Tag MAC Address
network_id	number	Yes	*N/A*	Network Id
address	number	Yes	*N/A*	Network Address
from_cache	bool	Yes	*N/A*	Whether the information are recovered from the previous save
located	bool	Yes	*N/A*	Whether the tag is located
map	number	Yes	*N/A*	Numeric Map Id
include_anchors	bool	Yes	*true*	Whether to include the anchors that detect the tag
include_position	bool	Yes	*true*	Whether to include the information about the position of the tag. Note: if include_position is false, positioning_anchors will not be listed too
include_positioning_anchors	bool	Yes	*true*	Whether to include the list of positioning anchors (only if include_position is true)

Response:

The response structure is shown below. The Data Object is the same as described in 2.3.

```
{
  "status": true,
  "data": { ~ },
  "error": "<error_message>"
}
```

Tab. 7: getTagInfo Response

Name	Type	Description
status	bool	Specifies if the request was responded successfully
data	object	Data object containing all the information
error	string	Empty if there are no errors, otherwise the error type is specified

2.4. http://<URL>[:port]/bpe/getSensorData

This command allows to request a Sensors' Data.

Method**:** Get, Post

Response:

```
{
    "status": true,
    "data": {
        "count": 2,
        "items": [
            {
                "id": "SENSOR_R9000007-123456",
                "node_id": "R9000007-123456",
                "node_address": 9000007,
                "node_network_id": 123456,
                "node_role": "router_node",
                "from_cache": false,
                "temperature": {
                    "unit": "celsius",
                    "value": 25.8
                },
                "humidity": 46,
                "pressure": 97863,
                "co2": {
```

```
                    "ppm": 1222,
                    "classification": "medium"
                },
                "last_update": 1636034460,
                "last_update_elapsed": {
                    "min": 21,
                    "sec": 8
                },
                "status": "offline"
            },
            {
                "id": "SENSOR_R9000010-123456",
                "node_id": "R9000010-123456",
                "node_address": 9000010,
                "node_network_id": 123456,
                "node_role": "router_node",
                "from_cache": false,
                "temperature": {
                    "unit": "celsius",
                    "value": 24.2
                },
                "humidity": 47,
                "pressure": 98296,
                "co2": {
                    "ppm": 1057,
                    "classification": "medium"
                },
                "last_update": 1636035665,
                "last_update_elapsed": {
                    "min": 1,
                    "sec": 3
                },
                "status": "online"
            }
        ]
    }
}
```

Tab. 8: getSensorData

Name	Type	Description
status	bool	Response status true|false
data	object	Contains the response parameters
count	number	Number of devices
id	string	Sensor ID
node_id	string	Node ID
node_address	string	Node Address
node_network_id	string	Node Network ID
node_role	string	Node Role
from_cache	bool	Whether the information is recovered from the previous save
temperature	object	Object containing temperature data
unit	string	Temperature measure Unit
value	number	Temperature measured value
humidity	number	Humidity measured value
pressure	number	Pressure measured value
co2	object	Object containing CO2 data
ppm	number	Pressure measured value in ppm
classification	string	Classification of the quality of the air, based on ppm value
last_update	number	UNIX timestamp of the last update
last_update_elapsed	object	Time elapsed from the last update
status	string	Node status

2.5. http://<URL>[:port]/bpe/getMeshConfiguration

Retrieves the configuration of the parameters of the Mesh devices.

Method: Get, Post with parameters specified in body

Request:

```
{
       "format": “json” | “hex” | “base64”, [optional]
       "network_id": <number> [optional]
}
```

Response:

```
{
    "status": true,
    "data": {
        "diagnostics_interval": 30,
        "configuration": [
            {
                "node_class": "F9",
                "packet_type": "A2",
                "parameters": {
                    "sensing_interval": 300,
                    "alerts_enabled": true,
                    "beaconing": "disabled"
                }
            },
            {
                "node_class": "FA",
                "packet_type": "B2",
                "parameters": {
                    "default_positioning_interval": 60,
                    "default_timeout": 300,
                    "default_autoscan_enabled": false,
                    "motion_positioning_interval": 0,
                    "motion_timeout": 60,
                    "motion_autoscan_enabled": true,
                    "alarm_positioning_interval": 30,
                    "alarm_timeout": 60,
                    "alarm_autoscan_enabled": true,
                    "sleep_positioning_interval": 0,
```

```
                    "sleep_autoscan_enabled": true,
                    "alerts_enabled": true,
                    "button_enabled": true,
                    "accelerometer_mode": "motion_detection",
                    "motion_threshold": 20,
                    "motion_duration": 3,
                    "beaconing": "disabled"
                }
            }
        ]
    }
}
```


Tab. 9: getMeshConfiguration

Name	Type	Description
diagnostics_interval	number	Diagnostics interval of the devices
configuration	array	Array containing configuration parameters for each possible node class and/or packet type

Tab. 10: getMeshConfiguration - Anchor Tag configuration parameters

Name	Type	Description
node_class	string	F9 is the Anchor Tag node class
packet_type	string	A2 is the Anchor Tag configuration packet type
parameters	object	Configuration parameters
sensing_interval	number	Sensing interval for the sensors
alerts_enabled	bool	Whether the alerts are enabled
beaconing	string	Beaconing configuration: • disabled • ibeacon • eddystone
beaconing_advertising_interval	string	Advertising interval if beaconing is enabled
beaconing_advertising_tx_power	number	Advertising TX power if beaconing in enabled
ibeacon_uuid	string	IBeacon UUID if iBeacon beaconing is enabled
ibeacon_major_format	number	If iBeacon beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address MSB
ibeacon_minor_format	number	If iBeacon beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address LSB

Name	Type	Description
eddystone_uid_namespace	string	Eddystone-UID Namespace if Eddystone-UID is enabled
eddystone_uid_instance_id_format	number	Possible values of Eddystone-UID InstanceID format if Eddystone-UID is enabled: 0-Set to 0 1-Tag model number + Serial number 2-Node Address

Tab. 11: getMeshConfiguration – Mobile Tag configuration parameters

Name	Type	Description
node_class	string	FA is the Mobile Tag node class
packet_type	string	B2 is the Mobile Tag configuration packet type
parameters	object	Configuration parameters
default_positioning_interval	number	Positioning interval when in default state
default_timeout	number	Timeout from default to sleep state
default_autoscan_enabled	bool	Whether autoscan is enabled in default state
motion_positioning_interval	number	Positioning interval when in motion state
motion_timeout	number	Timeout from motion to default state
motion_autoscan_enabled	bool	Whether autoscan is enabled in motion state
alarm_positioning_interval	number	Positioning interval when in alarm state
alarm_timeout	number	Timeout from alarm to default state
alarm_autoscan_enabled	bool	Whether autoscan is enabled in alarm state
sleep_positioning_interval	number	Positioning interval when in sleep state
sleep_autoscan_enabled	bool	Whether autoscan is enabled in sleep state
alerts_enabled	bool	Whether the alerts are enabled
button_enabled	bool	Whether the button is enabled
accelerometer_mode	string	Possible accelerometer modes are: • powerdown • motion_detection • position_detection • position_mandown • freefall • freefall_mandown • shock • shock_mandown
motion_threshold	number	For motion_detection/position_mandown/ freefall_mandown/shock_mandown mode
motion_duration	number	For motion_detection/position_mandown/ freefall_mandown/shock_mandown mode
position_angle	number	For position_detection/position_mandown mode
position_duration	number	For position_detection/position_mandown mode
freefall_height	number	For freefall/freefall_mandown mode
freefall_sensitivity	number	For freefall/freefall_mandown mode
shock_intensity	number	For shock/shock_mandown mode
mandown_debounce_duration	number	For mandown modes
mandown_stationary_duration	number	For mandown modes
mandown_alert_duration	number	For mandown modes
beaconing	string	Beaconing configuration:

		• disabled • quuppa_emulation_mode • ibeacon • safety
default_advertising_interval	number	Default advertising interval if beaconing is enabled
motion_advertising_interval	number	Motion advertising interval if beaconing is enabled
alarm_advertising_interval	number	Alarm advertising interval if beaconing is enabled
sleep_advertising_interval	number	Sleep advertising interval if beaconing is enabled
beaconing_advertising_tx_power	number	Advertising TX power if beaconing in enabled
quuppa_tag_id_format	number	0-MAC Address 1-Tag model number + Serial number 2-Node address
ibeacon_uuid	string	Ibeacon UUID if iBeacon beaconing is enabled
ibeacon_major_format	number	If iBeacon beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address MSB
ibeacon_minor_format	number	If iBeacon beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address LSB
safety_uuid	0	BlueUp Safety UUID if Safety beaconing is enabled
safety_major_format	number	If Safety beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address LSB

2.6. http://<URL>[:port]/bpe/setMeshConfiguration

This command allows to set configuration parameters.

Request:

```
{
    "format": "json",
    "setup": {
        "diagnostics_interval": 30,
        "configuration": [
            {
                "node_class": "F9",
                "packet_type": "A2",
                "parameters": {
                    "sensing_interval": 300,
                    "alerts_enabled": true,
                    "beaconing": "disabled"
                }
            },
            {
                "node_class": "FA",
                "packet_type": "B2",
                "parameters": {
                    "default_positioning_interval": 60,
                    "default_timeout": 300,
                    "default_autoscan_enabled": false,
                    "motion_positioning_interval": 0,
                    "motion_timeout": 60,
                    "motion_autoscan_enabled": true,
                    "alarm_positioning_interval": 30,
                    "alarm_timeout": 60,
                    "alarm_autoscan_enabled": true,
                    "sleep_positioning_interval": 0,
                    "sleep_autoscan_enabled": true,
                    "alerts_enabled": true,
                    "button_enabled": true,
                    "accelerometer_mode": "motion_detection",
                    "motion_threshold": 15,
                    "motion_duration": 1,
                    "beaconing": "disabled"
```

```
                }
            }
        ]
    }
}
```

Tab. 12: setMeshConfiguration

Name	Type	Description
format	string	Format of the response. Possible values are: • json • hex • base64 The default format is json.
setup	object	Object containing setup parameters
diagnostics_interval	number	Diagnostics interval of the devices
configuration	array	Array containing configuration parameters

Tab. 13: setMeshConfiguration – Anchor Tag configuration object

Name	Type	Description
node_class	string	F9 is the Anchor Tag node class
packet_type	string	A2 is the Anchor Tag configuration packet type
parameters	object	Configuration parameters
sensing_interval	number	Sensing interval for the sensors
alerts_enabled	bool	Whether the alerts are enabled
beaconing	string	Beaconing configuration: • disabled • ibeacon • eddystone
beaconing_advertising_interval	string	Advertising interval if beaconing is enabled
beaconing_advertising_tx_power	number	Advertising TX power if beaconing in enabled
ibeacon_uuid	string	IBeacon UUID if iBeacon beaconing is enabled
ibeacon_major_format	number	If iBeacon beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address MSB
ibeacon_minor_format	number	If iBeacon beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address LSB
eddystone_uid_namespace	string	Eddystone-UID Namespace if Eddystone-UID is enabled
eddystone_uid_instance_id_format	number	Possible values of Eddystone-UID InstanceID format if Eddystone-UID is enabled: 0-Set to 0 1-Tag model number + Serial number 2-Node Address

Tab. 14: setMeshConfiguration – Mobile Tag configuration object

Name	Type	Description
node_class	string	FA is the Mobile Tag node class
packet_type	string	B2 is the Mobile Tag configuration packet type
parameters	object	Configuration parameters
default_positioning_interval	number	Positioning interval when in default state
default_timeout	number	Timeout from default to sleep state
default_autoscan_enabled	bool	Whether autoscan is enabled in default state
motion_positioning_interval	number	Positioning interval when in motion state
motion_timeout	number	Timeout from motion to default state
motion_timeout_enabled	bool	Whether autoscan is enabled in motion state
alarm_positioning_interval	number	Positioning interval when in alarm state
alarm_timeout	number	Timeout from alarm to default state
alarm_autoscan_enabled	bool	Whether autoscan is enabled in alarm state
sleep_positioning_interval	number	Positioning interval when in sleep state
sleep_autoscan_enabled	bool	Whether autoscan is enabled in sleep state
alerts_enabled	bool	Whether the alerts are enabled
button_enabled	bool	Whether the button is enabled
accelerometer_mode	string	Possible accelerometer modes are: • powerdown • motion_detection • position_detection • position_mandown • freefall • freefall_mandown • shock • shock_mandown
motion_threshold	number	For motion_detection/position_mandown/ freefall_mandown/shock_mandown mode
motion_duration	number	For motion_detection/position_mandown/ freefall_mandown/shock_mandown mode
position_angle	number	For position_detection/position_mandown mode
position_duration	number	For position_detection/position_mandown mode
freefall_height	number	For freefall/freefall_mandown mode
freefall_sensitivity	number	For freefall/freefall_mandown mode
shock_intensity	number	For shock/shock_mandown mode
mandown_debounce_duration	number	For mandown modes
mandown_stationary_duration	number	For mandown modes
mandown_alert_ duration	number	For mandown modes
beaconing	string	Beaconing configuration: • disabled • quuppa_emulation_mode • ibeacon • safety
default_advertising_interval	number	Default advertising interval if beaconing is enabled
motion_advertising_interval	number	Motion advertising interval if beaconing is enabled
alarm_advertising_interval	number	Alarm advertising interval if beaconing is enabled
sleep_advertising_interval	number	Sleep advertising interval if beaconing is enabled

beaconing_advertising_tx_power	number	Advertising TX power if beaconing in enabled
quuppa_tag_id_format	number	0-MAC Address 1-Tag model number + Serial number 2-Node address
ibeacon_uuid	string	IBeacon UUID if iBeacon beaconing is enabled
ibeacon_major_format	number	If iBeacon beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address MSB
ibeacon_minor_format	number	If iBeacon beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address LSB
safety_uuid	0	BlueUp Safety UUID if Safety beaconing is enabled
safety_major_format		If Safety beaconing is enabled, possible values are: 0-Set to 0 1-Tag model number 2-Tag serial number 3-Node Address LSB

2.7. http://<URL>[:port]/bpe/getMeshNodeInfo

This request allows to receive Mesh Node Information.

Method: Get / Post

Request:

```
{
   "role": <number*>, #optional
    "network_id": <network_id>, #optional
    "address": <node_address> #optional
}
```

*Role: if this parameter is not specified, the response contains all the Nodes Information, otherwise the possible values are:

- 1 = Router
- 2 = Non Router
- 4 = Sink

2.7.1 Response example for Sink Nodes

```
{
    "data": {
        "count": 8,
        "items": [
            {
                "from_cache": false,
                "id": "S2-123456",
                "address": 2,
                "network_id": 123456,
                "role": "sink_node",
                "low_latency": true,
                "autorole": false,
                "max_buffer_usage": 0,
                "avg_buffer_usage": 0,
                "routers_in_neighborhood": 2,
                "scans": 53,
                "routing": {
                    "advertised_cost": 0,
                    "sink_address": 2,
                    "next_hop": {
```

```
                        "address": 2,
                        "quality": 100,
                        "rssi": 0,
                        "power": 0
                    }
                },
                "map": {
                    "id": 1,
                    "description": "BlueUp"
                },
                "position": {
                    "x": 9.84, "y": 3.54, "z": 0  },
                "zone": {
                    "id": "development",
                    "description": "Area sviluppo",
                    "color": "#f0008a",
                    "z": 0,
                    "vertexes": [ ~ ]
                },
                "gateway_id": "BWGW-0000000099c637f4",
                "anchor": {
                    "id": "ANCHOR_S2-123456",
                    "positioning_enabled": true,
                    "calibrated_rssi": -50,
                    "rssi_offset": 0,
                    "n": 1
                },
                "last_update": 1636039179,
                "last_update_elapsed": {},
                "status": "online"
            },
        ]
    }
}
```

Tab. 15: getMeshNodeInfo Sink Node response

Name	Type	Description
status	bool	Response status true|false
data	object	Contains the response parameters
count	number	Number of Nodes
from_cache	bool	Whether the information is recovered from the previous save
id	string	Sink ID
network_id	string	Node Network ID
role	string	Node Role
low_latency	bool	Whether the Node is Low Latency
autorole	bool	Whether the Node is in Autorole Mode
max_buffer_usage	number	Maximum amount of used buffers
avg_buffer_usage	number	Average amount of used buffers
routers_in_neighborhood	number	Number of Routers in Neighborhood
scans	number	Number of Scans performed by the Node
routing	object	Provides information about the routing of a Node in the Mesh network
advertised_cost	number	Price of the route
sink_address	number	Sink node to which the Locator refers
next_hop	object	Information about the next hop of the routing
address	number	Next Hop Node address
quality	number	Quality of the next hop
rssi	number	Average RSSI of the next hop
power	number	Radio power used to hop
map	object	Information about the Map where the Mesh Locator is positioned. It contains information about the Map Id number and the Map Description
position	object	Coordinates of the position of the Locator on the Map
zone	object	Information about the zone where the Node is placed
gateway_id	string	Gateway Id
anchor	object	Anchor Information
id	string	Anchor ID
positioning_enabled	bool	Whether the Locator contributes to the Tags Positioning
calibrated_rssi	number	Value of the Calibrated RSSI at 1 meter
rssi_offset	number	Offset set to adjust the received RSSI if it's higher or lower than the calibrated one
n	number	Environmental Factor
last_update	number	UNIX timestamp of the last update
last_update_elapsed	object	Time elapsed from the last update
status	string	Node status

2.7.2 Response example for Router / Anchor Nodes


```
{
    "data": {
        "count": 8,
        "items": [
            {
                "from_cache": false,
                "id": "R9000007-123456",
                "address": 9000007,
                "network_id": 123456,
                "device": {
                    "model": 9,
                    "name": "Ultra",
                    "serial": 7
                },
                "version": "3.0",
                "node_class": "F9",
                "operation_mode": "opportunistic_anchor",
                "alarm": "none",
                "state": "default",
                "role": "router_node",
                "low_latency": false,
                "autorole": false,
                "max_buffer_usage": 0,
                "avg_buffer_usage": 0,
                "routers_in_neighborhood": 2,
                "scans": 3,
                "boot_count": 0,
                "battery": 3012,
                "routing": {
                    "advertised_cost": 0.01,
                    "sink_address": 2,
                    "next_hop": {
                        "address": 2,
                        "quality": 100,
                        "rssi": -40,
                        "power": -12
                    }
```

```
        },
        "configuration": {
            "sensing_interval": 300,
            "alerts_enabled": true,
            "beaconing": "disabled"
        },
        "map": {
            "id": 1,
            "description": "BlueUp"
        },
        "position": { "x": 9.93, "y": 2.19, "z": 0 },
        "zone": {
            "id": "development",
            "description": "Area sviluppo",
            "color": "#f0008a",
            "z": 0,
            "vertexes": [ ~ ]
        },
        "anchor": {
            "id": "ANCHOR_R9000007-123456",
            "positioning_enabled": true,
            "calibrated_rssi": -50,
            "rssi_offset": 0,
            "n": 1
        },
        "sensor": {
            "id": "SENSOR_R9000007-123456",
            "temperature": {
                "unit": "celsius",
                "value": 26
            },
            "humidity": 42,
            "pressure": 97816,
            "co2": {
                "ppm": 1102,
                "classification": "medium"
            }
        },
        "last_update": 1636039173,
        "last_update_elapsed": {
```

```
                    "sec": 6
                },
                "status": "online"
            }
        ]
    }
}
```


Tab. 16: getMeshNodeInfo Router Node response

Name	Type	Description
status	bool	Response status true|false
data	object	Contains the response parameters
count	number	Number of devices
from_cache	bool	Whether the information is recovered from the previous save
id	string	Node ID
address	string	Node Address
network_id	string	Node Network ID
device	object	Object containing Device Information, such as Model Number, Model Name, Serial Number
version	string	Firmware Version
node_class	string	Node Class, where F9 means a Router / Anchor Node
operation_mode	string	Router Node Operating Mode.
alarm	string	What type of alarm was sent
state	string	Node State. Possible values are: • Default • Alarm • Motion • Sleep
role	string	Node Role
low_latency	bool	Whether the Node is Low Latency
autorole	bool	Whether the Node is in Autorole Mode
max_buffer_usage	number	Maximum amount of used buffers
avg_buffer_usage	number	Average amount of used buffers
routers_in_neighborhood	number	Number of Routers in Neighborhood
scans	number	Number of Scans performed by the Node
boot_count	number	Number of Reboots of the device
battery	number	Device Battery level
routing	object	Provides information about the routing of a Node in the Mesh network
advertised_cost	number	Price of the route
sink_address	number	Sink node to which the Locator refers
next_hop	object	Information about the next hop of the routing
address	number	Next Hop Node address
quality	number	Quality of the next hop
rssi	number	Average RSSI of the next hop
power	number	Radio power used to hop
configuration	object	Object containing configuration parameters
sensing_interval	number	Sensing interval for the sensors

alerts_enabled	bool	Whether the alerts are enabled
beaconing	string	Beaconing configuration
map	object	Information about the Map where the Mesh Locator is positioned. It contains information about the Map Id number and the Map Description
position	object	Coordinates of the position of the Locator on the Map
zone	object	Information about the zone where the Node is placed
anchor	object	Anchor Information
id	string	Anchor ID
positioning_enabled	bool	Whether the Locator contributes to the Tags Positioning
calibrated_rssi	number	Value of the Calibrated RSSI at 1 meter
rssi_offset	number	Offset set to adjust the received RSSI if it's higher or lower than the calibrated one
n	number	Environmental Factor
sensor	object	Sensor Information
id	string	Sensor ID
temperature	object	Object containing temperature data: Unit and Value
humidity	number	Humidity measured value
pressure	number	Pressure measured value
co2	object	Object containing CO2 data: PPM and Classification based on PPM value
last_update	number	UNIX timestamp of the last update
last_update_elapsed	object	Time elapsed from the last update
status	string	Node status

2.7.3 Response example for Non Router / Tag Nodes

***changes applied after MeshCube v5.5+**


```
{
    "status": true,
    "data": {
        "count": 37,
        "items": [
            {
                "from_cache": false,
                "id": "N10011560-12345678",
                "address": 10011560,
                "network_id": 12345678,
                "alias": "desk tag",
                "device": {
                    "model": 10,
                    "name": "SafeX",
                    "serial": 11560
                },
                "version": "3.7",
                "node_class": "FA",
                "operation_mode": "autoscan_tag",
                "alarm": "none",
                "state": "sleep",
                "state_time": 5069,
                "role": "non_router_node",
                "low_latency": false,
                "autorole": false,
                "max_buffer_usage": 8,
                "avg_buffer_usage": 1,
                "routers_in_neighborhood": 19,
                "scans": 198,
                "battery": 2994,
                "nodes": {
                    "timestamp": 1657102933,
                    "count": 16,
                    "items": [
                        {
                            "role": "router_node",
```

```
                        "id": "<router_id>",
                        "network_id": 12345678,
                        "address": 9000010,
                        "rssi": -42
                    },
                    { ~ }
                ]
            },
            "routing": {
                "advertised_cost": 1,
                "sink_address": 1,
                "next_hop": {
                    "address": 9001641,
                    "quality": 98,
                    "rssi": -55,
                    "power": -8
                }
            },
            "boot_count": 31,
            "stack_version": "5.1.0.97",
            "app_version": "3.7.1.0",
            "scratchpad_stored_sequence": 9,
            "scratchpad_processed_sequence": 255,
            "is_otap_enabled": true,
            "has_configuration": true,
            "last_update": 1657102957,
            "last_update_elapsed": {
                "sec": 13
            },
            "status": "online"
        },
```


Tab. 17: getMeshNodeInfo Non Router Node response

Name	Type	Description
status	bool	Response status true|false
data	object	Contains the response parameters
count	number	Number of devices
from_cache	bool	Whether the information is recovered from the previous save
id	string	Node ID
address	string	Node Address

network_id	string	Node Network ID
alias	string	Alias given to the Node by the user
device	object	Object containing Device Information, such as Model Number, Model Name, Serial Number
version	string	Firmware Version
node_class	string	Node Class, where FA means a Non Router / Tag Node
operation_mode	string	Router Node Operating Mode.
alarm	string	What type of alarm was sent
state	string	Node State. Possible values are: • Default • Alarm • Motion • Sleep
state _time	number	Time elapsed from the last state change
role	string	Node Role
low_latency	bool	Whether the Node is Low Latency
autorole	bool	Whether the Node is in Autorole Mode
max_buffer_usage	number	Maximum amount of used buffers
avg_buffer_usage	number	Average amount of used buffers
routers_in_neighborhood	number	Number of Routers in Neighborhood
scans	number	Number of Scans performed by the Node
battery	number	Device Battery level
nodes	object	Object containing the information about the nodes detected by the tag
role	string	The role of the detected node
id	string	ID of the detected node
network_id	number	Network ID
address	number	Network address
rssi	number	RSSI between the tag and the node
routing	object	Provides information about the routing of a Node in the Mesh network
advertised_cost	number	Price of the route
sink_address	number	Sink node to which the Locator refers
next_hop	object	Information about the next hop of the routing
address	number	Next Hop Node address
quality	number	Quality of the next hop
rssi	number	Average RSSI of the next hop
power	number	Radio power used to hop
boot_count	number	Counter signaling the number of reboots of the device
stack_version	string	Stack version
app_version	string	App version
scratchpad_stored_seq	number	Sequence number stored in scratchpad
scratchpad_processed_seq	number	Processed sequence number
is_otap_enabled	bool	Signals if the over-the-air-programming is enabled
has_configuration	bool	Signals if the device has updated its configuration, using the network configuration
last_update	number	Timestamp of the last tag update
last_update_elapsed	object	Time elapsed from the last update
status	string	Node status

2.8. http://<URL>[:port]/bpe/meshAnchorPositioningSettings

API available starting from MeshCube version 5.6.3.

This API allows to get the positioning settings from an anchor.

Method: Post

Request:

```
{
    "action": "get",
    "target":{
        "role": "router", // or "sink"
        "address": <nodeaddress> // The address of the anchor
    }
}
```

Tab. 18: meshAnchorPositioningSettings parameters

Name	Type	Description
action	string	Specify that the user wants to get the Anchor configuration
role	string	Node role, that can be Router or Sink
address	number	Node address

Response:

The response example is shown below.

```
{
    "status": true,
    "data": {
        "positioning_enabled": true,
        "calibrated_rssi": -51,
        "rssi_offset": 0,
        "n": 0.5,
        "position": {
            "x": 2.14414893617021,
            "y": 13.1485978465283,
            "z": 0
        }
    }
}
```

Tab. 19: meshAnchorPositioningSettings Response

Name	Type	Description
status	bool	Specifies if the request was responded successfully
positioning_enabled	bool	Specifies if the positioning on the anchor is enabled
calibrated_rssi	number	Anchor Calibrated RSSI value
rssi_offset	number	Anchor RSSI offset value
n	number	Environmental factor
position	object	Object containing the coordinates of the anchor position

2.9. http://<URL>[:port]/bpe/meshAnchorPositioningSettings

API available starting from MeshCube version 5.6.3.

This API allows to set the positioning settings of an anchor.

Method: Post

Post payload example:

```
{
    "action": "set",
    "target":{
        "role": "router", // or "sink"
        "address": <nodeaddress>
    },
    "positioning_enabled": true,
    "calibrated_rssi": -50,
    "rssi_offset": 0,
    "n": 1.5,
    "position": {
        "x": 2.14,
        "y": 13.15
    }
}
```

Tab. 20: meshAnchorPositioningSettings set parameters

Name	type	Description
action	string	Specify that the user wants to get the Anchor configuration
role	string	Node role, that can be Router or Sink
address	number	Node address
positioning_enabled	bool	Specifies if the positioning on the anchor is enabled or disabled. Optional (default value is True)
calibrated_rssi	number	Anchor Calibrated RSSI value. Accepted values range [-70dB; -20dB]. Optional
rssi_offset	number	Anchor RSSI offset value. Accepted values range [-20dB; +20dB]. Optional
n	number	Environmental factor. Accepted values range [0.5; 4.0]. Optional
position	object	Object containing the coordinates of the anchor position on the map. Optional

Response:

The response example is shown below.

```
{
    "status": true
}
```

2.10. http://<URL>[:port]/bpe/meshNodeCommand

This request allows to send commands to the Mesh Nodes.

Method: Get / Post

Common parameters to all the Command types are “transmission” and “target”.

The possible transmission types are “unicast”, “broadcast” and “multicast” and the default transmission type is “unicast” and it’s the chosen type if this parameter is omitted.

Target type depends on transmission parameter:

- If “transmission”: “broadcast” the target type is not considered, therefore it can be omitted.
- If “transmission”: “multicast” the target type is a string that can be from “FA” to “FF”.
- If “transmission”: “unicast” the target type can be:
  - a string containing Node_ID
  - a number containing Node Address
  - an object containing “address”: <number>, “network_id” : <number> (optional).

2.10.1 Reboot

This command is used to reboot the device.

Request:

```
{
        "transmission": “unicast” | “multicast” | “broadcast”,
        "target": <target>,
        "command": {
            "type": “reboot”
        }
}
```

2.10.2 Button Switch

This command enables or disables the button.

Request:

```
{
        "transmission": “unicast” | “multicast” | “broadcast”,
        "target": <target>,
        "command": {
            "type": “button_switch”,
            "data": {
                "enabled": true | false
               }
        }
}
```

2.10.3 Alert

This command allows to configure the alarm of the device. The parameters that are contained in the “data” object can be omitted if not needed.

Request:

```
{
        "transmission": “unicast” | “multicast” | “broadcast”,
        "target": <target>,
        "command": {
            "type": “alert”,
            "data": {
                "buzzer_period": 2,
                "buzzer_repeats": 20,
                "buzzer_duty_cycle": 50,
                "vibration_period": 2,
                "vibration_repeats": 20,
                "vibration_duty_cycle": 50,
                "led_period": 2,
                "led_repeats": 20,
                "led_duty_cycle": 50,
                "led_sequential": false,
                "led_red": true,
                "led_green": true,
                "led_blue": true,
                "led_0": true,
                "led_1": true,
                "led_2": true,
                "led_3": true
            }
        }
}
```

2.10.4 Alert Off

This command is used to disable buzzer/vibration/led if they were configured as permanent on in the Alert command.

Request:

```
{
        "transmission": “unicast” | “multicast” | “broadcast”,
        "target": <target>,
        "command": {
            "type": “alert_off”,
             "data": <data parameters can be left specified or can be omitted>
        }
}
```

Equivalent Request: i.e. “alert” without data object.

```
{
        "transmission": “unicast” | “multicast” | “broadcast”,
        "target": <target>,
        "command": {
            "type": “alert”
        }
}
```

2.10.5 GPIO

Commands for GPIO Pin Configuration.

Request:

```
{
 "transmission": “unicast” | “multicast” | “broadcast”,
 "target": <target>,
       "command": {
           "type": “gpio”,
           "data": {
                "pin": <pin_number>,
               "mode": “reset” | “output” | “input”,
               "pullcfg": “pulldown” | “pullup”, #only in Input mode
               "send_interval": <number> #range [0; 255]
              }
       }    }
```

2.11. https://<URL>[:port]/bpe/addEventListener

This request is used to retrieve a list of Events.

Method**:** Post

It's possible to request all the Events at the same time or just some of them.

In Body section of the Post it has to be specified the type of Events requested. To select all Events it's possible to write them all or just write an empty array.

Request:

```
[
	"meshAnchorPositioningSettingsUpdated",
	"meshAnchorRemoved",
	"meshBeaconRemoved",
	"meshConfigurationChanged",
	"meshNodeBatteryLevelChanged",
	"meshNodeBatteryVeryLow",
	"meshNodeRemoved",
	"meshNodeStatusChanged",
	"meshNonRouterNodesDetectionUpdate",
	"meshSensorDataUpdated",
	"meshSensorRemoved",
	"meshSensorTelemetry",
	"meshTagAlarmOff",
	"meshTagAlarmTriggered",
	"meshTagOperationModeChanged",
	"meshTagRemoved",
	"meshTagStateChanged",
	"tagPositionChanged"
]

<is equivalent to an empty array>

[
]
```

When requesting the list of Events the system will generate a Token to use to visualize the events. There are two uses of the token:

1. **Get Method**. http://<URL>[:port]/bpe/getEvents?token=<Token>
2. **Post Method**. http://<URL>[:port]/bpe/getEvents

```
{
       “token”: “<Token>”
}
```

By sending the command the system will send in response a list of events. After every request the memory empties. The following events are saved in the memory until they are requested.

If the events are not requested for 300 seconds (=5 minutes) the token won’t be available any more and there will be an error displayed: ERROR 410: Invalid Token.

2.11.1 meshAnchorPositioningSettingsUpdated

This event is raised when the positioning settings of a Mesh Anchor are changed. The event sums up the new parameter values.

```
{
        "id": "meshAnchorPositioningSettingsUpdated",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
             "id": "ANCHOR_1",
             "node_id": "node_id",
             "positioning_enabled": true | false,
             "calibrated_rssi": -50,
             "rssi_offset": 0,
             "n": 1
        }
}
```

Tab. 21: meshAnchorPositioningSettingsUpdated

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Anchor ID
node_id	string	Node ID
positioning_enabled	bool	Indicates if the positioning is enabled
calibrated_rssi	number	Anchor Calibrated RSSI value
rssi_offset	number	Anchor RSSI offset value
n	number	Environmental factor

2.11.2 meshAnchorRemoved

This event is raised when a Mesh Anchor is removed from the list of Anchors detected by the system.

```
{
        "id": "meshAnchorRemoved",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
             "id": "anchor_id"
        }
    }
```

Tab. 22: meshAnchorRemoved

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Anchor ID

2.11.3 meshBeaconRemoved

This event is raised when a Mesh Beacon is removed from the list of MeshBeacons detected by the system.

```
{
        "id": "meshBeaconRemoved",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
              "type": "tag | anchor",
              "id": <beacon_id>
        }
    }
```

Tab. 23: meshBeaconRemoved

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
type	string	Tag | Anchor
id	string	Beacon ID

2.11.4 meshConfigurationChanged

This event is raised when the Mesh configuration is changed, either via web interface or via API. In this event are summed up all the current parameter values on each Node class.

```
{
        "id": "meshConfigurationChanged",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
               "network_id": <network_id>,
               "setup": [
                      {
                      "node_class": "node_class",
                      "packet_type": "packet_type",
                      "parameters": {
                             <new_parameters>
                         }
                      },
                      {
                      "node_class": "node_class",
                      "packet_type": "packet_type",
                      "parameters": {
                             <new_parameters>
                         }
                      }
               ]
        }
    }
```

Tab. 24: meshConfigurationChanged

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
network_id	number	Network ID
setup	array	Object containing new parameters
node_class	string	Node class
packet_type	string	Packet type
parameters	object	New parameters, as in 2.5

2.11.5 meshNodeBatteryLevelChanged

This event is raised when the battery voltage (in millivolts) of a device changes.

```
{
    "id": "meshNodeBatteryLevelChanged",
    "ticks": 637786307177340630,
    "timestamp": 1643033917,
    "timestamp_ms": 1643033917734,
    "data": {
      "id": "Node_ID",
      "previous_battery_voltage": 2970,
      "battery_voltage": 2958,
      "battery_status": "full"
    }
 }
```

Tab. 25: meshNodeBatteryLevelChanged

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Node ID
previous_battery_voltage	number	Previous battery voltage value
battery_voltage	number	Current battery voltage value
battery_status	string	Battery status: • Full – if voltage ≥ 2900 mV • Half – if voltage ≥ 2800 mV • Low – if voltage ≥ 2700 mV • Very Low – if voltage < 2700 mV

2.11.6 meshNodeBatteryVeryLow

Event raised when the battery of a device goes under 2700 mV. This event is repeated once per hour to notify the Low Battery status of the device.

```
{
  "id": "meshNodeBatteryVeryLow",
  "ticks": 637786324440325150,
  "timestamp": 1643035644,
  "timestamp_ms": 1643035644032,
  "data": {
    "id": "N10011561-12345678",
    "battery_voltage": 2682
  }
},
```

Tab. 26: meshNodeBatteryVeryLow

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Node ID
battery_voltage	number	Current battery voltage

2.11.7 meshNodeRemoved

This event is raised when a Mesh Node is removed from the system. In the event is specified the ID of the removed node and its role, that can be Sink, Router or Non-Router node.

```
{
        "id": "meshNodeRemoved",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
                "role": "sink | router | non_router",
                "id": "node_id"
        }
    }
```

Tab. 27: meshNodeRemoved

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
role	string	Node Role: sink | router | non-router
id	string	Node ID

2.11.8 meshNodeStatusChanged

This event is raised when the status of a Mesh Node changes, specifying the previous and the current state of the Node.

```
{
        "id": "meshNodeStatusChanged",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
                "id": "node_id"
                "previous_status": “<previous_status>”,
                "current_status": “<current_status>”
        }
    }
```

Tab. 28: meshNodeStatusChanged

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Node ID
previous_status	string	Previous status: None | Online | Offline
current_status	string	Current status: None | Online | Offline

2.11.9 meshNonRouterNodesDetectionUpdate

This event is raised when there is an update of the router nodes that detect the Tag, with respect to the last update.


```
[
    {
        "id": "meshNonRouterNodesDetectionUpdate",
        "ticks": 637932135366172160,
        "timestamp": 1657616736,
        "timestamp_ms": 1657616736617,
        "data": {
            "from_cache": false,
            "id": "N5019479-12345678",
            "address": 5019479,
            "network_id": 12345678,
            "alias": "Dev 2",
            "device": {
                "model": 5,
                "name": "Tag",
                "serial": 19479
            },
            "version": "3.7",
            "node_class": "FA",
            "operation_mode": "autoscan_tag",
            "alarm": "none",
            "state": "sleep",
            "state_time": 2969,
            "role": "non_router_node",
            "low_latency": false,
            "autorole": false,
            "max_buffer_usage": 8,
            "avg_buffer_usage": 2,
            "routers_in_neighborhood": 19,
            "scans": 72,
            "battery": 2700,
            "nodes": {
                "timestamp": 1657616736,
                "travel_time": 4609,
                "count": 16,
```

```
                "items": [
                    {
                        "role": "router_node",
                        "id": "R4051017-12345678",
                        "network_id": 12345678,
                        "address": 4051017,
                        "rssi": -39
                    },
                    { }
                ]
            },
            "routing": {
                "advertised_cost": 1,
                "sink_address": 1,
                "next_hop": {
                    "address": 9000001,
                    "quality": 97,
                    "rssi": -57,
                    "power": -8
                }
            },
            "boot_count": 6,
            "stack_version": "5.1.0.97",
            "app_version": "3.7.1.0",
            "scratchpad_stored_sequence": 9,
            "scratchpad_processed_sequence": 255,
            "is_otap_enabled": true,
            "has_configuration": true,
            "last_update": 1657616680,
            "last_update_elapsed": {
                "sec": 55
            },
            "status": "online"
        }
    },
```

Tab. 29: meshNonRouterNodesDetectionUpdate

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
data	object	Contains the response parameters
from_cache	bool	Whether the information is recovered from the previous save
id	string	Node ID
address	string	Node Address
network_id	string	Node Network ID
alias	string	Alias given to the Node by the user
device	object	Object containing Device Information, such as Model Number, Model Name, Serial Number
version	string	Firmware Version
node_class	string	Node Class, where FA means a Non Router / Tag Node
operation_mode	string	Router Node Operating Mode.
alarm	string	What type of alarm was sent
state	string	Node State. Possible values are: • Default • Alarm • Motion • Sleep
state _time	number	Time elapsed from the last state change
role	string	Node Role
low_latency	bool	Whether the Node is Low Latency
autorole	bool	Whether the Node is in Autorole Mode
max_buffer_usage	number	Maximum amount of used buffers
avg_buffer_usage	number	Average amount of used buffers
routers_in_neighborhood	number	Number of Routers in Neighborhood
scans	number	Number of Scans performed by the Node
battery	number	Device Battery level
nodes	object	Object containing the information about the nodes detected by the tag
travel_time	number	[from v.5.6.6] Time for the event to travel from the device to the system
role	string	The role of the detected node
id	string	ID of the detected node
network_id	number	Network ID
address	number	Network address
rssi	number	RSSI between the tag and the node
routing	object	Provides information about the routing of a Node in the Mesh network
advertised_cost	number	Price of the route
sink_address	number	Sink node to which the Locator refers
next_hop	object	Information about the next hop of the routing
address	number	Next Hop Node address
quality	number	Quality of the next hop
rssi	number	Average RSSI of the next hop
power	number	Radio power used to hop

boot_count	number	Counter signalling the number of reboots of the device
stack_version	string	Stack version
app_version	string	App version
scratchpad_stored_seq	number	Sequence number stored in scratchpad
scratchpad_processed_seq	number	Processed sequence number
is_otap_enabled	bool	Signals if the over-the-air-programming is enabled
has_configuration	bool	Signals if the device has updated its configuration, using the network configuration
last_update	number	Timestamp of the last tag update
last_update_elapsed	object	Time elapsed from the last update
status	string	Node status

2.11.10 meshSensorDataUpdated

This event is raised when a Mesh Sensor updates its telemetry values. It contains data of the possible supported sensors, that can be Temperature, Humidity, Pressure and CO2 with relative values.
In case of the CO2 sensor, a classification of the air quality is provided, based on the CO2 value.

```
{
 "id": "meshSensorDataUpdated",
 "ticks": 637780353049195300,
 "timestamp": 1642438504,
 "timestamp_ms": 1642438504919,
 "data": {
   "id": "SENSOR_R9000001-12345678",
   "temperature": {
     "unit": "celsius",
     "value": 23.7
   },
   "humidity": 34,
   "pressure": 99840,
   "co2": {
     "ppm": 976,
     "classification": "good"
   }
 }
},
```

Tab. 30: meshSensorDataUpdated

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Sensor ID
temperature	object	Temperature Information, containing measurement unit and measured value
humidity	number	Humidity value
pressure	number	Pressure value
co2	object	CO2 information, containing the ppm value and classification
classification	string	Air quality classification: • Excellent if CO2 < 700 ppm • Good if CO2 > 700ppm • Medium if CO2 > 1000ppm • Bad if CO2 > 1600ppm

2.11.11 meshSensorRemoved

Event raised when a Sensor is removed from the list of MeshSensors detected by the system.

```
{
        "id": "meshSensorRemoved",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
              "id": "sensor_id"
        }
    }
```

Tab. 31: meshSensorRemoved

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Sensor ID

2.11.12 meshSensorTelemetry

This event is raised when a Mesh Sensor updates its telemetry values.


```
{
    "id": "meshSensorTelemetry",
    "ticks": 637780353049195620,
    "timestamp": 1642438504,
    "timestamp_ms": 1642438504919,
    "data": {
      "id": "SENSOR_R9000001-12345678",
      "data": [
        {
          "type": "temperature",
          "unit": "celsius",
          "value": 23.7
        },
        {
          "type": "humidity",
          "unit": "percentage",
          "value": 34
        },
        {
          "type": "pressure",
          "unit": "pa",
          "value": 99840
        },
        {
          "type": "co2",
          "unit": "ppm",
          "value": 976
        },
        {
          "type": "co2",
          "unit": "classification",
          "value": "good"
        }
      ]
    }
 },
```

Tab. 32: meshSensorTelemetry

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Sensor ID
data	array	Array containing telemetry data related to temperature, humidity, pressure and CO2
type	string	Type of measurement: Temperature | humidity | pressure | CO2
unit	string	Measurement unit, depending on the type of measurement
value	number/string	Measurement value. In case of CO2 classification, the value is a string classifying the air quality: • Excellent if CO2 < 700 ppm • Good if CO2 > 700ppm • Medium if CO2 > 1000ppm • Bad if CO2 > 1600ppm

2.11.13 meshTagAlarmOff

This event is raised when a Tag switches from Alarm state to Default state, after the alarm trigger is over.

```
{
        "id": "meshTagAlarmOff",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
               "id": <tag_id>,
               "travel_time": 4156
        }
    }
```

Tab. 33: meshTagAlarmOff

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Tag ID
travel_time	number	[from v.5.6.6] Time for the event to travel from the device to the system

2.11.14 meshTagAlarmTriggered

This event is raised when an alarm is triggered on a Mesh Tag.

```
{
        "id": "meshTagAlarmTriggered",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
                "id": <tag_id>,
                "travel_time": 4156,
                "alarm": "<alarm_type>"
        }
    }
```

Tab. 34: meshTagAlarmTriggered

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Tag ID
travel_time	number	[from v.5.6.6] Time for the event to travel from the device to the system
alarm	string	The possible alarm triggers are: • Button pressed (single short press, single long press, double press) • Horizontal position detection (if configured) • Free-fall detection (if configured) • Shock detection (if configured) • Man-Down detection (if configured)

2.11.15 meshTagOperationModeChanged

This event is raised when a Tag changes its operation mode.

Operation modes are:

- NRLS Tag: the tag is in sleep mode between two measurement updates and wakes-up at periodic configurable intervals to perform a network scan, send the collected measurements, receive application configuration and switch back to sleep mode.
- Autoscan Tag: In the autoscan mode the connectivity to the WM is maintained as long as possible throughout the lifetime of the tag. Since the tag is continuously connected, it is possible to send and receive data messages at any moment.

```
{
        "id": "meshTagOperationModeChanged",
        "ticks": 637786313922869950,
        "timestamp": 1643034592,
        "timestamp_ms": 1643034592286,
        "data": {
            "id": "Tag_id",
            "travel_time": 4609,
            "operation_mode": "autoscan_tag"
        }
    },
```

Tab. 35: meshTagOperationModeChanged

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Tag ID
travel_time	number	[from v.5.6.6] Time for the event to travel from the device to the system
operation_mode	string	NRLS Tag or Autoscan Tag

2.11.16 meshTagRemoved

This event is raised when a Mesh Tag is removed from the list of the Mesh Tags detected by the system.

```
{
        "id": "meshTagRemoved",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
                "id": "tag_id"
        }
    }
```

Tab. 36: meshTagRemoved

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Tag ID

2.11.17 meshTagStateChanged

This event is raised when a MeshIPS Tag changes its state.

```
{
        "id": "meshTagStateChanged",
        "ticks": 637780349000281660,
        "timestamp": 1642438100,
        "timestamp_ms": 1642438100028,
        "data": {
            "id": "TAG_ID",
            "travel_time": 4156,
            "current_state": "motion",
            "previous_state": "sleep"
        }
    }
```

Tab. 37: MmeshTagStateChanged

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Trigger ID
travel_time	number	[from v.5.6.6] Time for the event to travel from the device to the system
current_state	string	Current State
previous_state	string	Previous State. Possible States are: • Default • Motion • Sleep • Alarm

2.11.18 tagPositionChanged

Event raised when a Tag changes its position.

```
{
    "id": "tagPositionChanged",
    "ticks": 637780350735169710,
    "timestamp": 1642438273,
    "timestamp_ms": 1642438273516,
    "data": {
        "id": "tag_id",
        "node_id": "node_id",
        "node_address": <node_address>,
        "node_network_id": <node_network_id>,
        "node_role": "node_role",
        "type": "mesh",
        "from_cache": false,
        "version": "3.3",
        "node_class": "FA",
        "operation_mode": "autoscan_tag",
        "alarm": "none",
        "state": "sleep",
        "last_update": 1642438209,
        "last_update_elapsed": {},
        "battery": 2994,
        "anchors": 13,
        "anchor": [
            {
                "id": "ANCHOR_1",
                "address": 2,
                "last_update": 1642438273,
                "rssi": -42,
                "detection_radius": 0.4,
                "position": {}
            },
            { ~ }
        ],
        "localization_mode": "trilateration",
        "location": {
        "updated_on": 1687939641,
        "travel_time": 4156,
        "map": 1,
            "map_description": "BlueUp",
            "position": { },
            "zone": {
                "id": "development",
```

```
                "description": "Area sviluppo"
            },
            "accuracy": 1.45,
            "anchors": [
                {
                    "id": "ANCHOR_R9000008-12345678",
                    "address": 9000008,
                    "last_update": 1642438273,
                    "rssi": -51,
                    "detection_radius": 1.17,
                    "position": {}
                },
                { ~ }
            ]
        },
        "status": "online"
    }
},
```

Tab. 38: TagPositionChanged

Name	Type	Description
id	string	Event ID
ticks	number	Date and Time of the Event in Ticks format
timestamp	number	UNIX timestamp of Date and Time of the Event
timestamp_ms	number	UNIX timestamp of Date and Time of the Event in ms
id	string	Tag ID
node_id	string	Node ID
node_address	string	Node Address
node_network_id	string	Node Network ID
node_role	string	Node role: Sink | router | non-router
type	string	Tag type
from_cache	bool	Whether the information are recovered from the previous save
version	string	Firmware version
node_class	string	Node class
operation_mode	string	Node operation mode
alarm	string	Type of the alarm triggered on the tag
state	string	Current Tag state
last_update	number	UNIX timestamp of the last update
last_update_elapsed	object	Time in hours, minutes and seconds from the last update
battery	number	Battery level in mV
anchors	number	Number of anchors that detect the tag
anchor	array	Information about the Anchors that detect the Tag
id	string	Anchor ID
address	number	Node address
last_update	number	UNIX timestamp of the last update
rssi	number	Instant RSSI value of the signal exchanged between

		the Tag and the Locator
detection_radius	number	Location Distance Factor in meters
position	object	Anchor position coordinates
localization_mode	string	Localization mode: Trilateration is the default one
location	object	Contains information about the location of the tag
travel_time	number	[from v.5.6.6] Time for the event to travel from the device to the system
map	number	Map number where the Tag is located
map_description	string	Map description
position	object	Coordinates of the position of the Tag on the Map
zone	object	Information about the Zone where the Tag is located
accuracy	number	Accuracy of the position of the Tag
anchors	array	List of anchors that contribute in positioning
id	string	Anchor ID
address	number	Node address
last_update	number	UNIX timestamp of the last update
rssi	number	Instant RSSI value of the signal exchanged between the Tag and the Locator
detection_radius	number	Location Distance Factor in meters
position	object	Anchor position coordinates
status	string	Tag status