



**inovonics**<sup>®</sup>

TapWatch API Documentation

# Inovonics Core AAA API

Organization, Site, and User Management

More information: <https://helloeverb.com>

Contact Info: [support@inovonics.com](mailto:support@inovonics.com)

Version: 1.0.0.10

BasePath: /v1

All rights reserved

<http://apache.org/licenses/LICENSE-2.0.html>

## Access

1. OAuth AuthorizationUrl:TokenUrl:<https://api.tapwatch.com/oauth/token/>
2. OAuth AuthorizationUrl:TokenUrl:<https://api.tapwatch.com/oauth/token/>

## Methods

[ Jump to [Models](#) ]

## Table of Contents

### [Enums](#)

- [GET /enums/timezones/](#)

### [OrganizationUsersSites](#)

- [GET /organizations/{organization\\_id}/users/{user\\_id}/sites/](#)
- [DELETE /organizations/{organization\\_id}/users/{user\\_id}/sites/{site\\_id}/](#)
- [PUT /organizations/{organization\\_id}/users/{user\\_id}/sites/{site\\_id}/](#)

### [Organizations](#)

- [GET /organizations/](#)
- [GET /organizations/{organization\\_id}/](#)
- [PUT /organizations/{organization\\_id}/](#)
- [POST /organizations/](#)

### [OrganizationsSites](#)

- [GET /organizations/{organization\\_id}/sites/](#)
- [POST /organizations/{organization\\_id}/sites/](#)
- [DELETE /organizations/{organization\\_id}/sites/{site\\_id}/](#)
- [PUT /organizations/{organization\\_id}/sites/{site\\_id}/](#)

### [OrganizationsUsers](#)

- [GET /organizations/{organization\\_id}/users/](#)
- [POST /organizations/{organization\\_id}/users/](#)
- [DELETE /organizations/{organization\\_id}/users/{user\\_id}/](#)
- [PUT /organizations/{organization\\_id}/users/{user\\_id}/](#)

### [Sites](#)

- [GET /sites/{site\\_id}/](#)
- [PUT /sites/{site\\_id}/](#)

### [Users](#)

- [GET /users/{user\\_id}/](#)
- [GET /users/{user\\_id}/organizations/](#)
- [PUT /users/{user\\_id}/password/](#)
- [PUT /users/{user\\_id}/](#)
- [PUT /users/{user\\_id}/resetpassword/](#)

# Enums

[Up](#)

GET /enums/timezones/

Get timezones enum (**enumsTimezonesGet**)

Get the list of supported timezones definitions.

### Return type

array[[EnumTimeZones](#)]

### Example data

Content-Type: application/json

```
[ {
  "offset" : "offset",
  "name" : "name",
  "short_name" : "short_name",
  "id" : "id"
}, {
  "offset" : "offset",
  "name" : "name",
  "short_name" : "short_name",
  "id" : "id"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

# OrganizationUsersSites

[Up](#)

GET /organizations/{organization\_id}/users/{user\_id}/sites/

List organization user's sites (**organizationsOrganizationIdUsersUserIdSitesGet**)

List of the sites which belong to the specified organization's user.

### Path parameters

**organization\_id (required)**

*Path Parameter* – UUID of the organization.

**user\_id (required)**

*Path Parameter* – UUID of the user.

### Return type

array[[UserSite](#)]

### Example data

Content-Type: application/json

```
[ {
  "end_date" : "end_date",
  "note" : "note",
  "address" : "address",
  "code" : "code",
  "timezone" : "timezone",
```

```

    "editable" : true,
    "site_id" : "site_id",
    "name" : "name",
    "start_date" : "start_date"
  }, {
    "end_date" : "end_date",
    "note" : "note",
    "address" : "address",
    "code" : "code",
    "timezone" : "timezone",
    "editable" : true,
    "site_id" : "site_id",
    "name" : "name",
    "start_date" : "start_date"
  } ]

```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

DELETE /organizations/{organization\_id}/users/{user\_id}/sites/{site\_id}/ [Up](#)

Remove organization user-site relationship (**organizationsOrganizationIdUsersUserIdSitesSiteIdDelete**)

Remove the relationship of the specified organization user to a site.

### Path parameters

**organization\_id (required)**

*Path Parameter* – UUID of the organization.

**user\_id (required)**

*Path Parameter* – UUID of the user.

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Responses

204

No Content - Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

PUT /organizations/{organization\_id}/users/{user\_id}/sites/{site\_id}/ [Up](#)

Add or Update organization user and site relationship (**organizationsOrganizationIdUsersUserIdSitesSiteIdPut**)

Add or Update the information for an organization user-site relationship. Any values not supplied will not be modified unless otherwise noted. Any values labeled as read-only will be ignored.

### Path parameters

**organization\_id (required)**

*Path Parameter* – UUID of the organization.

**user\_id (required)**

*Path Parameter* – UUID of the user.

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [UserSiteUpdate](#) (required)

*Body Parameter* – User site object containing data for relationship to be created.

### Return type

[UserSiteUpdate](#)

### Example data

Content-Type: application/json

```
{
  "end_date" : "end_date",
  "start_date" : "start_date"
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [UserSiteUpdate](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## Organizations

GET /organizations/



List organizations (**organizationsGet**)

List of the organizations that a user has permission to view

### Return type

array[[Organization](#)]

### Example data

Content-Type: application/json

```
[ {
  "address" : "address",
  "support_email" : "support_email",
  "phone" : "phone",
  "organization_id" : "organization_id",
  "name" : "name",
  "short_name" : "short_name"
}, {
  "address" : "address",
  "support_email" : "support_email",
  "phone" : "phone",
  "organization_id" : "organization_id",
```

```
"name" : "name",  
"short_name" : "short_name"  
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

## GET /organizations/{organization\_id}/



Retrieve organization information (**organizationsOrganizationIdGet**)

Retrieve the information for an organization.

### Path parameters

**organization\_id (required)**

*Path Parameter* – UUID of the organization.

### Return type

[Organization](#)

### Example data

Content-Type: application/json

```
{  
  "address" : "address",  
  "support_email" : "support_email",  
  "phone" : "phone",  
  "organization_id" : "organization_id",  
  "name" : "name",  
  "short_name" : "short_name"  
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [Organization](#)

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## PUT /organizations/{organization\_id}/



Update organization information (**organizationsOrganizationIdPut**)

Update the information for an organization. Any values not supplied will not be modified unless otherwise noted.

### Path parameters

**organization\_id (required)**

*Path Parameter* – UUID of the organization.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [Organization](#) (required)

*Body Parameter* – Organization object containing data for organization to be updated.

### Return type

[Organization](#)

### Example data

Content-Type: application/json

```
{
  "address" : "address",
  "support_email" : "support_email",
  "phone" : "phone",
  "organization_id" : "organization_id",
  "name" : "name",
  "short_name" : "short_name"
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [Organization](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## POST /organizations/

Add organization (**organizationsPost**)

Add an organization to the TapWatch Cloud. --- FOR IWC Tech Services Only.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [Organization](#) (required)

*Body Parameter* – Organization object containing data for organization to be created.

### Return type

[Organization](#)

### Example data

Content-Type: application/json

```
{
  "address" : "address",
  "support_email" : "support_email",
  "phone" : "phone",
  "organization_id" : "organization_id",
}
```

[Up](#)

```
"name" : "name",  
"short_name" : "short_name"  
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

201

Created [Organization](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

409

Conflict - Resource already exists.

## OrganizationsSites

GET /organizations/{organization\_id}/sites/

[Up](#)

List organization sites ([organizationsOrganizationIdSitesGet](#))

List of the sites which belong to the specified organization.

### Path parameters

**organization\_id (required)**

*Path Parameter* – UUID of the organization.

### Return type

array[[Site](#)]

### Example data

Content-Type: application/json

```
[ {  
  "note" : "note",  
  "address" : "address",  
  "code" : "code",  
  "timezone" : "timezone",  
  "editable" : true,  
  "site_id" : "site_id",  
  "name" : "name"  
}, {  
  "note" : "note",  
  "address" : "address",  
  "code" : "code",  
  "timezone" : "timezone",  
  "editable" : true,  
  "site_id" : "site_id",  
  "name" : "name"  
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope.



404

Not Found - The requested resource does not exist.



## POST /organizations/{organization\_id}/sites/

Create organization site (**organizationsOrganizationIdSitesPost**)

Create a site which belongs to the specified organization.

### Path parameters

**organization\_id (required)***Path Parameter* – UUID of the organization.

### Request body

body **Site** (required)*Body Parameter* – Site object containing data for site to be created.

### Return type

[Site](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "address" : "address",
  "code" : "code",
  "timezone" : "timezone",
  "editable" : true,
  "site_id" : "site_id",
  "name" : "name"
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

201

Created [Site](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

409

Conflict - Resource already exists.



## DELETE /organizations/{organization\_id}/sites/{site\_id}/

Delete organization site relationship (**organizationsOrganizationIdSitesSiteIdDelete**)

Delete the information for an organization-site relationship. This will be used by the organization administrators to delete a site from their portfolios without removing the site from the cloud platform.

### Path parameters

**organization\_id (required)***Path Parameter* – UUID of the organization.**site\_id (required)***Path Parameter* – UUID of the site.

**Responses****204**

No Content - Success

**401**

Unauthorized - Invalid OAuth token or scope.

**404**

Not Found - The requested resource does not exist.

[Up](#)

PUT /organizations/{organization\_id}/sites/{site\_id}/

Update organization site relationship (**organizationsOrganizationIdSitesSiteIdPut**)

Add or Update the information for an organization-site relationship. Any values not supplied will not be modified unless otherwise noted. Any values labeled as read-only will be ignored. --- FOR IWC Tech Services Only.

**Path parameters****organization\_id (required)***Path Parameter* – UUID of the organization.**site\_id (required)***Path Parameter* – UUID of the site.**Consumes**

This API call consumes the following media types via the Content-Type request header:

- application/json

**Return type**[Site](#)**Example data**

Content-Type: application/json

```
{
  "note" : "note",
  "address" : "address",
  "code" : "code",
  "timezone" : "timezone",
  "editable" : true,
  "site_id" : "site_id",
  "name" : "name"
}
```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses****200**Success [Site](#)**400**

Bad Request - The supplied data is invalid.

**401**

Unauthorized - Invalid OAuth token or scope.

**404**

Not Found - The requested resource does not exist.

## OrganizationsUsers

GET /organizations/{organization\_id}/users/

[Up](#)List organization users (**organizationsOrganizationIdUsersGet**)

List of the users which belong to the specified organization.

**Path parameters****organization\_id (required)***Path Parameter* – UUID of the organization.**Return type**array[[OrganizationUser](#)]**Example data**

Content-Type: application/json

```
[ {
  "is_registered" : true,
  "role" : "role",
  "user_id" : "user_id",
  "mobile_phone" : "mobile_phone",
  "last_name" : "last_name",
  "first_name" : "first_name",
  "other_phone" : "other_phone",
  "all_sites" : true,
  "username" : "username"
}, {
  "is_registered" : true,
  "role" : "role",
  "user_id" : "user_id",
  "mobile_phone" : "mobile_phone",
  "last_name" : "last_name",
  "first_name" : "first_name",
  "other_phone" : "other_phone",
  "all_sites" : true,
  "username" : "username"
} ]
```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses****200**

Success

**401**

Unauthorized - Invalid OAuth token or scope.

**404**

Not Found - The requested resource does not exist.

[Up](#)

POST /organizations/{organization\_id}/users/

Add user in a specified organization (**organizationsOrganizationIdUsersPost**)

Add a user to a specific organization. If user doesn't exist, it will be created.

**Path parameters****organization\_id (required)***Path Parameter* – UUID of the organization.**Consumes**

This API call consumes the following media types via the Content-Type request header:

- application/json

**Request body**body [OrganizationUser](#) (required)*Body Parameter* – User object containing data for user to be created including email address as username

**Return type**[OrganizationUser](#)**Example data**

Content-Type: application/json

```

{
  "is_registered" : true,
  "role" : "role",
  "user_id" : "user_id",
  "mobile_phone" : "mobile_phone",
  "last_name" : "last_name",
  "first_name" : "first_name",
  "other_phone" : "other_phone",
  "all_sites" : true,
  "username" : "username"
}

```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses**

201

Success [OrganizationUser](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

409

Conflict - Resource already exists.

DELETE /organizations/{organization\_id}/users/{user\_id}/

[Up](#)

Remove organization user relationship (**organizationsOrganizationIdUsersUserIdDelete**)

Remove the relationship of the specified user to an organization.

**Path parameters****organization\_id (required)***Path Parameter* – UUID of the organization.**user\_id (required)***Path Parameter* – UUID of the user.**Responses**

204

No Content - Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

PUT /organizations/{organization\_id}/users/{user\_id}/

[Up](#)

Update organization user relationship (**organizationsOrganizationIdUsersUserIdPut**)

Add or Update the information for an organization-user relationship. Any values not supplied will not be modified unless otherwise noted. Any values labeled as read-only will be ignored.

**Path parameters****organization\_id (required)***Path Parameter* – UUID of the organization.**user\_id (required)**

*Path Parameter* – UUID of the user.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [OrganizationUserRole](#) (required)

*Body Parameter* – Organization user object containing data for relationship to be created.

### Return type

[OrganizationUserRole](#)

### Example data

Content-Type: application/json

```
{
  "role" : "role",
  "all_sites" : true
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [OrganizationUserRole](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## Sites

GET /sites/{site\_id}/

[Up](#)

Retrieve site information ([sitesSiteIdGet](#))

Retrieve the information for the specified site.

### Path parameters

*site\_id* (required)

*Path Parameter* – UUID of the site.

### Return type

[Site](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "address" : "address",
  "code" : "code",
  "timezone" : "timezone",
  "editable" : true,
  "site_id" : "site_id",
  "name" : "name"
}
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

## Responses

200

Success [Site](#)

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

PUT /sites/{site\_id}/

[Up](#)

Update site information (**sitesSiteIdPut**)

Update the information for the specified site. Any values not supplied will not be modified unless otherwise noted.

## Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

## Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

## Request body

**body [Site](#) (required)**

*Body Parameter* – Site object containing data for site to be updated.

## Return type

[Site](#)

## Example data

Content-Type: application/json

```
{
  "note" : "note",
  "address" : "address",
  "code" : "code",
  "timezone" : "timezone",
  "editable" : true,
  "site_id" : "site_id",
  "name" : "name"
}
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

## Responses

200

Success [Site](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

# Users

[Up](#)

GET /users/{user\_id}/

Retrieve user information (**usersUserIdGet**)

Retrieve the information for the specified user.

## Path parameters

**user\_id (required)**

*Path Parameter* – UUID of the user.

## Return type

[User](#)

## Example data

Content-Type: application/json

```
{
  "is_registered" : true,
  "user_id" : "user_id",
  "mobile_phone" : "mobile_phone",
  "last_name" : "last_name",
  "first_name" : "first_name",
  "other_phone" : "other_phone",
  "username" : "username"
}
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

## Responses

200

Success [User](#)

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

GET /users/{user\_id}/organizations/

[Up](#)

List user organizations (**usersUserIdOrganizationsGet**)

List of the organizations to which the user belongs.

## Path parameters

**user\_id (required)**

*Path Parameter* – UUID of the user.

## Return type

array[[UserOrganization](#)]

## Example data

Content-Type: application/json

```
[ {
  "address" : "address",
  "support_email" : "support_email",
  "role" : "role",
  "phone" : "phone",
  "organization_id" : "organization_id",
}
```

```
"name" : "name",
"short_name" : "short_name",
"all_sites" : true
}, {
"address" : "address",
"support_email" : "support_email",
"role" : "role",
"phone" : "phone",
"organization_id" : "organization_id",
"name" : "name",
"short_name" : "short_name",
"all_sites" : true
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope.

403

Forbidden - Resource unavailable or access not allowed.

404

Not Found - The requested resource does not exist.

PUT /users/{user\_id}/password/



Updates user password ([usersUserIdPasswordPut](#))

Update the password for the specified user. This will require the old and new passwords to be supplied.

### Path parameters

**user\_id (required)**

*Path Parameter* – UUID of the user.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [UserPassword](#) (required)

*Body Parameter* – User object containing data for user to be updated.

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

204

Success

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.





## PUT /users/{user\_id}/

### Update user information (usersUserIdPut)

Update the information for the specified user. Any values not supplied will not be modified unless otherwise noted.

#### Path parameters

**user\_id (required)**

*Path Parameter* – UUID of the user.

#### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

#### Request body

body **User** (required)

*Body Parameter* – User object containing data for user to be updated.

#### Return type

[User](#)

#### Example data

Content-Type: application/json

```
{
  "is_registered" : true,
  "user_id" : "user_id",
  "mobile_phone" : "mobile_phone",
  "last_name" : "last_name",
  "first_name" : "first_name",
  "other_phone" : "other_phone",
  "username" : "username"
}
```

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

#### Responses

200

Success [User](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## PUT /users/{user\_id}/resetpassword/

### Resets user password (usersUserIdResetpasswordPut)

Resets the password for the specified user. This will require the old and new passwords to be supplied.

#### Path parameters

**user\_id (required)**

*Path Parameter* – UUID of the user.

#### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json



## Request body

body [UserResetPassword](#) (required)

*Body Parameter* – User object containing data for user to be updated.

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

## Responses

204

Success

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## Models

[ [Jump to Methods](#) ]

### Table of Contents

1. [EnumTimeZones](#) -
2. [Organization](#) -
3. [OrganizationUser](#) -
4. [OrganizationUserRole](#) -
5. [Site](#) -
6. [User](#) -
7. [UserOrganization](#) -
8. [UserPassword](#) -
9. [UserResetPassword](#) -
10. [UserSite](#) -
11. [UserSiteUpdate](#) -

#### EnumTimeZones -

[Up](#)

id (optional)

[String](#) Time zone string

name (optional)

[String](#) Display descriptive time zone string.

short\_name (optional)

[String](#) Display time zone abbreviation

offset (optional)

[String](#) Timezone offset from the UTC.

#### Organization -

[Up](#)

organization\_id (optional)

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

name

[String](#) Maximum of 127 characters. Must be unique.

short\_name (optional)

[String](#) Minimum of 3 characters. Maximum of 12 characters.

address (optional)

[String](#) Maximum of 4096 characters.

phone

[String](#) Maximum of 30 characters. Limited to any combination of 0123456789.()-#\*wp and spaces.

support\_email

[String](#) Maximum of 127 characters. API will validate email format. Web App will validate email format.

## OrganizationUser -

[Up](#)

### user\_id (optional)

[String](#) Read-only property. Will hold the user\_id from the User object.

### username

[String](#) Will hold the username from the User object.

### first\_name (optional)

[String](#) Will hold the first\_name from the User object.

### last\_name (optional)

[String](#) Will hold the last\_name from the User object.

### mobile\_phone (optional)

[String](#) Will hold the mobile phone from the User object. Maximum of 30 characters. Limited to any combination of 0123456789.()-#\*wp and spaces.

### other\_phone (optional)

[String](#) Will hold the other phone from the User object. Maximum of 30 characters. Limited to any combination of 0123456789.()-#\*wp and spaces.

### is\_registered (optional)

[Boolean](#) Flag to indicate if a user is registered i.e. the password for the user is set.

### role

[String](#) Enum: ['admin', 'technician', 'viewer'] Will hold the text for the enum value.

### all\_sites (optional)

[Boolean](#) Setting flag to true allows user to access all sites for the organization. Defaults to 'False'.

## OrganizationUserRole -

[Up](#)

### role

[String](#) Enum: ['admin', 'technician', 'viewer'] Will hold the text for the enum value.

### all\_sites

[Boolean](#) Setting flag to true allows user to access all sites for the organization. Defaults to 'False'.

## Site -

[Up](#)

### site\_id

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

### name

[String](#) Maximum of 127 characters.

### address (optional)

[String](#) Maximum 4096 characters. This will be a text field for containing the entire address.

### timezone (optional)

[String](#) Enum: Will hold the text for the timezone that the property is located within.

### code

[String](#) A separate code used to identify sites. Maximum of 127 characters. Must be unique.

### note (optional)

[String](#) Maximum of 4096 characters.

### editable (optional)

[Boolean](#) Indicates if the site is editable.

## User -

[Up](#)

### user\_id (optional)

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

### username

[String](#) Must be unique. Maximum of 127 characters. Must be an email. API will validate email format.

### first\_name

[String](#) Maximum of 127 characters.

last\_name

[String](#) Maximum of 127 characters.

mobile\_phone (optional)

[String](#) Maximum of 30 characters. Limited to any combination of 0123456789.()-#\*wp and spaces.

other\_phone (optional)

[String](#) Maximum of 30 characters. Limited to any combination of 0123456789.()-#\*wp and spaces.

is\_registered (optional)

[Boolean](#) Flag to indicate if a user is registered i.e. the password for the user is set.

#### UserOrganization -

[Up](#)

organization\_id (optional)

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

name (optional)

[String](#) Maximum of 127 characters. Must be unique.

short\_name (optional)

[String](#) Minimum of 3 characters. Maximum of 12 characters.

address (optional)

[String](#) Maximum of 4096 characters.

phone (optional)

[String](#) Maximum of 30 characters. Maximum of 30 characters. Limited to any combination of 0123456789.()-#\*wp and spaces.

support\_email (optional)

[String](#) Maximum of 127 characters. API will validate email format. Web App will validate email format.

role (optional)

[String](#) Enum: ['admin', 'technician', 'viewer'] Will hold the text for the enum value.

all\_sites (optional)

[Boolean](#) Setting flag to true allows user to access all sites for the organization. Enforced to 'True' for administrator. Defaults to 'False' for others.

#### UserPassword -

[Up](#)

current\_password

[String](#) Current password for the user. API will validate the current user login session.

new\_password

[String](#) New password for the user. API will validate the password. Should be minimum 8 characters, maximum 127 characters and contain no fewer than 1 lowercase, 1 uppercase, and 1 numeric (symbols are optional).

#### UserResetPassword -

[Up](#)

new\_password

[String](#) New password for the user. API will validate the password. Should be minimum 8 characters, maximum 127 characters and contain no fewer than 1 lowercase, 1 uppercase, and 1 numeric (symbols are optional).

#### UserSite -

[Up](#)

site\_id (optional)

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

name (optional)

[String](#) Maximum of 127 characters.

address (optional)

[String](#) Maximum 4096 characters. This will be a text field for containing the entire address.

timezone (optional)

[String](#) Enum: Will hold the text for the timezone that the property is located within.

code (optional)

[String](#) A separate code used to identify sites. Maximum of 127 characters. Must be unique.

note (optional)

[String](#) Maximum of 4096 characters.

editable (optional)

[Boolean](#) Indicates if the site is editable

start\_date (optional)

[String](#) ISO8601 datetime. Empty string if not set.

end\_date (optional)

[String](#) ISO8601 datetime. Empty string if not set.

UserSiteUpdate -

[Up](#)

start\_date (optional)

[String](#) ISO8601 datetime. Empty string if not set.

end\_date (optional)

[String](#) ISO8601 datetime. Empty string if not set.

# Inovonics Core Devices API

API to allow applications, dataloggers, and users to push or pull data to/from the cloud.

More information: <https://helloverb.com>

Contact Info: [support@inovonics.com](mailto:support@inovonics.com)

Version: 1.0.0.3

BasePath: /v1

All rights reserved

<http://apache.org/licenses/LICENSE-2.0.html>

## Access

1. OAuth AuthorizationUrl:TokenUrl:<https://api.tapwatch.com/oauth/token/>
2. OAuth AuthorizationUrl:TokenUrl:<https://api.tapwatch.com/oauth/token/>

## Methods

[ [Jump to Models](#) ]

## Table of Contents

### [Enums](#)

- [GET /enums/gatewaymodel/](#)
- [GET /enums/repeatermodel/](#)
- [GET /enums/transmittermodel/](#)

### [Gateways](#)

- [DELETE /sites/{site\\_id}/gateways/{gateway\\_id}/](#)
- [GET /sites/{site\\_id}/gateways/{gateway\\_id}/](#)
- [PUT /sites/{site\\_id}/gateways/{gateway\\_id}/](#)
- [GET /sites/{site\\_id}/gateways/](#)
- [POST /sites/{site\\_id}/gateways/](#)

### [Repeaters](#)

- [GET /sites/{site\\_id}/repeaters/](#)
- [POST /sites/{site\\_id}/repeaters/](#)
- [DELETE /sites/{site\\_id}/repeaters/{repeater\\_id}/](#)
- [GET /sites/{site\\_id}/repeaters/{repeater\\_id}/](#)
- [PUT /sites/{site\\_id}/repeaters/{repeater\\_id}/](#)

### [Transmitters](#)

- [GET /sites/{site\\_id}/transmitters/](#)
- [POST /sites/{site\\_id}/transmitters/](#)
- [DELETE /sites/{site\\_id}/transmitters/{transmitter\\_id}/](#)
- [GET /sites/{site\\_id}/transmitters/{transmitter\\_id}/](#)
- [PUT /sites/{site\\_id}/transmitters/{transmitter\\_id}/](#)

## Enums



## GET /enums/gatewaymodel/

Get gateway model enum (**enumsGatewaymodelGet**)

Get the list of gateway model definitions.

### Return type

array[[EnumGatewayModel](#)]

### Example data

Content-Type: application/json

```
[ {
  "name" : "name",
  "id" : "id"
}, {
  "name" : "name",
  "id" : "id"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

## GET /enums/repeatermodel/



Get repeater model enum (**enumsRepeatermodelGet**)

Get the list of repeater model definitions.

### Return type

array[[EnumRepeaterModel](#)]

### Example data

Content-Type: application/json

```
[ {
  "name" : "name",
  "id" : "id"
}, {
  "name" : "name",
  "id" : "id"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

[Up](#)

GET /enums/transmittermodel/

Get transmitter model enum (**enumsTransmittermodelGet**)

Get the list of transmitter model definitions.

### Return type

array[[EnumTransmitterModel](#)]

### Example data

Content-Type: application/json

```
[ {
  "name" : "name",
  "id" : "id"
}, {
  "name" : "name",
  "id" : "id"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

## Gateways

[Up](#)

DELETE /sites/{site\_id}/gateways/{gateway\_id}/

Delete gateway (**sitesSiteIdGatewaysGatewayIdDelete**)

Remove the specified gateway.

### Path parameters

**site\_id** (required)

*Path Parameter* – UUID of the site.

**gateway\_id** (required)



*Path Parameter* – UUID of the gateway.

## Responses

204

No Content - Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

[Up](#)

GET /sites/{site\_id}/gateways/{gateway\_id}/

Get gateway (**sitesSiteIdGatewaysGatewayIdGet**)

Get the information for the specified gateway.

## Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**gateway\_id (required)**

*Path Parameter* – UUID of the gateway.

## Return type

[Gateway](#)

## Example data

Content-Type: application/json

```
{
  "note" : "note",
  "last_sync_date" : "last_sync_date",
  "authorization_code" : "authorization_code",
  "serial_number" : "serial_number",
  "location" : "location",
  "model" : "model",
  "gateway_id" : "gateway_id"
}
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

## Responses

200

Success [Gateway](#)

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

## PUT /sites/{site\_id}/gateways/{gateway\_id}/

### Update gateway (sitesSiteIdGatewaysGatewayIdPut)

Update the information for the specified gateway.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**gateway\_id (required)**

*Path Parameter* – UUID of the gateway.

#### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

#### Request body

**body [Gateway](#) (required)**

*Body Parameter* – Gateway object containing data about the gateway.

#### Return type

[Gateway](#)

#### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "last_sync_date" : "last_sync_date",
  "authorization_code" : "authorization_code",
  "serial_number" : "serial_number",
  "location" : "location",
  "model" : "model",
  "gateway_id" : "gateway_id"
}
```

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

#### Responses

**200**

Success [Gateway](#)

**400**

Bad Request - The supplied data is invalid.

**401**

Unauthorized - Invalid OAuth token or scope.

**404**

Not Found - The requested resource does not exist.



## GET /sites/{site\_id}/gateways/

### List gateways (sitesSiteIdGatewaysGet)

Get a list of the gateways for the specified site.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

#### Return type

array[[Gateway](#)]

#### Example data

Content-Type: application/json

```
[ {
  "note" : "note",
  "last_sync_date" : "last_sync_date",
  "authorization_code" : "authorization_code",
  "serial_number" : "serial_number",
  "location" : "location",
  "model" : "model",
  "gateway_id" : "gateway_id"
}, {
  "note" : "note",
  "last_sync_date" : "last_sync_date",
  "authorization_code" : "authorization_code",
  "serial_number" : "serial_number",
  "location" : "location",
  "model" : "model",
  "gateway_id" : "gateway_id"
} ]
```

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

#### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope



## POST /sites/{site\_id}/gateways/

### Add gateway (sitesSiteIdGatewaysPost)

Add a gateway to the specified site.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**Consumes**

This API call consumes the following media types via the Content-Type request header:

- application/json

**Request body**

body [Gateway](#) (required)

*Body Parameter* – Gateway object containing data about the gateway.

**Return type**

[Gateway](#)

**Example data**

Content-Type: application/json

```
{
  "note" : "note",
  "last_sync_date" : "last_sync_date",
  "authorization_code" : "authorization_code",
  "serial_number" : "serial_number",
  "location" : "location",
  "model" : "model",
  "gateway_id" : "gateway_id"
}
```

**Responses**

201

Created [Gateway](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

## Repeaters

[Up](#)

GET /sites/{site\_id}/repeaters/

List repeaters ([sitesSiteIdRepeatersGet](#))

Get a list of the repeaters for the specified site.

**Path parameters**

**site\_id (required)**

*Path Parameter* – UUID of the site.

**Return type**

array[[Repeater](#)]

## Example data

Content-Type: application/json

```
[ {
  "note" : "note",
  "repeater_id" : "repeater_id",
  "txid" : 0,
  "location" : "location",
  "model" : "model"
}, {
  "note" : "note",
  "repeater_id" : "repeater_id",
  "txid" : 0,
  "location" : "location",
  "model" : "model"
} ]
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

## Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope



POST /sites/{site\_id}/repeaters/

Add repeater (**sitesSiteIdRepeatersPost**)

Add a repeater to the specified site.

## Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

## Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

## Request body

body [Repeater](#) (required)

*Body Parameter* – Repeater object containing data about the repeater.

## Return type

[Repeater](#)

## Example data

Content-Type: application/json

```
{
  "note" : "note",
  "repeater_id" : "repeater_id",
  "txid" : 0,
  "location" : "location",
  "model" : "model"
}
```

### Responses

201

Created [Repeater](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

[Up](#)

DELETE /sites/{site\_id}/repeaters/{repeater\_id}/

Delete repeater (**sitesSiteIdRepeatersRepeaterIdDelete**)

Remove the specified repeater.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**repeater\_id (required)**

*Path Parameter* – UUID of the repeater.

### Responses

204

No Content - Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

[Up](#)

GET /sites/{site\_id}/repeaters/{repeater\_id}/

Get repeater (**sitesSiteIdRepeatersRepeaterIdGet**)

Get the information for the specified repeater.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**repeater\_id (required)**

*Path Parameter* – UUID of the repeater.

## Return type

[Repeater](#)

## Example data

Content-Type: application/json

```
{
  "note" : "note",
  "repeater_id" : "repeater_id",
  "txid" : 0,
  "location" : "location",
  "model" : "model"
}
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

## Responses

200

Success [Repeater](#)

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

[Up](#)

PUT /sites/{site\_id}/repeaters/{repeater\_id}/

Update repeater (**sitesSiteIdRepeatersRepeaterIdPut**)

Update the information for the specified repeater.

## Path parameters

**site\_id** (required)

*Path Parameter* – UUID of the site.

**repeater\_id** (required)

*Path Parameter* – UUID of the repeater.

## Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

## Request body

**body** [Repeater](#) (required)

*Body Parameter* – Repeater object containing data about the repeater.

## Return type

[Repeater](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "repeater_id" : "repeater_id",
  "txid" : 0,
  "location" : "location",
  "model" : "model"
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [Repeater](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## Transmitters



GET /sites/{site\_id}/transmitters/

List transmitters ([sitesSiteIdTransmittersGet](#))

Get a list of the transmitters for the specified site.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Return type

array[[Transmitter](#)]

### Example data

Content-Type: application/json

```
[ {
  "note" : "note",
  "txid" : 0,
  "location" : "location",
  "model" : "model",
  "transmitter_id" : "transmitter_id"
}, {
```



```
"note" : "note",
"txid" : 0,
"location" : "location",
"model" : "model",
"transmitter_id" : "transmitter_id"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

[Up](#)

POST /sites/{site\_id}/transmitters/

Add transmitter (**sitesSiteIdTransmittersPost**)

Add a transmitter to the specified site.

### Path parameters

**site\_id** (required)

*Path Parameter* – UUID of the site.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [Transmitter](#) (required)

*Body Parameter* – Transmitter object containing data about the transmitter.

### Return type

[Transmitter](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "txid" : 0,
  "location" : "location",
  "model" : "model",
  "transmitter_id" : "transmitter_id"
}
```

### Responses

**201**Created [Transmitter](#)**400**

Bad Request - The supplied data is invalid.

**401**

Unauthorized - Invalid OAuth token or scope.

DELETE /sites/{site\_id}/transmitters/{transmitter\_id}/ [Up](#)

Delete transmitter (sitesSiteIdTransmittersTransmitterIdDelete)

Remove the specified transmitter.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**transmitter\_id (required)**

*Path Parameter* – UUID of the transmitter.

### Responses

**204**

No Content - Success

**401**

Unauthorized - Invalid OAuth token or scope.

**404**

Not Found - The requested resource does not exist.

GET /sites/{site\_id}/transmitters/{transmitter\_id}/ [Up](#)

Get transmitter (sitesSiteIdTransmittersTransmitterIdGet)

Get the information for the specified transmitter.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**transmitter\_id (required)**

*Path Parameter* – UUID of the transmitter.

### Return type

[Transmitter](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "txid" : 0,
  "location" : "location",
  "model" : "model",
```

```
"transmitter_id" : "transmitter_id"
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [Transmitter](#)

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

[Up](#)

PUT /sites/{site\_id}/transmitters/{transmitter\_id}/

Update transmitter (sitesSiteIdTransmittersTransmitterIdPut)

Update the information for the specified transmitter.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**transmitter\_id (required)**

*Path Parameter* – UUID of the transmitter.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

**body [Transmitter](#) (required)**

*Body Parameter* – Transmitter object containing data about the transmitter.

### Return type

[Transmitter](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "txid" : 0,
  "location" : "location",
  "model" : "model",
  "transmitter_id" : "transmitter_id"
}
```

## Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

## Responses

200

Success [Transmitter](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## Models

[ [Jump to Methods](#) ]

### Table of Contents

1. [EnumGatewayModel](#) -
2. [EnumRepeaterModel](#) -
3. [EnumTransmitterModel](#) -
4. [Gateway](#) -
5. [Repeater](#) -
6. [Transmitter](#) -

#### EnumGatewayModel -

[Up](#)

id (optional)

[String](#) Identifier of gateway model entry. This will be read-only.

name (optional)

[String](#) Display name of gateway model entry. This will be read-only.

#### EnumRepeaterModel -

[Up](#)

id (optional)

[String](#) Identifier of repeater model entry. This will be read-only.

name (optional)

[String](#) Display name of repeater model entry. This will be read-only.

#### EnumTransmitterModel -

[Up](#)

id (optional)

[String](#) Identifier of transmitter model entry. This will be read-only.

name (optional)

[String](#) Display name of transmitter model entry. This will be read-only.

## Gateway -

### gateway\_id (optional)

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

### serial\_number (optional)

[String](#) Maximum 127 characters.

### authorization\_code

[String](#) Maximum 127 characters.

### note (optional)

[String](#) Maximum 4096 characters.

### location (optional)

[String](#) Maximum 4096 characters.

### model

[String](#) Enum: Will hold the id string for the enum value defined in EnumGatewayModel.

### last\_sync\_date (optional)

[String](#) Last date and time at which Gateway synchronized with the cloud. Read only.

## Repeater -

### repeater\_id (optional)

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

### txid

[Integer](#) Transmitter ID. Must be unique Integer. format: int32

### note (optional)

[String](#) Maximum 4096 characters.

### location (optional)

[String](#) Maximum 4096 characters.

### model

[String](#) Enum: Will hold the id string for the enum value defined in EnumRepeaterModel.

## Transmitter -

### transmitter\_id (optional)

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

### txid

[Integer](#) Transmitter ID. Must be unique Integer. format: int32

### note (optional)

[String](#) Maximum 4096 characters.

### location (optional)

[String](#) Maximum 4096 characters.

### model

[String](#) Enum: Will hold the id string for the enum value defined in EnumTransmitterModel.

# Inovonics TapWatch API

API to allow applications, dataloggers, and users to push or pull data to/from the cloud.

More information: <https://helloverb.com>

Contact Info: [support@inovonics.com](mailto:support@inovonics.com)

Version: 1.0.0.11

BasePath: /v1/tapwatch

All rights reserved

<http://apache.org/licenses/LICENSE-2.0.html>

## Access

1. OAuth AuthorizationUrl:TokenUrl:<https://api.tapwatch.com/oauth/token/>
2. OAuth AuthorizationUrl:TokenUrl:<https://api.tapwatch.com/oauth/token/>

## Methods

[ Jump to [Models](#) ]

## Table of Contents

### [Buildings](#)

- [DELETE /sites/{site\\_id}/buildings/{building\\_id}/](#)
- [GET /sites/{site\\_id}/buildings/{building\\_id}/](#)
- [PUT /sites/{site\\_id}/buildings/{building\\_id}/](#)
- [GET /sites/{site\\_id}/buildings/{building\\_id}/units/](#)
- [DELETE /sites/{site\\_id}/buildings/{building\\_id}/units/{unit\\_id}/](#)
- [PUT /sites/{site\\_id}/buildings/{building\\_id}/units/{unit\\_id}/](#)
- [GET /sites/{site\\_id}/buildings/](#)
- [POST /sites/{site\\_id}/buildings/](#)

### [Enums](#)

- [GET /enums/flowunit/](#)
- [GET /enums/metermodel/](#)
- [GET /enums/outfiletype/](#)
- [GET /enums/utilitytype/](#)

### [Meters](#)

- [GET /sites/{site\\_id}/meters/](#)
- [DELETE /sites/{site\\_id}/meters/{meter\\_id}/](#)
- [GET /sites/{site\\_id}/meters/{meter\\_id}/flowcount/](#)
- [GET /sites/{site\\_id}/meters/{meter\\_id}/](#)
- [GET /sites/{site\\_id}/meters/{meter\\_id}/metercount/](#)
- [PUT /sites/{site\\_id}/meters/{meter\\_id}/](#)
- [GET /sites/{site\\_id}/meters/{meter\\_id}/transmitters/](#)
- [DELETE /sites/{site\\_id}/meters/{meter\\_id}/transmitters/{transmitter\\_id}/](#)
- [PUT /sites/{site\\_id}/meters/{meter\\_id}/transmitters/{transmitter\\_id}/](#)
- [POST /sites/{site\\_id}/meters/](#)

### [Outfiles](#)

- [GET /sites/{site\\_id}/outfiles/{outfile\\_type}/](#)

### [Sites](#)

- [GET /sites/{site\\_id}/export/](#)
- [PUT /sites/{site\\_id}/import/](#)

### [Units](#)

- [GET /sites/{site\\_id}/units/](#)
- [POST /sites/{site\\_id}/units/](#)
- [DELETE /sites/{site\\_id}/units/{unit\\_id}/](#)

- [GET /sites/{site\\_id}/units/{unit\\_id}/](#)
- [GET /sites/{site\\_id}/units/{unit\\_id}/meters/](#)
- [DELETE /sites/{site\\_id}/units/{unit\\_id}/meters/{meter\\_id}/](#)
- [PUT /sites/{site\\_id}/units/{unit\\_id}/meters/{meter\\_id}/](#)
- [PUT /sites/{site\\_id}/units/{unit\\_id}/](#)

## Buildings

DELETE /sites/{site\_id}/buildings/{building\_id}/

[Up](#)

Delete building (sitesSiteIdBuildingsBuildingIdDelete)

Remove the specified building.

### Path parameters

**site\_id** (required)

*Path Parameter* – UUID of the site.

**building\_id** (required)

*Path Parameter* – UUID of the building.

### Responses

204

No Content - Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

GET /sites/{site\_id}/buildings/{building\_id}/

[Up](#)

Get building (sitesSiteIdBuildingsBuildingIdGet)

Get the information for the specified building.

### Path parameters

**site\_id** (required)

*Path Parameter* – UUID of the site.

**building\_id** (required)

*Path Parameter* – UUID of the building.

### Return type

[Building](#)

### Example data

Content-Type: application/json

```
{
  "building_id" : "building_id",
  "note" : "note",
  "name" : "name"
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [Building](#)

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.



PUT /sites/{site\_id}/buildings/{building\_id}/

Update building (sitesSiteIdBuildingsBuildingIdPut)

Update the information for the specified building.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**building\_id (required)**

*Path Parameter* – UUID of the building.

#### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

#### Request body

body **Building** (required)

*Body Parameter* – Building object containing data about the building.

#### Return type

[Building](#)

#### Example data

Content-Type: application/json

```
{
  "building_id" : "building_id",
  "note" : "note",
  "name" : "name"
}
```

#### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

#### Responses

200

Success [Building](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.



GET /sites/{site\_id}/buildings/{building\_id}/units/

List building-units (sitesSiteIdBuildingsBuildingIdUnitsGet)

Get a list of the building-unit relationships for the specified building.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**building\_id (required)**

*Path Parameter* – UUID of the building.



**Return type**array[[Unit](#)]**Example data**

Content-Type: application/json

```
[ {
  "note" : "note",
  "name" : "name",
  "unit_id" : "unit_id"
}, {
  "note" : "note",
  "name" : "name",
  "unit_id" : "unit_id"
} ]
```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses****200**

Success

**401**

Unauthorized - Invalid OAuth token or scope

**404**

Not Found - The requested resource does not exist.

DELETE /sites/{site\_id}/buildings/{building\_id}/units/{unit\_id}/



Delete building-unit (**sitesSiteIdBuildingsBuildingIdUnitsUnitIdDelete**)

Remove the specified building-unit relationship.

**Path parameters****site\_id (required)***Path Parameter* – UUID of the site.**building\_id (required)***Path Parameter* – UUID of the building.**unit\_id (required)***Path Parameter* – UUID of the unit.**Responses****204**

No Content - Success

**401**

Unauthorized - Invalid OAuth token or scope.

**404**

Not Found - The requested resource does not exist.

PUT /sites/{site\_id}/buildings/{building\_id}/units/{unit\_id}/



Add building-unit (**sitesSiteIdBuildingsBuildingIdUnitsUnitIdPut**)

Add a building-unit relationship to the specified site.

**Path parameters****site\_id (required)***Path Parameter* – UUID of the site.**building\_id (required)***Path Parameter* – UUID of the building.**unit\_id (required)**

*Path Parameter* – UUID of the unit.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Responses

204

Created

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

GET /sites/{site\_id}/buildings/

[Up](#)

List buildings (**sitesSiteIdBuildingsGet**)

Get a list of the buildings for the specified site.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Return type

array[[Building](#)]

### Example data

Content-Type: application/json

```
[ {
  "building_id" : "building_id",
  "note" : "note",
  "name" : "name"
}, {
  "building_id" : "building_id",
  "note" : "note",
  "name" : "name"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

POST /sites/{site\_id}/buildings/

[Up](#)

Add building (**sitesSiteIdBuildingsPost**)

Add a building to the specified site.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body **Building** (required)

*Body Parameter* – Building object containing data about the building.

### Return type

[Building](#)

### Example data

Content-Type: application/json

```
{
  "building_id" : "building_id",
  "note" : "note",
  "name" : "name"
}
```

### Responses

201

Created [Building](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

## Enums

GET /enums/flowunit/

[Up](#)

Get flow unit enum (**enumsFlowunitGet**)

Get the list of flow unit definitions.

### Return type

array[[EnumFlowUnit](#)]

### Example data

Content-Type: application/json

```
[ {
  "name" : "name",
  "id" : "id"
}, {
  "name" : "name",
  "id" : "id"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

GET /enums/metermodel/

[Up](#)

Get meter model enum (**enumsMetermodelGet**)

Get the list of meter model definitions.

### Return type

array[[EnumMeterModel](#)]

### Example data

Content-Type: application/json

```
[ {
  "name" : "name",
  "id" : "id",
  "category" : "category",
  "count_factor" : 0.8008281904610115
}, {
  "name" : "name",
  "id" : "id",
  "category" : "category",
  "count_factor" : 0.8008281904610115
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

## GET /enums/outfiletype/

[Up](#)

Get outfile type enum ([enumsOutfiletypeGet](#))

Get the list of outfile type definitions.

### Return type

array[[EnumOutfileType](#)]

### Example data

Content-Type: application/json

```
[ {
  "extension" : "extension",
  "name" : "name",
  "id" : "id"
}, {
  "extension" : "extension",
  "name" : "name",
  "id" : "id"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

## GET /enums/utilitytype/

[Up](#)

**Get utility type enum (enumsUtilitytypeGet)**

Get the list of utility type definitions.

**Return type**

array[[EnumUtilityType](#)]

**Example data**

Content-Type: application/json

```
[ {
  "name" : "name",
  "id" : "id",
  "category" : "category"
}, {
  "name" : "name",
  "id" : "id",
  "category" : "category"
} ]
```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses**

200

Success

401

Unauthorized - Invalid OAuth token or scope

## Meters



GET /sites/{site\_id}/meters/

**List meters (sitesSiteIdMetersGet)**

Get a list of the meters for the specified site.

**Path parameters**

**site\_id (required)**

*Path Parameter* – UUID of the site.

**Return type**

array[[Meter](#)]

**Example data**

Content-Type: application/json

```
[ {
  "note" : "note",
  "initial_meter_count" : 0.8008281904610115,
  "utility_type" : "utility_type",
  "name" : "name",
  "location" : "location",
  "serial_number" : "serial_number",
  "model" : "model",
  "flow_unit" : "flow_unit",
  "meter_id" : "meter_id",
  "count_factor" : 6.027456183070403
}, {
  "note" : "note",
  "initial_meter_count" : 0.8008281904610115,
  "utility_type" : "utility_type",
  "name" : "name",
```

```

"location" : "location",
"serial_number" : "serial_number",
"model" : "model",
"flow_unit" : "flow_unit",
"meter_id" : "meter_id",
"count_factor" : 6.027456183070403
} ]

```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses**

200

Success

401

Unauthorized - Invalid OAuth token or scope

DELETE /sites/{site\_id}/meters/{meter\_id}/

[Up](#)

Delete meter (**sitesSiteIdMetersMeterIdDelete**)

Remove the specified meter.

**Path parameters****site\_id (required)**

*Path Parameter* – UUID of the site.

**meter\_id (required)**

*Path Parameter* – UUID of the meter.

**Responses**

204

No Content - Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

GET /sites/{site\_id}/meters/{meter\_id}/flowcount/

[Up](#)

Get the highest flow count in the specified period. Flow count is defined by number of 15 minute periods in past 24 hours when a flow was detected. (**sitesSiteIdMetersMeterIdFlowcountGet**)

Get the highest flow count for the specified meter seen in the specified period.

**Path parameters****site\_id (required)**

*Path Parameter* – UUID of the site.

**meter\_id (required)**

*Path Parameter* – UUID of the meter.

**Request headers****Return type**[FlowCount](#)**Example data**

Content-Type: application/json

```

{
  "flow_count" : 0.8008281904610115,

```

```

    "timestamp" : "timestamp"
  }

```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses**

200

Success [FlowCount](#)

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

[Up](#)

GET /sites/{site\_id}/meters/{meter\_id}/

Get meter ([sitesSiteIdMetersMeterIdGet](#))

Get the information for the specified meter.

**Path parameters**

**site\_id** (required)

*Path Parameter* – UUID of the site.

**meter\_id** (required)

*Path Parameter* – UUID of the meter.

**Return type**

[Meter](#)

**Example data**

Content-Type: application/json

```

{
  "note" : "note",
  "initial_meter_count" : 0.8008281904610115,
  "utility_type" : "utility_type",
  "name" : "name",
  "location" : "location",
  "serial_number" : "serial_number",
  "model" : "model",
  "flow_unit" : "flow_unit",
  "meter_id" : "meter_id",
  "count_factor" : 6.027456183070403
}

```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses**

200

Success [Meter](#)

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

[Up](#)

GET /sites/{site\_id}/meters/{meter\_id}/metercount/

Get meter count for the meter ([sitesSiteIdMetersMeterIdMetercountGet](#))

Get the meter count for the specified meter. Caller should check the change in meter count and multiply by meter count factor to get the flow.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**meter\_id (required)**

*Path Parameter* – UUID of the meter.

### Return type

[MeterCount](#)

### Example data

Content-Type: application/json

```
{
  "meter_count" : 0.8008281904610115,
  "timestamp" : "timestamp"
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [MeterCount](#)

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

PUT /sites/{site\_id}/meters/{meter\_id}/

[Up](#)

Update meter ([sitesSiteIdMetersMeterIdPut](#))

Update the information for the specified meter.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**meter\_id (required)**

*Path Parameter* – UUID of the meter.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [Meter](#) (required)

*Body Parameter* – Meter object containing data about the meter.

### Return type

[Meter](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "initial_meter_count" : 0.8008281904610115,
  "utility_type" : "utility_type",
}
```



```

    "name" : "name",
    "location" : "location",
    "serial_number" : "serial_number",
    "model" : "model",
    "flow_unit" : "flow_unit",
    "meter_id" : "meter_id",
    "count_factor" : 6.027456183070403
  }

```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses****200**Success [Meter](#)**400**

Bad Request - The supplied data is invalid.

**401**

Unauthorized - Invalid OAuth token or scope.

**404**

Not Found - The requested resource does not exist.

GET /sites/{site\_id}/meters/{meter\_id}/transmitters/

[Up](#)

List meter-transmitters ([sitesSiteIdMetersMeterIdTransmittersGet](#))

Get a list of the meter-transmitter relationships for the specified unit.

**Path parameters****site\_id (required)***Path Parameter* – UUID of the site.**meter\_id (required)***Path Parameter* – UUID of the meter.**Return type**array[[Transmitter](#)]**Example data**

Content-Type: application/json

```

[ {
  "note" : "note",
  "txid" : 0,
  "location" : "location",
  "model" : "model",
  "transmitter_id" : "transmitter_id"
}, {
  "note" : "note",
  "txid" : 0,
  "location" : "location",
  "model" : "model",
  "transmitter_id" : "transmitter_id"
} ]

```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

**Responses****200**

Success

**401**

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

DELETE /sites/{site\_id}/meters/{meter\_id}/transmitters/{transmitter\_id}/ [Up](#)

Delete meter-transmitter (sitesSiteIdMetersMeterIdTransmittersTransmitterIdDelete)

Remove the specified meter-transmitter relationship.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**meter\_id (required)**

*Path Parameter* – UUID of the meter.

**transmitter\_id (required)**

*Path Parameter* – UUID of the transmitter.

#### Responses

204

No Content - Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

PUT /sites/{site\_id}/meters/{meter\_id}/transmitters/{transmitter\_id}/ [Up](#)

Add meter-transmitter (sitesSiteIdMetersMeterIdTransmittersTransmitterIdPut)

Add a meter-transmitter relationship to the specified site.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**meter\_id (required)**

*Path Parameter* – UUID of the meter.

**transmitter\_id (required)**

*Path Parameter* – UUID of the transmitter.

#### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

#### Responses

204

Created

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

POST /sites/{site\_id}/meters/ [Up](#)

Add meter (sitesSiteIdMetersPost)

Add a meter to the specified site.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [Meter](#) (required)

*Body Parameter* – Meter object containing data about the meter.

### Return type

[Meter](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "initial_meter_count" : 0.8008281904610115,
  "utility_type" : "utility_type",
  "name" : "name",
  "location" : "location",
  "serial_number" : "serial_number",
  "model" : "model",
  "flow_unit" : "flow_unit",
  "meter_id" : "meter_id",
  "count_factor" : 6.027456183070403
}
```

### Responses

201

Created [Meter](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

## Outfiles

GET /sites/{site\_id}/outfiles/{outfile\_type}/



Generate an outfile ([sitesSiteIdOutfilesOutfileTypeGet](#))

Create an outfile of reads from an IP Gateway in one of the specified outfile types. If no header parameters are specified, an outfile for the most recent 24 hours will be generated. At most, two header parameters may be used to calculate the range of time.

### Path parameters

**site\_id** (required)

*Path Parameter* – UUID of the site.

**outfile\_type** (required)

*Path Parameter* – The format type of the outfile can be mod1, mod2, mod3, mod4, mod5, csv1, xml1, xml2, json.

### Request headers

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/plain
- application/csv
- application/xml
- application/json

## Responses

200

Success

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

## Sites

GET /sites/{site\_id}/export/

[Up](#)

Export site (**sitesSiteIdExportGet**)

Export all of the data for the specified site. NOTE: Use the xlsx format. The xlsxwriter python module should help.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/vnd.openxmlformats-officedocument.spreadsheetml.sheet

## Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

PUT /sites/{site\_id}/import/

[Up](#)

Import (**sitesSiteIdImportPut**)

Replace Buildings/Units/Meters/Transmitters/Repeaters for the Site. Import file is in xlsx format.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/vnd.openxmlformats-officedocument.spreadsheetml.sheet

### Request body

body **binary** (required)

*Body Parameter* – Buildings/Units/Meters/Transmitters/Repeaters for the Site. Import file is in xlsx format.

## Responses

204

No Content - Success

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## Units



GET /sites/{site\_id}/units/

List units (**sitesSiteIdUnitsGet**)

Get a list of the units for the specified site.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Return type

array[[Unit](#)]

### Example data

Content-Type: application/json

```
[ {
  "note" : "note",
  "name" : "name",
  "unit_id" : "unit_id"
}, {
  "note" : "note",
  "name" : "name",
  "unit_id" : "unit_id"
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope



POST /sites/{site\_id}/units/

Add unit (**sitesSiteIdUnitsPost**)

Add a unit to the specified site.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [Unit](#) (required)

*Body Parameter* – Unit object containing data about the unit.

### Return type

[Unit](#)

**Example data**

Content-Type: application/json

```
{
  "note" : "note",
  "name" : "name",
  "unit_id" : "unit_id"
}
```

**Responses**

201

Created [Unit](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

**DELETE** /sites/{site\_id}/units/{unit\_id}/Delete unit (**sitesSiteIdUnitsUnitIdDelete**)

Remove the specified unit.

**Path parameters****site\_id (required)***Path Parameter* – UUID of the site.**unit\_id (required)***Path Parameter* – UUID of the unit.**Responses**

204

No Content - Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

**GET** /sites/{site\_id}/units/{unit\_id}/Get unit (**sitesSiteIdUnitsUnitIdGet**)

Get the information for the specified unit.

**Path parameters****site\_id (required)***Path Parameter* – UUID of the site.**unit\_id (required)***Path Parameter* – UUID of the unit.**Return type**[Unit](#)**Example data**

Content-Type: application/json

```
{
  "note" : "note",
  "name" : "name",
  "unit_id" : "unit_id"
}
```

**Produces**

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [Unit](#)

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

GET /sites/{site\_id}/units/{unit\_id}/meters/



List unit-meters ([sitesSiteIdUnitsUnitIdMetersGet](#))

Get a list of the unit-meter relationships for the specified unit.

### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**unit\_id (required)**

*Path Parameter* – UUID of the unit.

### Return type

array[[Meter](#)]

### Example data

Content-Type: application/json

```
[ {
  "note" : "note",
  "initial_meter_count" : 0.8008281904610115,
  "utility_type" : "utility_type",
  "name" : "name",
  "location" : "location",
  "serial_number" : "serial_number",
  "model" : "model",
  "flow_unit" : "flow_unit",
  "meter_id" : "meter_id",
  "count_factor" : 6.027456183070403
}, {
  "note" : "note",
  "initial_meter_count" : 0.8008281904610115,
  "utility_type" : "utility_type",
  "name" : "name",
  "location" : "location",
  "serial_number" : "serial_number",
  "model" : "model",
  "flow_unit" : "flow_unit",
  "meter_id" : "meter_id",
  "count_factor" : 6.027456183070403
} ]
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success

401

Unauthorized - Invalid OAuth token or scope

404

Not Found - The requested resource does not exist.

DELETE /sites/{site\_id}/units/{unit\_id}/meters/{meter\_id}/

[Up](#)

Delete unit-meter (sitesSiteIdUnitsUnitIdMetersMeterIdDelete)

Remove the specified unit-meter relationship.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**unit\_id (required)**

*Path Parameter* – UUID of the unit.

**meter\_id (required)**

*Path Parameter* – UUID of the meter.

#### Responses

204

No Content - Success

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

PUT /sites/{site\_id}/units/{unit\_id}/meters/{meter\_id}/

[Up](#)

Add unit-meter (sitesSiteIdUnitsUnitIdMetersMeterIdPut)

Add a unit-meter relationship to the specified site.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**unit\_id (required)**

*Path Parameter* – UUID of the unit.

**meter\_id (required)**

*Path Parameter* – UUID of the meter.

#### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

#### Responses

204

Created

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

PUT /sites/{site\_id}/units/{unit\_id}/

[Up](#)

Update unit (sitesSiteIdUnitsUnitIdPut)

Update the information for the specified unit.

#### Path parameters

**site\_id (required)**

*Path Parameter* – UUID of the site.

**unit\_id (required)**



*Path Parameter* – UUID of the unit.

### Consumes

This API call consumes the following media types via the Content-Type request header:

- application/json

### Request body

body [Unit](#) (required)

*Body Parameter* – Unit object containing data about the unit.

### Return type

[Unit](#)

### Example data

Content-Type: application/json

```
{
  "note" : "note",
  "name" : "name",
  "unit_id" : "unit_id"
}
```

### Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

### Responses

200

Success [Unit](#)

400

Bad Request - The supplied data is invalid.

401

Unauthorized - Invalid OAuth token or scope.

404

Not Found - The requested resource does not exist.

## Models

[ [Jump to Methods](#) ]

### Table of Contents

1. [Building -](#)
2. [EnumFlowUnit -](#)
3. [EnumMeterModel -](#)
4. [EnumOutfileType -](#)
5. [EnumUtilityType -](#)
6. [FlowCount -](#)
7. [Meter -](#)
8. [MeterCount -](#)
9. [Repeater -](#)
10. [Transmitter -](#)
11. [Unit -](#)

#### Building -

**building\_id** (optional)

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

**name**

[String](#) Maximum of 127 characters.

**note** (optional)

[String](#) Maximum 4096 characters.

[Up](#)

**EnumFlowUnit -**[Up](#)**id (optional)**[String](#) Identifier of flow unit entry. This will be read-only.**name (optional)**[String](#) Name of flow unit entry. This will be read-only.**EnumMeterModel -**[Up](#)**id (optional)**[String](#) Identifier of meter model entry. This will be read-only.**name (optional)**[String](#) Display name of meter model entry. This will be read-only.**count\_factor (optional)**[Double](#) Count factor of meter model entry from factory. This will be read-only. format: double**category (optional)**[String](#) The generic utility category of the meter. Ex: Water**EnumOutfileType -**[Up](#)**id (optional)**[String](#) Identifier of outfile type entry. This will be read-only.**name (optional)**[String](#) Display name of outfile type entry. This will be read-only.**extension (optional)**[String](#) Display the extension of outfile type entry. This will be read-only**EnumUtilityType -**[Up](#)**id (optional)**[String](#) Identifier of utility type entry. This will be read-only.**name (optional)**[String](#) Display name of utility type entry. This will be read-only.**category (optional)**[String](#) The generic utility category of the utility type. Ex: Water**FlowCount -**[Up](#)**flow\_count (optional)**[Double](#) The number of 15 minute slots in the 24 hours period before timestamp, where a flow was detected. Is null if no flow was reported. format: double**timestamp (optional)**[String](#) ISO8601 datetime, time at which the flow count was reported. Is null if no flow was reported.**Meter -**[Up](#)**meter\_id (optional)**[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.**name (optional)**[String](#) Maximum of 127 characters.**note (optional)**[String](#) Maximum 4096 characters.**location (optional)**[String](#) Maximum 4096 characters.**initial\_meter\_count (optional)**[Double](#) The initial reading from the meter face. Defaults to zero. format: double

**count\_factor**

[Double](#) The number of units measured per pulse of the meter. format: double

**serial\_number (optional)**

[String](#) Maximum 127 characters.

**model**

[String](#) Enum: Will hold the id string for the enum value defined in EnumMeterModel.

**utility\_type**

[String](#) Enum: Will hold the id string for the enum value defined in EnumUtilityType.

**flow\_unit (optional)**

[String](#) Enum: Will hold the id string for the enum value defined in EnumFlowUnit.

**MeterCount -**[Up](#)**meter\_count (optional)**

[Double](#) The most recent meter count reported. Is null if no meter count was reported. format: double

**timestamp (optional)**

[String](#) ISO8601 datetime, time at which the meter count was reported. Is null if no meter count was reported.

**Repeater -**[Up](#)**repeater\_id (optional)**

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

**txid**

[Integer](#) Transmitter ID. Must be unique Integer. format: int32

**note (optional)**

[String](#) Maximum 4096 characters.

**location (optional)**

[String](#) Maximum 4096 characters.

**model**

[String](#) Enum: Will hold the id string for the enum value defined in EnumRepeaterModel.

**Transmitter -**[Up](#)**transmitter\_id (optional)**

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

**txid**

[Integer](#) Transmitter ID. Must be unique Integer. format: int32

**note (optional)**

[String](#) Maximum 4096 characters.

**location (optional)**

[String](#) Maximum 4096 characters.

**model**

[String](#) Enum: Will hold the id string for the enum value defined in EnumTransmitterModel.

**Unit -**[Up](#)**unit\_id (optional)**

[String](#) Must be unique. UUID. Will be randomly generated. This will be read-only.

**name**

[String](#) Maximum of 127 characters.

**note (optional)**

[String](#) Maximum 4096 characters.