M212133EN-E

API Reference

Vaisala Beacon Cloud XML API

Beacon Cloud

VAISALA

PUBLISHED BY

Vaisala Oyj
Vanha Nurmijärventie 21, FI-01670 Vantaa, Finland
P.O. Box 26, FI-00421 Helsinki, Finland
+358 9 8949 1

Visit our Internet pages at www.vaisala.com.

© Vaisala 2021

Permission is hereby granted, free of charge, to any person obtaining a copy of this document (the "Document"), to deal in the Document without restriction, including without limitation the rights to use, copy, modify, merge, publish and distribute copies of the Document, and to permit persons to whom the Document is furnished to do so, subject to the following conditions:

This permission notice shall be included in all copies or substantial portions of the Document.

THE DOCUMENT IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE DOCUMENT OR THE USE OR OTHER DEALINGS IN THE DOCUMENT.

Table of contents

1. About this document

1.1 Version information

This document provides a description of Vaisala Beacon Cloud XML API data reporting interface.

Table 1 Document versions (English)

Document code	Date	Description
M212133EN-E	April 2021	• Updated examples • Moved Wx Beacon specific content to document M212639EN
M212133EN-D	March 2021	• Added air quality parameters • Updated request and response data and all examples • Updated document layout
M212133EN-C	December 2020	–

1.2 Documentation conventions

WARNING! Warning alerts you to a serious hazard. If you do not read and follow instructions carefully at this point, there is a risk of injury or even death.

CAUTION! Caution warns you of a potential hazard. If you do not read and follow instructions carefully at this point, the product could be damaged or important data could be lost.

Note highlights important information on using the product.

Tip gives information for using the product more efficiently.

Lists tools needed to perform the task.

Indicates that you need to take some notes during the task.

1.3 Trademarks

Vaisala® is a registered trademark of Vaisala Oyj.

Python® is a trademark or registered trademark of the Python Software Foundation.

All other product or company names that may be mentioned in this publication are trade names, trademarks, or registered trademarks of their respective owners.

2. XML API endpoint

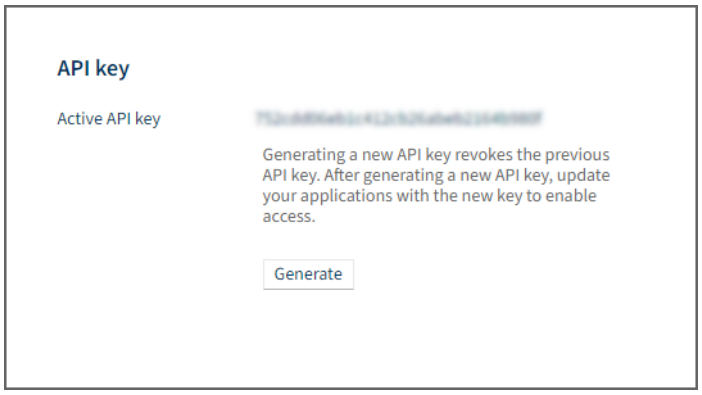
The XML API interface endpoint is:

- Beacon Cloud XML API endpoint: https://beacon.vaisala.com/api/

3. Enabling API from Beacon Cloud UI

To use the XML API, an API key needs to be generated from the Beacon Cloud application.

1. In Beacon Cloud, go to **Device settings** and in **API key**, select **Generate**.



2. On that same page, check the device serial number to be used in the query.

4. HTTP request and response

The XML API supports only HTTP GET. There is also a limitation that a user can fetch a maximum of 100 000 measurements with a single request. If more requests are needed, then subsequent requests need to be made.

The following is a cURL example of making a request with GET:

```
curl https://beacon.vaisala.com/api?
d=R1234567&k=752cdd06eb1c412cb26abeb2164b980f&t0=2020-12-29T00:
00:00&t1=2020-12-29T00:01:00&c=100
```

Table 2 Request parameters

Name	Value	Example	Description
d	Serial number	R1234567	Serial number of the device.
k	API key	752cdd06ed1c412cb26abeb2164b980f	Device specific API key. Needs to be generated from Beacon Cloud application to be able to use the API.
t0	Start timestamp in UTC time	2020-12-29T00:00:00	Start timestamp in ISO 8601 format. Time must be UTC time.
t1	Stop timestamp in UTC time	2020-12-29T12:00:00	End timestamp in ISO 8601 format. Time must be UTC time.
c	Max count	1000	Parameter to restrict the amount of measurements retrieved. Optional parameter.

Example response

```
<?xml version="1.0" encoding="UTF-8"?>
<envitemsdata>
    <device>
        <name>Laaksolahti</name>
        <serial>R1234567</serial>
        <type>5</type>
        <description>AQT420: R1234567</description>
        <location></location>
        <lat>0</lat>
        <lon>0</lon>
        <alt>0</alt>
    </device>
    <measurements>
        <meas id="0">
            <timestamp>2021-04-21 11:20:22</timestamp>
            <type>Air Temp.</type>
            <value>11.7</value>
        </meas>
        <meas id="1">
            <timestamp>2021-04-21 11:20:22</timestamp>
            <type>Air Hum.</type>
            <value>41.2</value>
        </meas>
        <meas id="2">
            <timestamp>2021-04-21 11:20:22</timestamp>
            <type>Air Pres./type>
            <value>992</value>
        </meas>
    </measurements>
</envitemsdata>
```

If authentication fails, the XML API returns the following:

```
<error>Access denied!</error>
```

4.1 HTTP status codes

The XML API returns the following HTTP status codes.

HTTP status code	Status	Description
200	Success	Successful request

4.2 Response schema

The XML API returns an XML message that is based on the following schema.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
 <xs:element name="envitemsdata">
  <xs:complexType>
   <xs:sequence>
    <xs:element name="device">
     <xs:complexType>
      <xs:choice maxOccurs="unbounded" minOccurs="0">
       <xs:element type="xs:string" name="name"/>
       <xs:element type="xs:string" name="serial"/>
       <xs:element type="xs:byte" name="type"/>
       <xs:element type="xs:string" name="description"/>
       <xs:element type="xs:string" name="location"/>
       <xs:element type="xs:string" name="lat"/>
       <xs:element type="xs:string" name="lon"/>
       <xs:element type="xs:string" name="alt"/>
      </xs:choice>
     </xs:complexType>
    </xs:element>
    <xs:element name="measurements">
     <xs:complexType>
      <xs:sequence>
       <xs:element name="meas" maxOccurs="unbounded" minOccurs="0">
        <xs:complexType>
         <xs:sequence>
          <xs:element type="xs:string" name="timestamp"/>
          <xs:element type="xs:string" name="type"/>
          <xs:element type="xs:float" name="value"/>
         </xs:sequence>
         <xs:attribute type="xs:byte" name="id" use="optional"/>
        </xs:complexType>
       </xs:element>
      </xs:sequence>
     </xs:complexType>
    </xs:element>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
</xs:schema>
```

5. Measurement parameters and units

Here is a list of all possible parameter types from the XML API and their fixed units.

Parameter	Unit	Unit description
Air Temp.	°C	Degrees Celsius
Air Pres.	hPa	Hectopascal
Air Hum.	%RH	Percentage 0 ... 100 %
NO_2	ppb	Parts per billion
NO	ppb	Parts per billion
O_3	ppb	Parts per billion
CO	ppb	Parts per billion
SO_2	ppb	Parts per billion
H_2S	ppb	Parts per billion
$PM_{2.5}$	$\mu g/m^3$	Micrograms per cubic meter
PM_{10}	$\mu g/m^3$	Micrograms per cubic meter

6. Request example with Python 3

The following example retrieves data from the XML API and prints the response.

xml-api-request.py file

```
#!/usr/bin/env python
import requests
import pandas as pd
import xml.etree.ElementTree as et
from datetime import datetime, timedelta
from collections import defaultdict

def wxBeaconApiRequest(startUTC, endUTC, serial, apikey):
    endpoint = 'https://beacon.vaisala.com/api/'
    query = '?d='+serial+'&k='+apikey+'&t0='+startUTC+'&t1='+endUTC
    URL = endpoint + query

    observations = et.fromstring(requests.get(URL).content)[1]
    d = defaultdict(dict)

    for obs in observations:
        timestamp = obs.find("timestamp").text
        type = obs.find("type").text
        value = obs.find("value").text
        d[timestamp][type] = float(value)

    df = pd.DataFrame.from_dict(d, orient='index') # convert index to datetime
objects
    df.index = pd.to_datetime(df.index)
    return df

# get last 24h data
utcNow = datetime.utcnow()
startTimeUTC = (utcNow-timedelta(days=1)).strftime('%Y-%m-%dT%H:%M:%S')
endTimeUTC = utcNow.strftime('%Y-%m-%dT%H:%M:%S')
serial = 'T05207'20
apikey = 'ee6fa67fd2ee42e6b3fa68286e94af44'

df = wxBeaconApiRequest(startTimeUTC, endTimeUTC, serial, apikey)

print(df)
```

Example when run in command line interface:

```
PS C:\Users\jyveh\proj\tools\data_analysis_tools\python> python .\xml-api-
test.py
                      NO2    CO     O3     NO PM2.5  PM10 Air Temp. Air Hum. Air Pres.
2021-04-21 09:44:05  0.006 0.008  0.005 -0.002   6.8  41.9      11.4     43.7      993.1
2021-04-21 09:45:09  0.005 0.012  0.008 -0.003   6.8  41.9      11.4     43.7      993.1
2021-04-21 09:46:05  0.003 0.015  0.010 -0.001   6.8  41.9      11.3     44.0      993.1
2021-04-21 09:47:06  0.002 0.019  0.012 -0.003   6.8  41.9      11.2     44.2      993.0
2021-04-21 09:48:07  0.001 0.024  0.013 -0.001   6.8  41.9      11.2     44.7      993.0
                       ...   ...    ...    ...   ...   ...       ...      ...        ...
2021-04-21 11:35:06 -0.002 0.073 -0.002  0.009   5.7  27.3      11.6     42.1      991.8
2021-04-21 11:36:06 -0.002 0.072 -0.002  0.009   5.7  27.3      11.8     41.4      991.7
2021-04-21 11:37:07 -0.001 0.073 -0.001  0.007   5.7  27.3      11.9     40.9      991.7
2021-04-21 11:38:05 -0.001 0.072 -0.001  0.007   5.7  27.3      12.2     40.8      991.7
2021-04-21 11:39:06 -0.001 0.071 -0.001  0.006   5.7  27.3      12.4     40.3      991.7


[116 rows x 9 columns]
```

Warranty

For standard warranty terms and conditions, see www.vaisala.com/warranty.

Please observe that any such warranty may not be valid in case of damage due to normal wear and tear, exceptional operating conditions, negligent handling or installation, or unauthorized modifications. Please see the applicable supply contract or Conditions of Sale for details of the warranty for each product.

Technical support

Contact Vaisala technical support at helpdesk@vaisala.com. Provide at least the following supporting information as applicable:

- Product name, model, and serial number
- Software/Firmware version
- Name and location of the installation site
- Name and contact information of a technical person who can provide further information on the problem

For more information, see www.vaisala.com/support.

VAISALA

www.vaisala.com