

Munisense import

Available since 2019.02.

Data is imported from <https://opendata.munisense.net>. Example query:

https://opendata.munisense.net/api/v2/wareco-water2/groundwaterwells/1082172/water_level_filtered/query/presets/last_month

Example data set:

```
{
  "results": [
    {
      "timestamp": "2019-03-21T10:15:23.000+00:00",
      "first_timestamp": 1553164216716,
      "last_timestamp": 1553164995278,
      "avg_manual_validation_state": 1,
      "min_manual_validation_state": 1,
      "max_manual_validation_state": 1,
      "count": 16,
      "avg_value": 8.677,
      "min_value": 8.389,
      "max_value": 8.698
    },
    {
      "timestamp": "2019-03-21T11:15:23.000+00:00",
      "first_timestamp": 1553168033620,
      "last_timestamp": 1553168033620,
      "avg_manual_validation_state": 1,
      "min_manual_validation_state": 1,
      "max_manual_validation_state": 1,
      "count": 1,
      "avg_value": 8.7,
      "min_value": 8.7,
      "max_value": 8.7
    },
    {
      "timestamp": "2019-03-21T12:15:23.000+00:00",
      "first_timestamp": 1553171633620,
      "last_timestamp": 1553171633620,
      "avg_manual_validation_state": 1,
      "min_manual_validation_state": 1,
      "max_manual_validation_state": 1,
      "count": 1,
      "avg_value": 8.698,
      "min_value": 8.698,
      "max_value": 8.698
    },
    {
      "timestamp": "2019-03-21T13:15:23.000+00:00",
      "first_timestamp": 1553175233620,
      "last_timestamp": 1553175233620,
      "avg_manual_validation_state": 1,
      "min_manual_validation_state": 1,
      "max_manual_validation_state": 1,
      "count": 1,
      "avg_value": 8.7,
      "min_value": 8.7,
      "max_value": 8.7
    }
  ],
  "meta": {}
}
```

From the data set "avg_value" is imported.

Important notice:

The timestamp is used for parsing the time in the time series. However, the returned Json contains time steps that are not exactly round hour, such as: "11:15:23.000+00:00". How much it differs from the exact hour changes regularly. Time series sets should be configured either as non-equidistant (which will import data at the exact time the timestep shows) or the need tolerance configured, which will round the time to the closest exact time step. If the difference between the timestamp and the closest time step is bigger than the tolerance, the data will not be imported.

Data can be imported from different time periods. The longer the time period, the larger the data aggregation period. Available choices are:

last_day=sampled by minute
last_week=sampled by 10 minutes
last_month=sampled by hour
last_year=sampled by 8 hour
last_three_years= sampled by 24 hour

The name of the period needed should be configured as external qualifierId of the time series. It should be configured either in the import module configuration or an id mapping, but not both.

Configuration example:

```

<?xml version="1.0" encoding="UTF-8"?>
<timeSeriesImportRun xmlns="http://www.wldelft.nl/fews"
                      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                      xsi:schemaLocation="http://www.wldelft.nl/fews http://fews.wldelft.nl
/schemas/version1.0/timeSeriesImportRun.xsd">
  <import>
    <general>
      <importType>MunisenseImport</importType>
      <serverUrl>https://opendata.munisense.net/api/v2/wareco-water2/groundwaterwells<
/serverUrl>

      <user>dummy_username</user>
      <password>dummy_password</password>
      <relativeViewPeriod unit="day" start="-10" end="8" startOverrutable="true"
endOverrutable="true"/>
      <idMapId>MunisenseImportIdMapper</idMapId>
      <importTimeZone>
        <timeZoneOffset>+00:00</timeZoneOffset>
      </importTimeZone>
    </general>
    <tolerance parameterId="Tide" timeUnit="minute" unitCount="29"/>
    <timeSeriesSet>
      <moduleInstanceId>MunisenseImport</moduleInstanceId>
      <valueType>scalar</valueType>
      <parameterId>Tide</parameterId>
      <!-- Do not configure a qualifierId here if you are using id-mapping -->
      <!-- <qualifierId>last_year</qualifierId> -->
      <locationId>LocA</locationId>
      <timeSeriesType>external historical</timeSeriesType>
      <timeStep unit="hour" multiplier="1"/>
      <readWriteMode>add originals</readWriteMode>
    </timeSeriesSet>
  </import>
</timeSeriesImportRun>

```

ID mapping example:

```

<?xml version="1.0" encoding="UTF-8"?>
<idMap version="1.1" xmlns="http://www.wldelft.nl/fews" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.wldelft.nl/fews http://fews.wldelft.nl/schemas/version1.0/idMap.xsd">
  <map externalLocation="1082172" externalParameter="water_level_filtered" internalLocation="LocA"
internalParameter="Tide" externalQualifier="last_month" />
</idMap>

```

The external location ID should be the same as the id in the URL (1082172 in the case of the example.)

External parameter id should be the same as in the url (water_level_filtered in this case.).