

Traffic light data API User's Guide

API Description

The API provides real sensor data from monitored traffic light and detector equipment in Tampere and in Oulu. Currently, only several junctions are supported, but all the junctions are going to provide data soon.

The API divides traffic light data into two categories:

- meta, which describes the configuration data for a certain junction with traffic lights
- data, which contains the actual monitored traffic light data

Meta is updated more infrequently, whereas data is updated more frequently. Recommended polling frequencies for a client are:

- meta: once a day
- data: once a second

The API endpoints provide data as JSON formatted strings over HTTP GET (please see API Endpoints below for endpoint URLs). Also PDF files are provided which present the physical characteristics of junctions, such as locations of various sensors inside the junction.

API Endpoints

<http://data.itsfactory.fi/trafficlights/meta/tampere> allows the client to obtain meta from the city of Tampere.

<http://data.itsfactory.fi/trafficlights/data/tampere> allows the client to obtain data from the city of Tampere.

<http://data.itsfactory.fi/trafficlights/meta/oulu> allows the client to obtain meta from the city of Oulu.

<http://data.itsfactory.fi/trafficlights/data/oulu> allows the client to obtain data from the city of Oulu.

Queries and Responses

Response

The response has a list of data elements. One data element contains the response data about one junction at some time point. Depending on the query, the response returns metadata or real-time data from Tampere or Oulu junctions.

Queries

Meta

A client can access the latest configuration information about Tampere junctions by issuing request at <http://data.itsfactory.fi/trafficlights/meta/tampere>

A client can access the latest configuration information about Oulu junctions by issuing request at <http://data.itsfactory.fi/trafficlights/meta/oulu>

This request will produce the following response:

```
{
  "Meta" : [
    {
      "location" : "TRE306",
      "packageTime" : "2015-10-01T12:41:51Z",
      "signals" : [
        {
          "index" : "0",
          "name" : "A"
        },
        {
          "index" : "1",
          "name" : "B"
        }
      ],
      ...
    ],
    "detectors" : [
      {
        "index" : "0",
        "name" : "A40_1",
        "signalIndex" : "0",
        "threshold" : "40"
      },
      {
        "index" : "1",
        "name" : "A40_2",
        "signalIndex" : "0",
        "threshold" : "40"
      }
    ]
  }
}
```

```

        ...
    ]
},
...
]
}

```

The response contains a list of elements where each element corresponds to one metadata package received from one junction. Following fields are present:

- location specifies the unique name of the junction
- packageTime is a combined date and time in UTC expressed according to ISO 8601 in the format “YYYY-MM-DDThh:mm:ssZ”. It specifies the point of time when the metadata package was received from the provider. The frequency of updating metadata is not defined, but generally a new package is received after some configuration changes were done at the specific junction
- signals is a list of elements where each element corresponds to one specific traffic light signal group. Each element contains the following fields:
 - index specifies the ordering number of the signal group. The index is referenced by the order of the characters in the field “signalStates” in Data entity (see later)
 - name specifies the identifier of the signal group. This identifier enables one to find a signal group’s location on the map of the junction
- detectors is a list of elements where each element corresponds to one specific detector. Each element contains the following fields:
- index specifies the ordering number of the detector. The index is referenced by the order of the characters in the fields “detectorStates” and “detectorLevels” in Data entity (see later)
 - name specifies the identifier of the detector. This identifier enables one to find a detector’s location on the map of the junction
 - signalIndex corresponds to the “index” of the signal group (see above) which regulates the traffic on the lane where the detector is installed
 - threshold specifies the sensitivity level that must be exceeded before counting and reporting the detection of a vehicle

Note! A detector name helps one to identify its type and understand its operation.

- Inductive loops are the most useful detectors that consist of insulated loop wires under the surface of lanes. They usually have a name starting with one letter and continuing with few numbers (e.g. “A40_1”, “D1_1_1”, etc.) and also have a threshold equal or greater than 30.
- Bus priority detectors have a prefix “VP”. They are in action when they receive a request via GPS from a delayed bus.
- Congestion detectors have the prefix “Ruuhka”. They are congestion monitoring detectors, which count the number of 5 second occupancy periods per collection session. The counter value is increased by one when the detector has been occupied for 5 seconds, and it is reset after that.

The response contains a list of elements where each element corresponds to one data package received from one junction. Following fields are present:

- time is a combined date and time in UTC expressed according to ISO 8601 in the format “YYYY-MM-DDThh:mm:ssZ”. It specifies the point of time when the traffic light signals and detectors activity was monitored
- location specifies the unique name of the junction
- sequenceNumber specifies the sequence number of the data package received from the provider.
- tickCount specifies a number of milliseconds multiplied by 100 that passed from the time stamp in “time” value. **Note!** The data might come unsorted by tickCount, it is a responsibility of a user to sort the rows.
- timestamp specifies the time stamp when traffic light signal and detectors activity was monitored. It is calculated by adding the “tickCount” to the “time” value.
- signalStates specifies traffic light signal states in the specific junction at a certain time stamp. The length of the “signalStates” string is the number of traffic light signal groups at the junction. Every character in the string defines the status of a single traffic light (green, red or amber phase). The order of the characters corresponds to the signal indexes defined in the Meta entity. The defined possible states are:
 - o '0' - RED_AMBER,
 - o '1' - MINIMUM_GREEN,
 - o '3' - VAR_MINIMUM_GREEN,
 - o '4' - PASSIVE_GREEN,
 - o '5' - GREEN_EXTENSION,
 - o '6' - GREEN_EXTENSION_LC,
 - o '7' - FIXED_PAST_END_GREEN,
 - o '8' - VA_PAST_END_GREEN,
 - o '9' - RED_SYNCH,
 - o ':' - GREEN_BLINKING,
 - o ';' - AMBER_FLASHING,
 - o '<' - FIXED_AMBER,
 - o '=' - AMBER_DARK,
 - o '>' - VA_AMBER,
 - o '?' - VA_MINIMUM_RED,
 - o '@' - RED_CLEARANCE,
 - o 'A' - MINIMUM_RED,
 - o 'B' - PASSIVE_RED,
 - o 'C' - RED_REQUEST,
 - o 'D' - RED_PRIORITY,
 - o 'E' - RED_PRIVILEGE,
 - o 'F' - RED_WAIT,
 - o 'G' - RED_STOP,
 - o 'H' - INTERGREEN,
 - o 'I' - FIXED_AMBER_START,

- o 'J' - FIXED_RED_START.
All other states are undefined
- detectorStates shows detectors' states at the intersection at a certain moment. The order of the characters corresponds to the detector indexes defined in the Meta entity. The value can be either 1 or 0 while 1 means occupancy of a detector.
- detectorLevels presents a list of integers corresponding to the occupancy level of each detector. If a detector is occupied, its sensitivity value exceeds the threshold value which is defined in the Meta entity.

Junction Description files

You can find the relevant PDF files of the junctions in:

<http://data.itsfactory.fi/files/trafficlights/>

Additional information about the junctions in Tampere is available in the catalog of the city of Tampere:

- traffic signal detectors

<http://palvelut2.tampere.fi/tietovaranto/tietovaranto.php?id=52&alasivu=1&vapaasana>

- traffic signal devices

<http://palvelut2.tampere.fi/tietovaranto/tietovaranto.php?id=51&alasivu=1&vapaasana>

- intersections

<http://palvelut2.tampere.fi/tietovaranto/tietovaranto.php?id=50&alasivu=1&vapaasana>

Existing applications:

- Tampere Liki 3D <http://liki3d.herokuapp.com/junction>
- Patterns <http://geodata.link>

License

The city of Tampere open data license:

<http://www.tampere.fi/tampereen-kaupunki/tietoa-tampereesta/avoin-data/avoin-data-lisenssi.html>