

API - Frequently asked questions

- [Guides](#)
- [Questions and answers](#)
 - [How can I get information about the structural metadata?](#)
 - [Does the API provide versioned information?](#)
 - [Why does SDMX API reject my data query with 400 error code](#)
 - [Why does TSV data file contains extra whitespace ?](#)



This section will be enriched on base of received questions, to be submitted through the [Eurostat user support](#).

Guides

Some information would not fit in the short Q&A section below, a dedicated page is proposed for the following topics:

- [API - FAQ - TSV data format](#)
- [API - FAQ - Usage in Excel Power Query](#)

Questions and answers

How can I get information about the structural metadata?

The API provides every available information about the structural metadata. More information is available on [this page](#).

Does the API provide versioned information?

The API provides:

Following structural artefacts are versioned and final following SDMX specification: code lists, concept schemes and data structure definitions: each version of each item is available and can be downloaded via the API.

For statistical dataset, corresponding SDMX artefacts are non-final and always with version=1.0.

There is no history available on data.

Why does SDMX API reject my data query with 400 error code

HTTP 400 Bad Request is a generic error status to inform that the requested URL could not be processed by the server. It can happens in several cases

Case 1 : Invalid request

Network appliance receiving HTTP requests may reject them with the following information

Request Error (invalid_request)

Your request could not be processed. Request could not be handled
This could be caused by a misconfiguration, or possibly a malformed request.

Transaction ID: 81f26aca0ed8b6a6-000000000c55bc8d-0000000064b76cba

For assistance, contact your network support team.

Main case is when a URL would exceed the character size limit supported by the network appliance.

A simple example is when building an explicit query for a NUTS3 regional datas for API statistics, the URL would contains a lot of **&geo=** parameters



In most case, it would be recommended to leave such dimension unfiltered in the query URL and filter the received data or use the dedicated geoLevel parameter in this particular example:

```
https://ec.europa.eu/eurostat/api/dissemination/statistics/1.0/data/DEMO_R_GIND3?
format=JSON&lang=en&freq=A&indic_de=JAN&time=2018&time=2019&time=2020&time=2021&time=2022&geoLevel
=nuts3
```

```
https://ec.europa.eu/eurostat/api/dissemination/statistics/1.0/data/DEMO_R_GIND3?
format=JSON&lang=en&freq=A&indic_de=JAN&time=2018&time=2019&time=2020&time=2021&time=2022
```

Case 2: Invalid query

SDMX API REST URLs are validated that they contain dimension codes and position codes actually declared in the dataset definition. In other words validation ensures that the codes present in the query are present in the SDMX Constraint of the dataset.

Example REST URL: https://ec.europa.eu/eurostat/api/dissemination/sdmx/2.1/data/ISOC_CI_ID_H/A.H_IPC.PC_HH.TOTAL.EU27

This URL would not pass validation and an HTTP 400 Bad Request response would be issued with following content

```
<S:Fault xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <faultcode>150</faultcode>
  <faultstring>INVALID_QUERY_DIMENSION_VALUE: Query is invalid as per its structure's definition.
    The following values for dimension are not allowed: GEO=EU27.</faultstring>
</S:Fault>
```

This error message is reporting that **EU27** cannot be found in the list of position actually present in the GEO dimension of the dataset.

The valid code to use in this case is **EU27_2020**

The SDMX Constraint can be retrieved to allow client side query validation. It can be retrieved from the SDMX API as follow :

https://ec.europa.eu/eurostat/api/dissemination/sdmx/2.1/contentconstraint/ESTAT/ISOC_CI_ID_H

Why does TSV data file contains extra whitespace ?

As dataset may contains attributes flags attached to an observation value, the TSV format is adding them next to the value separated by.

In case of data results without such flags, these extra spaces could look like as unnecessary but they always be present to ensure format consistency.

In below example, space are visible as a dot and tabulation as an arrow.

```
freq,unit,s_adj,nace_r2,indic,geo\TIME_PERIOD→1980-01.→1980-02.→1980-03.→1980-04.→
M,I2015,CA,B,IS-IP,IE→59.0.→72.4.→74.2.→73.4.→116.0.→101.1.→94.2.→81.2.→81.7.→
M,I2015,CA,B,IS-IP,NL→239.1.p>203.1.p>207.3.p>161.9.p>127.4.p>112.0.p>106.2.p>82.4.p→102.6.p>
```

The space is present in all data columns to ensure generic processing, for example separating by both tabulation and whitespace will result in having the attributes in an extra column next to the value.

Please consult the format description page for more details : [API - FAQ - TSV data format](#)