
ILOSTAT SDMX API

Data Dissemination Web Service Reference Guide

Author:

Department of **STATISTICS**

Publication Date:

2020-05-01

Table of Contents

I	I Introduction	2
I.1	Document tracking.....	2
II	II Project description	4
III	III Data and metadata dissemination	4
III.1	Description	4
III.1.1	Overview.....	4
III.1.2	URL format.....	4
III.2	Structural metadata	7
III.2.1	Common query string parameters.....	7
III.2.2	Agency scheme	7
III.2.3	Concept scheme	7
III.2.4	Code lists.....	8
III.2.5	Data structure definitions.....	9
III.2.6	Data flows.....	10
III.2.6.1	Annotations for Default Presentation.....	11
III.2.7	Category Scheme and Categorization.....	11
III.2.8	Content Constraint.....	12
III.3	XML Validation Schema.....	12
III.4	Data and metadata dissemination.....	12
III.4.1	Selection of a dataflow.....	12
III.4.2	Filtering.....	13
III.4.3	Query string parameters	14
III.4.4	Attributes and descriptive metadata	15
III.4.5	Range specification and total observation values count	15
IV	IV Annex I: SDMX 2.1 RESTful web services cheat sheet (v1.4.0)	17

I Introduction

I.1 Document tracking

Version	Date	Author	Comment
3.1	2018-11-09	I. Pogor	<ul style="list-style-type: none"> * New Base URLs using HTTPS * New default standard is Generic v2.1 * CSV has been added as a new format for data messages * COUNTRY (CL_COUNTRY) attribute has been replaced by REF_AREA (CL_AREA) * Annotations are now identified by "Type" instead of "Title" * SORT annotation has been changed to ORDER * CATEGORYSCHEME, AGENCY SCHEME, CATEGORISATION, CONTENT CONSTRAINT have been added as new resources in structural metadata * Section IV "Data Collection" has been removed from this "Data Dissemination" Guide
3.2	2019-07-10	E. Greising, W. Lei, I. Pogor	<ul style="list-style-type: none"> * JSON added as a valid format for structural metadata messages * Country groups code extension for data queries (X##_COU) * Annotations for default presentations * Total values count and range specification
3.3	2020-05-01	E. Greising, W. Lei, I. Pogor	<ul style="list-style-type: none"> * SURVEY dimension removed * Descriptive metadata (notes) coded attributes transformed into four free text attributes containing Currency and Source,

			<p>Indicator and Classification level notes.</p> <ul style="list-style-type: none">* Fixed detail=serieskeyonly delivery* firstNObservations temporary disabled* lastNObservations=1 only implemented* Support for detail=referencepartial query parameter* New test entry-point
--	--	--	--

II Project description

III Data and metadata dissemination

III.1 Description

III.1.1 Overview

The dissemination service is a partial implementation of the standard SDMX RESTful API, as defined in the [SDMX Standards: Section 7 – Guidelines for the use of Web services](#). This API defines a set of HyperText Transfer Protocol Secure (HTTPS) resources which give access to:

- Structural metadata (code lists, concepts, definition of data structures, etc.)
- Data and informational metadata (values, notes)

III.1.2 URL format

The defined resources are all accessible through HTTPS GET requests, using a Web browser or any HTTPS-enabled software component. Each resource is associated to a Uniform Resource Locator (URL), as summarized in the picture next page.

The SDMX connector is currently deployed under the following Base URLs¹:

- Development/Integration testing: <https://www.ilo.org/sdmx-test/rest>

¹ The Old Base URLs (<http://www.ilo.org/ilostat/sdmx/ws/rest> and <http://www.ilo.org/ilostat/sdmx-test/ws/rest> for production and development respectively) still give access to the previous version's web service. These services will be decommissioned in **June 2020** and its structure is not maintained in accordance with the latest standards. Users are urged to transition to the new Web Service as per the new guidelines.

- Production/public service: <https://www.ilo.org/sdmx/rest>

Structural metadata queries:

.../resource/agencyID/resourceID/version/itemID?queryStringParameters

Resource: The type of metadata to be returned. Valid values: datastructure, conceptscheme, codelist, dataflow, categoryscheme, categorisation, agencyscheme, contentconstraint.

Not implemented queries: metadatastructure, hierarchicalcodelist, organisationscheme, organisationunitscheme, process, reportingtaxonomy, structure, dataproviderscheme, dataconsumerscheme, provisionagreement, structureset, attachmentconstraint, metadataflow.

agencyID: Agency maintaining the artefact (e.g.: ILO) Default: ILO

resourceID: Artefact ID (e.g.: CL_FREQ) Default: all

version: Artefact version (e.g.: 1.0) Default: latest

itemID: ID of the Item to be returned (e.g.: Q). Valid for Item schemes only. Default: all

queryStringParameters:

detail: Desired amount of information to be returned.

Valid values: allstubs, referencestubs, referencepartial, full.
Default: full

References: References to be returned with the artefact.

Valid values: none, children, descendants, parents, parentsandsiblings, all, or any type of resource (e.g. dataflow).

Default: none

Data queries:

.../resource/flowRef/key/providerRef?queryStringParameters

Resource: Type of query. Valid value: data. Not implemented: metadata

flowRef: Dataflow ref (e.g. ILO,DF_YI_ALL_EMP_TEMP_SEX_AGE_NB,latest)

key: Key of the series to be returned (e.g: KI.FRA.2854.A.EAP_DWAF_RT). Wildcarding (e.g: STI.ESP..Q) and ORing (e.g: YI.KHI+JPN..A) is supported.

providerRef: Data provider (e.g.: ILO) Default: all

queryStringParameters:

startPeriod: Start period (inclusive). ISO8601 (e.g. 2014-01) or SDMX reporting period (e.g. 2014-Q3)

endPeriod: End period (inclusive). ISO8601 (e.g. 2014-01-01) or SDMX reporting period (e.g. 2014-M10).

If dates are written in ISO8601 format (with dashes, i.e. YYYY-MM-DD), every point-in-time which reference period is entirely comprised between *startPeriod* and *endPeriod* (inclusive) will be included in the query response.

<code>firstNObservations</code> :	Maximum number of observations starting from the first observation
<code>lastNObservations</code> :	Maximum number of observations counting back from the most recent observation
<code>dimensionAtObservation</code> :	Id for the dimension attached at the observation level. Default: <code>TIME_PERIOD</code>
<code>detail</code> :	Desired amount of information to be returned. Valid values: <code>serieskeyonly</code> , <code>dataonly</code> , <code>nodata</code> , <code>full</code> . Default: <code>full</code>
<code>updatedAfter</code> :	Last time the data was updated in the database. Used to retrieve deltas. Must be percent-encoded

Not implemented:

<code>includeHistory</code> :	Whether to return vintages. Default: <code>false</code>
-------------------------------	---

Message format:

Following the specification of the SDMX 2.1 RESTful Web Service, the message format should be specified through the *Accept* HTTP header.

Valid formats in this implementations are:

Structural metadata queries:

- *SDMX-ML Structure* `application/vnd.sdmx.structure+xml;version=2.1` (valid only for structural metadata queries)
- *SDMX-JSON Structure* `application/vnd.sdmx.structure+json;version=1.0` (valid only for structural metadata queries)

Data queries:

- *SDMX-ML Generic Data* `application/vnd.sdmx.genericdata+xml;version=2.x`
- *SDMX-ML Structure Specific Data* `application/vnd.sdmx.structurespecificdata+xml;version=2.1` (compact)
- *SDMX-JSON Data* `application/vnd.sdmx.data+json;version=1.0.0-wd`
- *SDMX-CSV Data* `application/vnd.sdmx.data+csv;version=1.0.0`

For *Generic Data* message, «version=» parameter can be 2.0 or 2.1.

The `format` and `formatVersion` parameter, an NSIWS extension to the standard URL, overrides the *Accept* header value and allows selecting the message format and version from the URL.

The valid format options are:

- `genericdata` will deliver a message in *Generic Time Series Data* format *Version 2.0* or *2.1* (the default format and version)

- `structurespecificdata` will deliver a message in *Structure Specific Time Series Data* format (compact) *Version 2.1*
- `jsondata` will deliver a message in *json* format (valid only for data messages)
- `csv` will deliver a message in *csv* format (valid only for data messages).
- `structure` will deliver a message in *Structure Time Series Data* format

`formatVersion` parameter can only be specified together with `format` in order to modify the default version.

In case no Accept value is provided and neither format nor version parameter is specified in the URL, the default message format is *SDMX-ML Generic Version 2.1*. If `formatVersion` parameter is omitted, the latest format version of the requested format is delivered.

Annotations:

Many annotations at different attachment levels are included in this API, most of them being agreed globally like “ORDER”. Annotations are identified by the AnnotationType value and may include information in the AnnotationText (localized content) or AnnotationTitle (non localized) depending on the usage.

III.2 Structural metadata

III.2.1 Common query string parameters

Each URL is composed of a mandatory path and an optional query string, respectively the parts of the URL before and after the question mark (“?”) sign. The parameters allowed in the query string of a structure query are defined at paragraph 4.3.2.2 of the SDMX standard, section 7.

In the ILO implementation, the detail parameter is not mandatory, defaulting to full.

III.2.2 Agency scheme

The `agency` resource gives access to the agency scheme. The ILO implementation currently defines only the ILO agency. The `identifier` and `version` variables are ignored, the agency scheme is thus accessible under `/agency/ILO`. Since in this implementation the agency always defaults to “ILO”, the same results are obtained with `/agency`.

III.2.3 Concept scheme

The `concept` resource gives access to the concept scheme. The ILO implementation defines the following concept schemes:

- `CS_ILOSTAT`: defines the concepts common to the whole ILOSTAT information model and that are not included in the following two concept schemes.
- `CS_CLASSIF_TYPE`: defines the concepts used as classifications (indicator-bound dimensions) in the ILOSTAT information model

- `CS_NOTE_TYPE`: defines the concepts used as notes (descriptive metadata presented as Table-level or value-level attributes) in the ILOSTAT information model

The `version` variable is ignored (defaulting to `latest` or `1.0`).

III.2.4 Code lists

The `codelist` resource gives access to the code lists used to represent the coded dimensions and attributes (i.e. classifications and coded notes). All the code lists include `Code`, `Name` and `Description` elements (when available) and some additional fields in the form of annotations that provide useful information for dissemination and/or presentation purposes. (e.g. “ORDER” gives an order of prevalence of the item)

The ILO implementation defines the following code lists:

- Classification code lists:
 - One general list per classification type, accessible under `/codelist/ILO/CL_<CLY_CODE>`. It includes the description of the type and codes and descriptions for all the classifications items of all classifications versions of this type. (e.g. `.../codelist/ILO/CL_ECO`). This code lists include a virtual classification item code of the form `_z` which description is “Not applicable” to be assigned to all classifications not belonging to a certain indicator in a Dataflow based in the generic DSD. (See III.2.6 below)
 - One per classification version, accessible under `/codelist/ILO/CL_<CLV_CODE>`. It includes codes and descriptions for the classifications items of this version. (e.g. `.../codelist/ILO/CL_ECO_ISIC4`)

Classifications’ code lists include the following annotations:

- `ORDER`: The sort/prevalence order.
- `Is_Total (Y/N)`: Indicates if the classification item is an aggregate.
- For the different variants (versions) of classifications, the following codelists are also available:
 - The resource `/codelist/ILO/CL_CLASSIF_<CLY_CODE>` will deliver a code list which items will be all classifications versions of this type (e.g. `.../codelist/ILO/CL_CLASSIF_ECO`).
 - `ORDER`: The sort/prevalence order.
 - `Display_Mode`: Indicates the suggested presentation for this note type, whether it is as `CODE` (codes only), `LABEL` (labels only) or `CODE_LABEL` (both).
- Reference area code list: Accessible through `/codelist/ILO/CL_AREA`, it provides the country code according to the ISO-3166 ALPHA3 standard and the name in the three official languages. The preferable sort order in each language is given as the annotation `ORDER`. The annotation `Group` is set to “Y” if this item represents a region. All the codes for items representing groups of countries are defined in the open space “X__” of the ISO-3166 standard using numbers. X01 has been assigned to “World”, and numbers have been assigned to code Income level categories as defined by the World Bank and ILO Geographical regions Africa, Americas, Arab States, Asia and the Pacific and Europe, and its sub-divisions,

respectively. `ISO2Code` and `M49Code` are provided as annotations for all the countries as per the ISO-3166 standard.

- Common dimensions and attributes: for the representation of indicator-invariant dimensions (frequency, collection, represented variable, etc.) and attributes (time format, value status, etc.). To get a full list of all available code lists, you can use <https://www.ilo.org/sdmx/rest/codelist?detail=allstubs>

The `version` variable is ignored (defaulting to `latest` or `1.0`). If the `identifier` variable is not specified, all existing code lists in the information model will be returned - i.e. `/codelist/ILO/`

To get all the code lists related to an indicator definition use the DSD query with the `references=children` Or `references=descendants` parameter. (See III.2.5)

III.2.5 Data structure definitions

The `datastructure` resource gives access to the Data Structure Definitions (DSD) which describes the data flows used to collect or disseminate data. In the ILO implementation, attributes within the same DSD are used to convey descriptive metadata (notes).

For example, for ILOSTAT Decent Work Indicators, each DSD is identified by the `{collection; country; indicator}` triplet, where:

- `collection` stands for the business code of a collection (ref. `CL_COLLECTION` codelist)
- `country` stands for the ISO 3166-1 alpha-3 code of a country (ref. `CL_AREA` codelist), or `ALL` for the generic, dissemination DSD valid for all countries for one specific indicator.
- `indicator` stands for the business code of an indicator (ref. `CL_INDICATOR` codelist). The keyword `ALL` is allowed to generate a generic DSD for the country with all the dimensions collapsed into a single hypercube, including all classifications used by every indicator. Including the `TOPIC` code instead of the indicator, a DSD for the Topic will be generated by collapsing in a single hypercube all the valid combinations of dimensions and attributes according to the indicators defined in ILOSTAT. Similarly, the `REPRESENTED_VARIABLE` code can be used to obtain the DSD for this Represented Variable.

The resource `/datastructure/ILO/<COLLECTION_CODE>_ALL_ALL?references=children` can be used to get all code lists in one single query.

- The keyword `MULTI` generates a multi-indicator collection DSD for a given country. It can also be combined with the “Topic-DSD” and “Represented Variable-DSD” features. This is useful for querying individual table structures per indicator.

Under this scheme, the following resources are valid:

- `/datastructure/ILO/YI_MEX_EMP_TEMP_SEX_AGE_NB`: country specific DSD, for the yearly collection of the “Employment by Sex and Age” indicator for Mexico
- `/datastructure/ILO/YI_MEX_EMP_TEMP_SEX_AGE_NB?references=descendants`: same DSD, with full references to the code lists and concepts embedded.
- `/datastructure/ILO/YI_ALL_UNE_MULTI?references=descendants`: a single xml message with multiple DSDs corresponding to all the indicators in the topic UNE, including all referenced code lists and concepts.

- /datastructure/ILO/YI_ALL_EMP_TEMP_SEX_ECO_NB: generic DSD, for the dissemination of the “Employment by Sex and Economic activity” indicator
- /datastructure/ILO/YI_MEX_ALL: country specific generic DSD for Mexico, including all indicators in the Yearly Indicators (YI) collection (around 200 as of writing) collapsed in a single hypercube of 31 dimensions: Collection, Reference Area, Frequency, Represented Variable, 26 classification breakdowns and Time.
- /datastructure/ILO/YI_MEX_MULTI: country specific multiple DSD for Mexico, including all indicators in the Yearly Indicators (YI) collection (around 200 as of writing) as independent Key families into a single xml file.

III.2.6 Data flows

The `dataflow` resource gives access to the data flows used to collect or disseminate data. In the ILO implementation, one `dataflow` is defined for each DSD, identified by `DF_<DSD_ID>`. The default behaviour is the same as when querying DSDs, either for one indicator, grouping by Topic or Represented variable or using the keywords “ALL” or “MULTI”. (See III.2.5)

Under this scheme, the query `.../dataflow/ILO` will return a message including all the dissemination `dataflows` (i.e. indicators) with data available, so it is the best way of discovering data availability. For simplification, the `dataflows` included in the response to this query are those representing indicators and the generics by collection (`<col>_ALL_ALL`).

Similarly to the `datastructure` resource query, querying a `dataflow` based in a generic ALL indicators’ flow reference like in the following example:

```
.../dataflow/ILO/DF_YI_MEX_ALL/<filters>?<parameters>
```

will return a single `dataflow` corresponding to the country specific generic DSD for Mexico, including all indicators in the Yearly Indicators (YI) collection (200+ as of writing) collapsed in a single hypercube.

On the other hand, the query `.../dataflow/ILO/DF_YI_MEX_MULTI/<filters>?<parameters>` will generate a message including all the valid DF for Mexico corresponding to indicators in the YI collection with data available. In other words, the API’s behaviour is the same as specifying “MULTI” for DSDs, but constraining the result by data availability, since the main intention of the API is data dissemination.

“Individual” `dataflows` can also be queried; the following URL gives access to the `dataflow` for yearly indicators, France, Employment by Sex and Age:

```
.../dataflow/ILO/DF_YI_FRA_EMP_TEMP_SEX_AGE_NB/<filters>?<parameters>.
```

As explained for DSDs above, under this logic, the following resources are valid `dataflows`:

- /dataflow/ILO/DF_YI_MEX_EMP_TEMP_SEX_AGE_NB: country specific, for the yearly collection of the “Employment by Sex and Age” indicator for Mexico
- /dataflow/ILO/DF_YI_MEX_EMP_TEMP_SEX_AGE_NB?references=descendants: same `dataflow`, with full references to the code lists, concepts and data structures embedded.
- /dataflow/ILO/DF_YI_ALL_EES?references=descendants: an xml message with a `dataflow` making use of the generic DSD for the all the indicators in the topic EES, including all referenced code lists, concepts and `datastructure`.

- `/dataflow/ILO/DF_YI_ALL_EMP_MULTI?references=descendants`: a single xml message with multiple dataflows corresponding to all the indicators in the topic EMP, including all referenced code lists, concepts and respective datastructures.
- `/dataflow/ILO/DF_YI_ALL_EES_XTMP_SEX_RT`: generic dataflow, for the dissemination of the “Share of temporary employees by sex” indicator
- `/dataflow/ILO/DF_YI_MEX_ALL`: country specific generic dataflow for Mexico, collapsing all indicators in the Yearly Indicators (YI) collection in a single hypercube.
- `/dataflow/ILO/DF_YI_MEX_MULTI`: country specific multiple dataflow for Mexico, including all indicators in the Yearly Indicators (YI) collection with data available.

III.2.6.1 Annotations for Default Presentation

In some cases, for the purpose of data and metadata presentation, the following annotations have been introduced:

- `LAYOUT_ROW` and `LAYOUT_COLUMN`: present the specified dimensions in rows or columns.
- `ORDER`: suggested sort order of the artefacts in the presentation.
- `NAME_COMPLEMENT`: introduced to all items in NOTE codelist. This annotation contains the note type name to be displayed in the presentation.
- `FULL_NAME`: introduced to all items in NOTE codelist. This annotation contains the `{notetype name}: {note value}`.
- `NOT_DISPLAYED`: can be used to hide dimension(s) and/or attribute(s) in the presentation.
- `DEFAULT`: annotation defines a set of filters so that a pre-defined default subset of data can be retrieved instead of the entire dataset.

III.2.7 Category Scheme and Categorization

The `categoryscheme` resource gives access to the different criteria defined to access SDMX artefacts (mainly data flows) used to disseminate data. In the ILO implementation, dataflows represent indicator tables, identified by `DF_<DSD_ID>`.

Under this scheme, the query `.../categoryscheme/ILO` will return a message including all the categories (i.e. access criteria), so it is the best way of start the navigation of ILOSTAT data.

A query for a category scheme with `references=categorisation` will return all artefacts categorized by the categories in the `categoryscheme`, e.g.:

```
/categoryscheme/ILO/CAS_SUBJECT?references=categorisation
```

will return all dataflows categorized by each category in the `CAS_SUBJECT` `categoryscheme` (i.e.subjects)

If one particular category is specified, the artefacts categorized by this category are returned, e.g.:

```
/categoryscheme/ILO/CAS_SUBJECT/latest/EMP?references=categorisation
```

will return all dataflows of the subject Employment (*this feature is not implemented in the NSIWS but will be available soon*).

For simplification, the dataflows included in the response to this query are those representing the generics by collection/indicators, not by country (<col>_ALL_<ind>).

III.2.8 Content Constraint

As mentioned in III.2.6, data flows used to disseminate data in the ILO implementation are in such a way that one `dataflow` is defined for each DSD, identified by `DF_<DSD_ID>`. Since all dissemination artefacts are generated “on the fly” according to the query requirements, item schemes’ content can be “tailored” by construction to include just the relevant items for the context by means of “virtual content constraints” which can be applied when requested through a `detail=referencepartial` query parameter. For example, for a given `dataflow` including referenced descendants artefacts, `codelists` will include only the relevant code items for this context, and will be flagged as `isPartial="true"`. This approach makes it not necessary to “materialize” the content constraints, thus making the artefacts smaller and facilitating its treatment. Nevertheless, in the current version, the queries issued with `detail=full` will deliver the full `codelists` but lack the content constraint.

III.3 XML Validation Schema

The `schema` URL space gives access to XML Schema files that can be used to validate data flows in SDMX-ML files.

III.4 Data and metadata dissemination

The `data` URL space gives access to the disseminated data and metadata. The URL scheme allows:

- The selection of a given `dataflow`
- The optional filtering of data/metadata across dimensions
- Optional query string parameters

The `format` parameter allows selecting the message format requested. The valid options are:

- `genericdata` will deliver a message in *Generic Time Series Data format Version 2.0* (`formatVersion=2.0`) or *2.1* (`formatVersion=2.1`, the default format and version)
- `structurespecificdata` will deliver a message in *Structure Specific Time Series Data format Version 2.1*
- `jsondata` will deliver a message in *json* format (valid only for data messages)
- `csv` will deliver a message in *csv* format (valid only for data messages).

III.4.1 Selection of a dataflow

The `dataflow` variable of the URL scheme is of the form `<agency_id>,<dataflow_id>,<version>`. In the ILO dissemination API, `agency_id` is always `ILO` and `version` can be omitted.

So in order to get data from the yearly collection of the Employment by Sex and Age indicator for all countries, the `dataflow` identifier is: `ILO,DF_YI_ALL_EMP_TEMP_SEX_AGE_NB`.

It should be noted that, when querying for data using the “generic” dataflows (i.e. those with “ALL” in the indicator portion of the id, or referring to a topic or represented variable), data is formatted according to the generic hypercube including all the breakdowns in the indicators referenced and thus, certain “key” combinations may be invalid (do not exist in any indicator). All the classification dimensions of the generic hypercube not corresponding to a certain key family (i.e. a valid indicator) will be filled with the global “Not applicable” code `_z` (See III.2.4 above). Please take into account that this type of queries in a big collection like “YI” or especially “STP” can be very time consuming.

III.4.2 Filtering

The next component of the URL is called the key in the SDMX specification. It is a dot-separated list of values for the dimensions, in the order defined by the DSD related to the dataflow. An omitted dimension is wild carded; the logical “or” operator is supported using the “+” character.

The ALL keyword wildcards all dimensions, allowing to get all the data for a given DSD.

Following on the previous example, the DSD for Employment by Sex and Age² defines the following dimensions, most general to most specific: COLLECTION, REF_AREA, FREQ, REPRESENTED_VARIABLE, and afterwards the classifications involved in the indicator like CLASSIF_SEX, CLASSIF_AGE, etc. in the order given in the DSD.

Any filter can be specified either explicitly (by providing a value or expression) or omitted by just adding the separator dot to its right. The characters “M”, “Q” and “A” in the FREQ dimension can be used to filter data points with Monthly, Quarterly and Annual time reference periods.

Examples:

- `/data/ILO,DF_YI_ALL_EMP_TEMP_SEX_AGE_NB/ALL`: fetch all data for the yearly collection
- `/data/ILO,DF_YI_ALL_EMP_TEMP_SEX_AGE_NB/.....`: idem (all dimensions wildcarded)
- `/data/ILO,DF_YI_ALL_EMP_TEMP_SEX_AGE_NB/.MEX+ESP.....`: fetch data for México and Spain, with all other dimensions wildcarded.
- `/data/ILO,DF_YI_ALL_EMP_TEMP_SEX_AGE_NB/....SEX_T.AGE_5YRBANDS_TOTAL`: filter on the “total” classification items (same information as in the multi-country report of the ILOSTAT website)
- `/data/ILO,DF_STI_ALL_EMP_TEMP_SEX_AGE_NB/.DEU+FRA+GBR+ITA.M...`: fetch monthly data only for Germany, France, the United Kingdom and Italy.
- `/data/ILO,DF_YI_ALL_EMP_TEMP_SEX_AGE_NB/YI.FRA.A..SEX_F.AGE_5YRBANDS_15-19+AGE_5YRBANDS_60-64`: fetch data for France, for female workers, ages 15-19 and 60-64.
- `/data/ILO,DF_YI_ESP_UNE_TUNE_SEX_AGE_NB/....AGE_5YRBANDS_Y00-04+AGE_5YRBANDS_Y05-09+AGE_5YRBANDS_Y10-14+AGE_5YRBANDS_Y15-19+AGE_5YRBANDS_Y20-24+AGE_5YRBANDS_Y25-29+AGE_5YRBANDS_Y30-34+AGE_5YRBANDS_Y35-39+AGE_5YRBANDS_Y40-44+AGE_5YRBANDS_Y45-49+AGE_5YRBANDS_Y50-54+AGE_5YRBANDS_Y55-59+AGE_5YRBANDS_Y60-64+AGE_5YRBANDS_YGE65+AGE_5YRBANDS_TOTAL`: fetch data for Spain for 5-year age bands classification version.

²An Indicator is defined as a Represented variable broken-down by none, one or more classifications. The Represented variable code is composed by the Topic (i.e. EMP), the variant (i.e. TEMP for Total EMPLOYMENT) and the Type of measure (i.e. NB for Number, RT for Rate, etc.). The indicator code is obtained by inserting the Classification Types codes (i.e. SEX & AGE) after the variant code: EMP_TEMP_SEX_AGE_NB.

- `/data/ILO,DF_KI_ARG_EES/...EES_LT35_RT+EES_3548_RT+EES_TG48_RT..`: fetch data for multiple indicators of the topic EES in a single query.

An “ILO extension” has been implemented to make it easier to query data for country groups³. Any country group is, at the same time, a reference to all the countries that compound it and an entity itself which holds indicators’ values (estimates). Thus, “X85” means at the same time the BRICS as an entity and Brazil (BRA), Russia (RUS), India (IND), China (CHN) and South Africa (ZAF) individually. To allow querying for both the indicator’s values for BRICS as a whole and for the countries in the BRICS, the following must be used:

- X85 means BRICS
- X85_COU means BRA+RUS+IND+CHN+ZAF

Examples:

- `/data/ILO,DF_ILOEST_ALL_EAP_2EAP_SEX_AGE_NB/.X85....`: fetch data for BRICS as a whole (only valid for estimates)
- `/data/ILO,DF_ILOEST_ALL_EAP_2EAP_SEX_AGE_NB/.X85_COU....`: fetch data for Brazil (BRA), Russia (RUS), India (IND), China (CHN) and South Africa (ZAF) individually
- `/data/ILO,DF_ILOEST_ALL_EAP_2EAP_SEX_AGE_NB/. BRA+RUS+IND+CHN+ZAF....`: same as above

III.4.3 Query string parameters

The standard SDMX REST interface defines a number of query string parameters that can be used to further tailor the data flow retrieved. These parameters are to be specified after a “?” and concatenated with “&”. The ILO implementation interprets a reduced set of these parameters, as described below:

- `startPeriod`: the starting date for the period for which results should be supplied (inclusive)
- `endPeriod`: the ending date for the period for which results should be supplied (inclusive)

Accepted formats: ISO8601 (e.g. 2014-01) or SDMX reporting period (e.g. 2014-Q3)

If dates are written in ISO8601 format (with dashes, i.e. YYYY-MM-DD), every point-in-time which reference period is entirely comprised between `startPeriod` and `endPeriod` (inclusive) will be included in the query response.

- `firstNObservations`: includes data for the first N time points⁴.
- `lastNObservations`: includes data for the last N time points⁵.
- `detail`: the desired amount of information to be returned; must be one of:
 - `full`: default value, return everything (data, annotations, attributes)

³ At the ILO, several groups of countries are represented in the CL_AREA code list by extension ISO3 codes in the format “X###”. There are two main criteria for grouping: geographical regions and income level. Besides, some well-known groups are also represented, like G7, BRICS, G20, ASEAN, etc.

⁴ In the current version, `firstNObservations` parameter has been temporary disabled due to performance issues.

⁵ In the current implementation, only `lastNObservations=1` is supported

- `dataonly`: values only (no attributes)
- `serieskeyonly`: can be used for performance reasons to fetch the series (with specified dimensions) matching a given query
- `nodata`: this parameter value delivers only attributes and annotations.
- `format`: The message format. Valid options are:
 - `genericdata` will deliver a message in *Generic Time Series Data format Version 2.0* (`formatVersion=2.0`) or *2.1* (`formatVersion=2.1`, the default format and version)
 - `structurespecificdata` will deliver a message in *Structure Specific Time Series Data format Version 2.1*
 - `jsondata` will deliver a message in *json* format (valid only for data messages)
 - `csv` will deliver a message in *csv* format (valid only for data messages).

So, going further on the previous examples, the following request is possible: `/data/ILO,DF_YI_ALL_EMP_TEMP_SEX_AGE_NB/YI.FRA.A..SEX_T.AGE_5YRBANDS_TOTAL?startPeriod=2009-01-01&endPeriod=2009-12-31&detail=dataonly&format=jsondata`: fetch the data point for France, total classifications, year 2009 in a json data message.

III.4.4 Attributes and descriptive metadata

For each observation value there will be a set of attributes describing the data point. These attributes include certain characteristics of the measured dimension like Unit of measure, Unit multiplier, Currency, Time format, etc. and useful information for presentation purposes like the default number of decimal positions and the suggested sort order for the indicators.

Two particular attributes provide data structure information: the `INDICATOR` that provides the indicator code and `SOURCE` which refers to the type of source of the survey e.g. LFS, PC, etc..

The descriptive metadata in ILOSTAT is presented in three free-text attributes associated to each observation value: `SOURCE_NOTE`, `INDICATOR_NOTE` and `CLASSIFICATION_NOTE`.

A fourth attribute `CURRENCY_NOTE` contains the currency in a "code: label" free text format, when applicable.

There is one type of metadata which is recognized by the name of the attribute: `OBS_STATUS` is a flag associated to the observation value, and there will be only one per observation.

III.4.5 Range specification and total observation values count

`Http Range` header specification can be used to specify the range of observation values to be returned by the query.

The value to be submitted for the `Range` key is `values=<start>-<end>`

For example, `values=1-10` will return only the first 10 cases (i.e. Primary measure values) regardless of any dimension values. Please note that this refers to actual cases in the dataset, while `firstNObservations` (See 4 above) and `lastNObservations` (See 5 above) parameters take into consideration the dimensions' key set up to the *Time dimension*.

Besides, in the `http` header of the returned message, the `Content-Range` key will contain the following information: `values <start>-<end>/<total>`. This provides a very simple and efficient

way of assessing the size of a dataset by querying for a 1 – 1 range to obtain the total count of observation values to be delivered by the full query.

Example:

The query `.../data/ILO,DF_ILOEST_ALL_EMP_2EMP_SEX_ECO_NB` would return a SDMX-ML message of several megabytes with a response time of more than one minute.

By specifying a `Range` with `values=1-1`, the response is received in less than a second with a `Content-Range` containing values `1-1/757350`, making it possible to warn the user before actually executing the query that it will return 757350 observation values, which is a huge amount of data to be transferred.

IV Annex I: SDMX 2.1 RESTful web services cheat sheet (v1.4.0)

Structural metadata queries: <https://ws-entry-point/resource/agencyID/resourceID/version/itemID?queryStringParameters>

Default

resource	The type of metadata to be returned. Values: datastructure, metadatastructure, categoryscheme, conceptscheme, codelist, hierarchicalcodelist, organisationscheme, agencyScheme, dataproviderscheme, dataconsumerscheme, organisationunitscheme, dataflow, metadataflow, reportingtaxonomy, provisionagreement, structureset, process, categorisation, contentconstraint, attachmentconstraint, actualconstraint, allowedconstraint, structure	
agencyID	Agency maintaining the artefact (e.g.: SDMX)	all
resourceID	Artefact ID (e.g.: CL_FREQ)	all
version	Artefact version (e.g.: 1.0)	latest
itemID	ID of the item (for item schemes) or hierarchy (for hierarchical codelists) to be returned.	all
detail	Desired amount of information. Values: allstubs, referencestubs, allcompletestubs, referencecompletestubs, referencepartial, full.	full
references	References to be returned with the artefact. Values: none, parents, parentsandsiblings, children, descendants, all, any type of resource.	none

Data queries: <https://ws-entry-point/resource/flowRef/key/providerRef?queryStringParameters>

Default

resource	data or metadata	
flowRef	Dataflow ref (e.g. ECB, EXR, latest)	
key	Key of the series to be returned (e.g: D.NOK.EUR.SP00.A). Wildcarding (e.g: D. .EUR.SP00.A) and OR (e.g: D.NOK+RUB.EUR.SP00.A) supported.	all
providerRef	Data provider (e.g.: ECB)	all
startPeriod	Start period (inclusive). ISO8601 (e.g. 2014-01) or SDMX reporting period (e.g. 2014-Q3).	
endPeriod	End period (inclusive). ISO8601 (e.g. 2014-01-01) or SDMX reporting period (e.g. 2014-W53).	
updatedAfter	Last time the query was performed. Used to retrieve deltas. Must be percent-encoded (e.g.: 2009-05-15T14%3A15%3A00%2B01%3A00)	
firstNObservations	Maximum number of observations starting from the first observation	
lastNObservations	Maximum number of observations counting back from the most recent observation	
dimensionAtObservation	Id of the dimension attached at the observation level	TIME_PERIOD
detail	Desired amount of information to be returned. Values: full, dataonly, serieskeyonly, nodata	full
includeHistory	Whether to return vintages	false

Legend: **Mandatory path parameter** / **Optional path parameter** / **Query string parameter (all optional)**

For **available data** queries, cf. <https://github.com/sdmx-twg/sdmx-rest/wiki/Data-Availability>

Useful HTTP headers

If-Modified-Since	Get the data only if something has changed
Accept	Select the desired format
Accept-Encoding	Compress the response

Supported formats

SDMX-ML Generic Data	<code>application/vnd.sdmx.genericdata+xml;version=2.1</code>
SDMX-ML StructureSpecific Data	<code>application/vnd.sdmx.structurespecificdata+xml;version=2.1</code>
SDMX-JSON Data	<code>application/vnd.sdmx.data+json;version=1.0.0</code>
SDMX-CSV Data	<code>application/vnd.sdmx.data+csv;version=1.0.0</code>
SDMX-ML Structure	<code>application/vnd.sdmx.structure+xml;version=2.1</code>
SDMX-JSON Structure	<code>application/vnd.sdmx.structure+json;version=1.0.0</code>
SDMX-ML Schemas	<code>application/vnd.sdmx.schema+xml;version=2.1</code>
SDMX-ML Generic Metadata	<code>application/vnd.sdmx.genericmetadata+xml;version=2.1</code>
SDMX-ML StructureSpecific Meta	<code>application/vnd.sdmx.structurespecificmetadata+xml;version=2.1</code>

Period formats

Daily/Business	<code>YYYY-MM-DD</code>
Weekly	<code>YYYY-W[01-53]</code>
Monthly	<code>YYYY-MM</code>
Quarterly	<code>YYYY-Q[1-4]</code>
Semi-annual	<code>YYYY-S[1-2]</code>
Annual	<code>YYYY</code>

Useful response codes

200 (OK)	Life is good
304 (No change)	No change since the timestamp supplied in the If-Modified-Since header
400 (Syntax error)	Your query checking you must
401 (Login needed)	Only available on a need to know basis
403 (Semantic error)	The syntax of the query is OK but it makes no sense anyway. Tolerated only on April Fools
404 (Not found)	No results. Mostly harmless, especially if you used updatedAfter
406 (Invalid format)	Not a supported format. No worries, SDMX is all you need
413 (Results too large)	You know greed is a sin, right?
414 (URI too long)	Write short sentences
500 (Server error)	Someone somewhere is having a bad day
501 (Not implemented)	Feature coming soon in a web service near you
503 (Unavailable)	Try again later.

Tips and tricks

- **Trust but verify!** Check the identity of the web service by connecting over https.
- **Know your mime type!** Specify the output format you want, using the HTTP **Accept** header. Else, the web service will determine the format and this choice may change over time (for example, when new versions of SDMX are released).
- **Laziness is a virtue!** Don't lose time, resources and bandwidth processing the response if nothing has changed! Use the **updatedAfter** parameter if you can handle deltas, or the **If-Modified-Since** header otherwise.
- **Travel light!** Compress the response, using the HTTP **Accept-Encoding** header.
- **The answer is 42.** Or not. Check the status code of the response. It's not always 200 (or 42).
- **Read The Fine Manual.** Check the SDMX RESTful specification (Section 7): http://sdmx.org/?page_id=10
- **Contact us!** You think a killer feature is missing? Drop us a line at twg@sdmx.org.