

# Package ‘eurodata’

May 8, 2026

**Type** Package

**Title** Fast and Easy Eurostat Data Import and Search

**Description** Interface to Eurostat’s API (SDMX 2.1) with fast data.table-based import of data, labels, and metadata. On top of the core functionality, data search and data description/comparison functions are also provided.

Use [https://github.com/alekrutkowski/eurodata\\_codegen](https://github.com/alekrutkowski/eurodata_codegen) — a point-and-click app for rapid and easy generation of richly-commented R code — to import a Eurostat dataset or its subset (based on the eurodata::importData() function).

**Version** 1.7.0

**Date** 2023-05-05

**License** GPL-2

**URL** <https://github.com/alekrutkowski/eurodata/>

**Encoding** UTF-8

**Depends** R (>= 3.5.0)

**Imports** utils, Rcpp, magrittr, data.table, R.utils, xtable, memoise, stringr, xml2

**Collate** RcppExports.R const.R helpers.R core\_import.R core\_search.R extras.R

**LinkingTo** Rcpp

**RoxygenNote** 7.2.3

**NeedsCompilation** yes

**Author** Aleksander Rutkowski [aut, cre]

**Maintainer** Aleksander Rutkowski <alek.rutkowski@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-05-06 10:00:02 UTC

## Contents

as.EurostatDataList . . . . .	2
browseDataList . . . . .	3
compare . . . . .	4
describe . . . . .	5
find . . . . .	6
importData . . . . .	7
importDataList . . . . .	8
importDimLabel . . . . .	8
importLabels . . . . .	9
importMetabase . . . . .	10
<b>Index</b>	<b>11</b>

---

as.EurostatDataList    *Coerce a data.frame to a EurostatDataList*

---

### Description

Some manipulations of the EurostatDataList data.frame (imported with [importDataList](#)) e.g. filtering with package **dplyr** may remove the S3 class tag EurostatDataList. This function coerces it back to EurostatDataList after checking that the critical columns (PCode, Dataset name,Link) are present. This is useful if a user wants to print and browse this filtered data.frame as a specially formatted HTML table.

### Usage

```
as.EurostatDataList(x, SearchCriteria = "", ...)
```

### Arguments

**x**                    A (most likely filtered subset of) EurostatDataList data.frame returned by [importDataList](#).

**SearchCriteria**    A string describing the search criteria used for filtering/subsetting.

**...**                Additional arguments to be passed to or from methods (currently not used).

### Value

A data.frame of S3 class EurostatDataList.

---

browseDataList	<i>Search Eurostat datasets and see the result as a table in a browser</i>
----------------	--

---

## Description

Search Eurostat datasets and see the result as a table in a browser

## Usage

```
browseDataList(subs)
```

## Arguments

subs	<p>An expression to be passed to <a href="#">subset</a>. The column names of the table of datasets can be used – those with spaces should be backtick (‘) quoted. See the examples below. The names of the available columns are:</p> <ul style="list-style-type: none"><li>• `Data subgroup, level 0`</li><li>• `Data subgroup, level 1`</li><li>• `Data subgroup, level 2`</li><li>• `Data subgroup, level 3`</li><li>• `Data subgroup, level 4`</li><li>• `Data subgroup, level 5`</li><li>• `Data subgroup, level 6`</li><li>• `Data subgroup, level 7`</li><li>• `Dataset name`</li><li>• `Code`</li><li>• `Type`</li><li>• `Last update of data`</li><li>• `Last table structure change`</li><li>• `Data start`</li><li>• `Data end`</li><li>• `Link`</li></ul>
------	---

## Value

- Side effect (via print) – a table opened in a browser via [browseURL](#).
- Value – a list with:
  - criteria – a string, search criteria,
  - time – the time of the query,
  - df – a data.frame, imported via [importDataList](#) and filtered based on the conditions specified in subs.
  - html – a string, with the HTML code that generated the table in a browser.

**Examples**

```
## Not run:
browseDataList(grepl('servic',`Dataset name`))
browseDataList(grepl('bop',Code) & !grepl('its',Code))

## End(Not run)
## Not run:
browseDataList(grepl('GDP',`Dataset name`) &
grepl('main',`Dataset name`) &
  grepl('international',`Dataset name`) &
  !grepl('quarterly',`Dataset name`))
browseDataList(grepl('bop',Code) & grepl('its',Code))

## End(Not run)
```

---

compare	<i>Compare specific Eurostat datasets on the basis of information from Metabase</i>
---------	---

---

**Description**

Compare specific Eurostat datasets on the basis of information from Metabase

**Usage**

```
compare(..., import_labels = TRUE, import_dim_labels = TRUE)
```

**Arguments**

... Two or more Eurostat dataset code names, e.g. "nama\_10\_gdp" or "bop\_its6\_det", as strings.

import\_labels Boolean: should labels for the codes inside dimensions be imported. Default: TRUE.

import\_dim\_labels Boolean: should the dimensions (e.g. geo, indic\_is, or nace\_r2) be labelled with a descriptive name (via [importDimLabel](#)). Default: TRUE.

**Value**

A [data.table](#) with columns Dim\_name, Dim\_name\_label (if import\_dim\_labels=TRUE), Dim\_val, Dim\_val\_label (if import\_labels=TRUE), and logical columns corresponding to the dataset names in ... indicating in which dataset a given dimension and dimension value appears and in which it does not.

**Examples**

```
## Not run:
compare('nama_10_gdp', 'nama_10_pe')

## End(Not run)
```

---

describe	<i>Describe a given Eurostat dataset on the basis of information from Metabase</i>
----------	--

---

**Description**

Describe a given Eurostat dataset on the basis of information from Metabase

**Usage**

```
describe(
  EurostatDatasetCode,
  import_labels = !wide,
  wide = FALSE,
  import_dim_labels = TRUE
)
```

**Arguments**

EurostatDatasetCode	A string with Eurostat dataset code name, e.g. "nama_10_gdp" or "bop_its6_det". See e.g.: <a href="https://ec.europa.eu/eurostat/databrowser/explore/all/all_themes">https://ec.europa.eu/eurostat/databrowser/explore/all/all_themes</a> where, once you follow one of the "branches" of the "tree" of datasets, the dataset codes are in tiny grey font in square brackets just under the full names of the datasets (the names are in navy blue and preceded by a cube icon).
import_labels	Boolean: should labels for the codes inside dimensions be imported. Default: if wide is FALSE then import_labels is TRUE and vice versa.
wide	Boolean: should each dimension be compressed to one row and all values within each dimension to a single, comma-separated string. Default: FALSE.
import_dim_labels	Boolean: should the dimensions (e.g. geo, indic_is, or nace_r2) be labelled with a descriptive name (via <a href="#">importDimLabel</a> ). Default: TRUE.

**Value**

A [data.table](#) with columns Dim\_name, Dim\_name\_label (if import\_dim\_labels=TRUE), either Dim\_val (if wide=FALSE) or Dim\_values (if wide=TRUE), Dim\_val\_label (if import\_labels=TRUE), and a column with a name = EurostatDatasetCode with all its values = TRUE.

## Examples

```
## Not run:
describe('nama_10_gdp')

## End(Not run)
```

---

find	<i>Search Eurostat datasets and see the result as text</i>
------	--

---

## Description

A tool for a quick ad-hoc search.

## Usage

```
find(...)
```

## Arguments

... A series of unquoted words to be searched either in Eurostat dataset codes or in dataset full names. All words not preceded by minus (-) will be linked with logical AND; all words preceded by a minus entail exclusion (logical NOT), a bit like in Google search. It is possible to search also with phrases that include spaces – in such a case the phrases should be quoted. Partial word/phrase match is applied. See the examples below.

## Value

- Side effect (via `print`) – a text report file opened via `file.show`.
- Value – a list with:
  - `criteria` – a string, search criteria,
  - `time` – the time of the query,
  - `df` – a `data.frame`, imported via `importDataList` and filtered based on the conditions specified in ...,
  - `report` – a string, with the text report.

## Examples

```
## Not run:
find(bop, its)
find(bop, -ybk, its)
find(nama_)
find(nama, 10, 64)
find('economic indic')

## End(Not run)
```



---

importDataList	<i>Import and reshape Eurostat inventory of datasets</i>
----------------	--

---

**Description**

Import and reshape Eurostat inventory of datasets

**Usage**

```
importDataList()
```

**Value**

The imported data.frame reflects the hierarchical structure of datasets (see the columns Data subgroup, level 0, Data subgroup, level 1, Data subgroup, level 2, etc.). It is tagged with S3 class EurostatDataList.

**Examples**

```
## Not run:
importDataList()

## End(Not run)
```

---

importDimLabel	<i>Import Eurostat label (description) of a given dimension code</i>
----------------	--

---

**Description**

Import the appropriate description file for the selected Eurostat dimension, e.g. for "geo" it is "Geopolitical entity (reporting)", for "nace\_r2" it is "Classification of economic activities - NACE Rev.2", for "indic\_sb" it is "Economical indicator for structural business statistics" etc. Click on "Code lists" just under "Apply download operations on" at <https://ec.europa.eu/eurostat/databrowser/bulk?lang=en> for the list of all codes. Each description is imported from inside the XML file (via the path: *m:Structure / m:Structures / s:Codelists / s:Codelist / c:Name xml:lang="en"*) from the respective URL, e.g. for "geo" it is <https://ec.europa.eu/eurostat/api/dissemination/sdmx/2.1/codelist/ESTAT/GEO>.

**Usage**

```
importDimLabel(EurostatDimCode)
```

**Arguments**

EurostatDimCode

A string – the code name of the Eurostat dimension, e.g. "geo" or "nace\_r2" or "indic\_sb", etc.

**Value**

A character vector of length 1: the label/description of EurostatDimCode.

**Examples**

```
## Not run:  
importDimLabel('nace_r2')  
  
## End(Not run)
```

---

importLabels	<i>Import Eurostat code list: labels (descriptions) for a given dimension code</i>
--------------	--

---

**Description**

Import the appropriate ‘code list’ from for the selected Eurostat dimension, e.g. geo (countries or other geographic entities), nace\_r2 (sectors), indic\_sb (indicators), etc.

**Usage**

```
importLabels(EurostatDimCode)
```

**Arguments**

EurostatDimCode  
A string – the code name of the Eurostat dimension, e.g. geo or nace\_r2 or indic\_sb.

**Value**

A data.frame with 2 columns: codes (with a name determined by EurostatDimCode) and corresponding labels (named with suffix \_labels).

**Examples**

```
## Not run:  
importLabels('nace_r2')  
  
## End(Not run)
```

---

importMetabase	<i>Import Eurostat “Metabase”</i>
----------------	-----------------------------------

---

### Description

The Eurostat “Metabase” shows which datasets contain which dimensions (where a dimension is e.g. geo or nace\_r2 or indic\_sb) and, within each dataset and dimension, which codes (e.g. which countries for the geo dimension).

### Usage

```
importMetabase()
```

### Value

The imported data.frame which reflects the hierarchical structure described above. It is a ‘flat’ data.frame with 3 columns, where each row corresponds to the combination of:

- Code – Eurostat dataset code names, e.g. "nama\_10\_a64"
- Dim\_name – Eurostat dimension code names, e.g. "nace\_r2"
- Dim\_val – Eurostat dimension code values, e.g. "EU28" if Dim\_name is "geo"; not to be confused with the actual numeric values in the actual datasets

### Examples

```
## Not run:  
importMetabase()  
  
## End(Not run)
```

# Index

`as.EurostatDataList`, 2

`browseDataList`, 3

`browseURL`, 3

`compare`, 4

`data.table`, 4, 5

`describe`, 5

`file.show`, 6

`find`, 6

`importData`, 7

`importDataList`, 2, 3, 6, 8

`importDimLabel`, 4, 5, 8

`importLabels`, 9

`importMetabase`, 10

`subset`, 3