

# Package ‘csodata’

August 11, 2023

**Type** Package

**Title** Download Data from the CSO 'PxStat' API

**Version** 1.4.2

**Date** 2023-08-10

**Maintainer** Conor Crowley <conor.crowley@csodata.ie>

**Description** Imports 'PxStat' data in JSON-stat format and (optionally) reshapes it into wide format. The Central Statistics Office (CSO) is the national statistical institute of Ireland and 'PxStat' is the CSO's online database of Official Statistics. This database contains current and historical data series compiled from CSO statistical releases and is accessed at <<http://data.csodata.ie>>.

The CSO 'PxStat' Application Programming Interface (API), which is accessed in this package, provides

access to 'PxStat' data in JSON-stat format at <<http://data.csodata.ie>>.

This dissemination tool allows developers machine to machine access to CSO 'PxStat' data.

**Imports** dplyr, httr, jsonlite, reshape2, rjstat, R.cache, sf, lubridate, tidyr, lifecycle

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Suggests** knitr, rmarkdown, leaflet, viridisLite

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Eoin Horgan [aut] (<<https://orcid.org/0000-0002-3446-6154>>),

Conor Crowley [aut, cre],

Vytas Vaiciulis [aut],

Mervyn O'Luing [aut],

James O'Rourke [aut]

**Repository** CRAN

**Date/Publication** 2023-08-11 14:33:47 UTC

## R topics documented:

csodata	2
csodata_clear_cache	3
csodata_disp_geo_meta	3
csodata_disp_meta	4
csodata_get_content	4
csodata_get_data	5
csodata_get_geo	6
csodata_get_geo_meta	8
csodata_get_interval	8
csodata_get_meta	9
csodata_get_toc	10
csodata_get_vars	11
csodata_get_var_values	12
csodata_search_toc	13
<b>Index</b>	<b>14</b>

---

csodata	<i>csodata: A package for downloading CSO data.</i>
---------	---

---

### Description

The csodata package allows for easily downloading CSO (Central Statistics Office, the National Statistics Institute of Ireland) PxStat data into R.

### Details

A specific table can be downloaded using [csodata\\_get\\_data](#), while a list of all tables currently available and their titles can be found using [csodata\\_get\\_toc](#) and [csodata\\_search\\_toc](#) is used to search their descriptions. Metadata for a specified table can be retrieved with [csodata\\_get\\_meta](#), or printed on the console using [csodata\\_disp\\_meta](#).

[csodata\\_get\\_vars](#), [csodata\\_get\\_interval](#), and [csodata\\_get\\_content](#) all return a subset of the full metadata of a table. [csodata\\_get\\_var\\_values](#) returns all the variables in the tables.

These functions provide the option to cache the returned data using the R.cache package. The cache can be deleted using [csodata\\_clear\\_cache](#).

ESRI shapefiles covering the country in varying degrees of granularity can be downloaded from [csodata.ie](#) and imported as an sf data frame using the [csodata\\_get\\_geo](#) function. Metadata about the map data can be retrieved with [csodata\\_get\\_geo\\_meta](#), and displayed on the console with [csodata\\_disp\\_geo\\_meta](#).

---

cso_clear_cache	<i>Clear csodata cache</i>
-----------------	----------------------------

---

**Description**

Deletes all data cached by the csodata package. The cached data from the csodata package is stored in a subdirectory of the default R.cache cache at `R.cache::getCachePath()`. This function provides a quick way to delete those files along with the directory to free up space.

**Usage**

```
cso_clear_cache()
```

**Value**

Does not return a value, deletes the csodata cache.

**Examples**

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

---

cso_disp_geo_meta	<i>Prints metadata from an ESRI shapefile to console</i>
-------------------	--

---

**Description**

Takes the output from `cso_get_geo` or otherwise and prints information about it to the console as formatted text.

**Usage**

```
cso_disp_geo_meta(shp)
```

**Arguments**

shp                   sf data.frame. Geographic data stored as an sf object.

**Value**

Does not return any values, rather the function prints the shapefile metadata to console.

**Examples**

```
## Not run:  
cso_disp_geo_meta(shp)  
  
## End(Not run)
```

---

cso_disp_meta	<i>Prints metadata from a PxStat table to the console</i>
---------------	---

---

**Description**

Takes the output from [cso\\_get\\_meta](#) and prints it to the console as formatted text.

**Usage**

```
cso_disp_meta(table_code)
```

**Arguments**

table\_code      string. A valid code for a table on data.cso.ie .

**Value**

Does not return any values, rather the function prints the tables metadata to console.

**Examples**

```
## Not run:  
cso_disp_meta("EP001")  
  
## End(Not run)
```

---

cso_get_content	<i>Returns a character vector listing the statistics in a CSO data table</i>
-----------------	--

---

**Description**

Returns a character vector listing the statistics in a CSO data table

**Usage**

```
cso_get_content(table_code, cache = TRUE, flush_cache = TRUE)
```

**Arguments**

table_code	string. A valid code for a table on data.cso.ie .
cache	logical. Whether to use cached data, if available. Default value is TRUE.
flush_cache	logical. If TRUE (default) the cache will be checked for old, unused files. Any files which have not been accessed in the last month will be deleted

**Value**

character vector. The names of the statistics included in the table, with one element for each statistic.

**Examples**

```
## Not run:
var_cont <- cso_get_content("EP008")

## End(Not run)
```

---

cso_get_data	<i>Return a CSO table as a data frame</i>
--------------	---

---

**Description**

Returns a CSO table from the CSO PxStat Application Programming Interface (API) as a data frame, with the option to give it in wide format (default) very wide or long format.

**Usage**

```
cso_get_data(
  table_code,
  pivot_format = "wide",
  wide_format = lifecycle::deprecated(),
  include_ids = FALSE,
  id_list = NULL,
  use_factors = TRUE,
  use_dates = FALSE,
  cache = TRUE,
  flush_cache = TRUE
)
```

**Arguments**

table_code	string. If the table_code is a filename or a path to a file, e.g. "QNQ22.json", it is imported from that file. Otherwise if it is only a table code e.g. "QNQ22", the file is downloaded from data.cso and checked to see if it is a valid table.
------------	---

<code>pivot_format</code>	string, one of "wide", "very_wide", "tall" or "tidy". If "wide" (default) the table is returned in wide (human readable) format, with statistic as a column (if it exists). If "very_wide" the table is returned wide format and spreads the statistic column to rows. If "tall" the table is returned in tall (statistic and value) format. If "tidy" will be returned in a tidy-like format.
<code>wide_format</code>	string. Deprecated argument as of 1.4.0. Please use <code>pivot_format</code> instead.
<code>include_ids</code>	logical. The JSON-stat format stores variables as ids i.e. IE11 and labels i.e. Border. While the label is generally preferred, sometimes it is useful to have the ids to match on. If <code>include_ids</code> is TRUE (default) then ids are retrieved and appended to the table to the right of the original column with the name <code>&lt;columnName&gt;.id</code> .
<code>id_list</code>	either NULL (default) or a character vector of columns that should have ids appended if <code>include_ids</code> is TRUE. if NULL then every column that is not included in the vector <code>remove_id</code> will be used.
<code>use_factors</code>	logical. If TRUE (default) factors will be used in strings.
<code>use_dates</code>	logical. If True dates will be returned as date-time competent. Default is FALSE.
<code>cache</code>	logical. if TRUE (default) <code>csodata</code> will cache the result using <code>R.cache</code> . The raw data downloaded from the <code>data.csi.ie</code> is cached, which means that calling <code>cso_get_data</code> with the same <code>table_code</code> but different parameters will result in cached data being used.
<code>flush_cache</code>	logical. If TRUE (default) the cache will be checked for old, unused files. Any files which have not been accessed in the last month will be deleted.

### Details

The data is pulled from the ResponseInstance service on the CSO API in JSON-Stat format, using the GET method from the `httr` package.

### Examples

```
## Not run:
tbl1 <- cso_get_data("QNQ22")
tbl2 <- cso_get_data("QLF07.json")

## End(Not run)
```

---

`cso_get_geo`

*Return geographic data as a sf data frame*

---

### Description

Retrieves an ESRI shapefile of vector data for Ireland from the `cso` website [cso.ie](http://cso.ie) and returns it as an `sf` data frame. The data is returned as a zip file, which is downloaded to and unzipped in a temporary directory.

**Usage**

```
cso_get_geo(map_data, cache = TRUE, flush_cache = TRUE)
```

**Arguments**

map_data	<p>string. Indicates which shapefile to download. Options are:</p> <ul style="list-style-type: none"> <li>• "Local Authorities", "County Councils", "la" OR "cc"</li> <li>• "Local Authorities 2016", "County Councils 2016", "la2016" OR "cc2016"</li> <li>• "Constituencies" OR "Constituencies (2017)" OR "con"</li> <li>• "Constituencies_2013" OR "Constituencies (2013)"</li> <li>• "Electoral Divisions" OR "elec_div" OR "ed"</li> <li>• "Gaeltacht" OR "g"</li> <li>• "Language Planning Areas" OR "lpa"</li> <li>• "Local Electoral Areas (2019)" , "lea_2019" , "lea (2019)" , "Local Electoral Areas" OR "lea"</li> <li>• "Local Electoral Areas (2014)" OR "lea_2014" OR "lea (2014)"</li> <li>• "NUTS3" OR "nuts3"</li> <li>• "Provinces" OR "p"</li> <li>• "Settlements" OR "s"</li> <li>• "Small Areas" OR "sa"</li> </ul> <p>Until v0.1.5 "NUTS2" and "NUTS3" gave access to the 2011 dataset.</p>
cache	logical. Indicates whether to cache the result using R.cache. TRUE by default.
flush_cache	logical. If TRUE (default) the cache will be checked for old, unused files. Any files which have not been accessed in the last month will be deleted

**Details**

The map data is from the 2011 census, and is 20m generalised, which offers a good balance of fidelity and low file size. More datasets, as well as 50m generalised, 100m generalised and ungeneralised versions of the map files can also be found on the OSi (Ordnance Survey Ireland) website at <https://data-osi.opendata.arcgis.com/search?tags=boundaries>.

The NUTS2 and NUTS3 map files are the updated versions for 2016, including three NUTS2 regions and the movement of Louth and South Tipperary into new NUTS3 regions. These files are downloaded directly from the OSi website, as they are not available on the CSO website, and do not contain the population and housing data contained in the map files from the CSO website.

**Value**

data frame of the requested CSO table.

**Examples**

```
## Not run:
shp <- cso_get_geo("NUTS2")

## End(Not run)
```

---

`cso_get_geo_meta`      *Returns a data frame with the metadata of a vector shapefile*

---

### Description

Takes the output from `cso_get_geo` or otherwise and returns information about it in a data frame.

### Usage

```
cso_get_geo_meta(shp)
```

### Arguments

`shp`      sf data.frame. Geographic data stored as an sf object.

### Value

list with eight elements:

- The coordinate reference system, itself a list with two elements, the EPSG code (if any, NA value if none), and the proj4string
- The number of polygons in the data
- If all the polygons are simple (not self-intersecting)
- If any polygons are empty
- If all of the polygons are valid
- The average area of the polygons, including units

### Examples

```
## Not run:  
shp_meta <- cso_get_geo_meta(shp)  
  
## End(Not run)
```

---

`cso_get_interval`      *Returns a the time interval used to record data in a CSO table*

---

### Description

Reads the metadata of a table to return an atomic character vector displaying the intervals at which the data included in the table was gathered/calculated.

### Usage

```
cso_get_interval(table_code, cache = TRUE, flush_cache = TRUE)
```



**Arguments**

table_code	string. A valid code for a table on data.cso.ie .
cache	logical. Whether to use cached data, if available. Default value is TRUE. Strongly recommended to use caching, as otherwise the entire table could be downloaded only to access a small part of its metadata.
flush_cache	logical. If TRUE (default) the cache will be checked for old, unused files. Any files which have not been accessed in the last month will be deleted

**Value**

character vector. The names of the statistics included in the table, with one element for each statistic.

**Examples**

```
## Not run:
interval <- cso_get_interval("C0636")

## End(Not run)
```

---

cso_get_meta	<i>Returns a data frame with the metadata of a CSO data table</i>
--------------	---

---

**Description**

Checks the CSO PxStat API for a metadata on a dataset and returns it as a list of metadata and contained statistics.

**Usage**

```
cso_get_meta(table_code, cache = TRUE, flush_cache = TRUE)
```

**Arguments**

table_code	string. A valid code for a table on data.cso.ie .
cache	logical. Whether to use cached data, if available. Default value is TRUE.
flush_cache	logical. If TRUE (default) the cache will be checked for old, unused files. Any files which have not been accessed in the last month will be deleted

**Value**

list with nine elements:

- The title of the table.
- The units used (the R class of the value column)
- The Copyright on the data.
- The time interval used in the data. (Census year, Quarter, Month)

- The date the table was last modified.
- The names of the variables included in the table, returned as a character vector with one element for each variable.
- The names of the statistics included in the table, returned as a character vector with one element for each statistic.
- An indicator if the statistics are experimental
- Returns if the data is geographic

### Examples

```
meta1 <- cso_get_meta("HS014", cache = FALSE)
```

---

cso_get_toc	<i>Returns a data frame with all valid CSO PxStat tables listed sequentially by id number, e.g. A0101, A0102, A0103, etc.</i>
-------------	---

---

### Description

Checks the CSO PxStat API for a list of all the table codes (e.g. A0101, A0102, A0103, etc.), which also includes date last modified and title for each table, and returns this list as an R data frame.

### Usage

```
cso_get_toc(
  cache = TRUE,
  suppress_messages = FALSE,
  get_frequency = FALSE,
  list_vars = FALSE,
  flush_cache = TRUE,
  from_date = lubridate::date(lubridate::today() - lubridate::years(2))
)
```

### Arguments

cache	logical. If TRUE (default) the table of contents is cached with the system date as a key.
suppress_messages	logical. If FALSE (default) a message is printed when loading a previously cached table of contents.
get_frequency	logical. If TRUE the frequency of each table(yearly, monthly etc...) will be returned as an additional column in the table of contents.
list_vars	logical. If TRUE an additional column will be added to the table of contents which lists each tables variables.
flush_cache	logical. If TRUE (default) the cache will be checked for old, unused files. Any files which have not been accessed in the last month will be deleted.
from_date	date in the format YYYY-MM-DD or Null. Will only return tables last modified after date provided. Default is 2 years from current date.

**Details**

The data is pulled from the ReadCollection on the CSO API. See <https://github.com/CSOIreland/PxStat/wiki/API-Cube-RESTful> for more information on this.

**Value**

data frame of three character columns:

- `id`. Contains all of the table codes currently available on the CSO API.
- `LastModified`. The date the table was last modified in POSIXct format.
- `title`. The title of the table.

**Examples**

```
## Not run:
head(cso_get_toc())

## End(Not run)
```

---

`cso_get_vars`

*Returns a character vector listing the contents of a CSO data table*

---

**Description**

Reads the metadata of a table to return a character vector of the included variables and statistics in the table.

**Usage**

```
cso_get_vars(table_code, cache = TRUE, flush_cache = TRUE)
```

**Arguments**

<code>table_code</code>	string. A valid code for a table on <a href="http://data.cso.ie">data.cso.ie</a> .
<code>cache</code>	logical. Whether to use cached data, if available. Default value is TRUE. Strongly recommended to use caching, as otherwise the entire table could be downloaded only to access a small part of its metadata.
<code>flush_cache</code>	logical. If TRUE (default) the cache will be checked for old, unused files. Any files which have not been accessed in the last month will be deleted.

**Value**

character vector. The names of the statistics included in the table.

## Examples

```
## Not run:  
cso_get_vars("IPA03")  
  
## End(Not run)
```

---

`cso_get_var_values`      *Returns a list of the values of variables of a CSO data table*

---

## Description

Reads the table to determine all the unique values taken by the variables in the table and returns them as a list.

## Usage

```
cso_get_var_values(table_code, cache = TRUE, flush_cache = TRUE)
```

## Arguments

<code>table_code</code>	string. A valid code for a table on data.cso.ie .
<code>cache</code>	logical. Whether to use cached data, if available. Default value is TRUE.
<code>flush_cache</code>	logical. If TRUE (default) the cache will be checked for old, unused files. Any files which have not been accessed in the last month will be deleted

## Value

list. It has length equal to the number of variables in the table, and each element is a character vector which has all the values taken by one variable.

## Examples

```
## Not run:  
var_val <- cso_get_var_values("IPA03")  
  
## End(Not run)
```

---

cso_search_toc	<i>Search list of all table descriptions for given string</i>
----------------	---

---

**Description**

Searches the list of all table descriptions returned by `cso_get_toc()` for a given substring.

**Usage**

```
cso_search_toc(  
  string,  
  toc = cso_get_toc(suppress_messages = TRUE, flush_cache = FALSE, from_date = NULL)  
)
```

**Arguments**

<code>string</code>	string. The text to search for. Case insensitive.
<code>toc</code>	data.frame. The table of contents as returned by <code>cso_get_toc</code> . If not given, will be re-downloaded (or retrieved from cache) using <code>cso_get_toc()</code> .
<code>flush_cache</code>	logical. If TRUE the cache will be checked for old, unused files. Any files which have not been accessed in the last month will be deleted strings.

**Value**

data frame of three character columns, with layout identical to that of `cso_get_toc`. A subset of the results of `cso_get_toc`, with only rows where the description field contains the entered string.

**Examples**

```
## Not run:  
trv <- cso_search_toc("travel")  
  
## End(Not run)
```

# Index

[cso\\_clear\\_cache](#), [2](#), [3](#)  
[cso\\_disp\\_geo\\_meta](#), [2](#), [3](#)  
[cso\\_disp\\_meta](#), [2](#), [4](#)  
[cso\\_get\\_content](#), [2](#), [4](#)  
[cso\\_get\\_data](#), [2](#), [5](#)  
[cso\\_get\\_geo](#), [2](#), [3](#), [6](#), [8](#)  
[cso\\_get\\_geo\\_meta](#), [2](#), [8](#)  
[cso\\_get\\_interval](#), [2](#), [8](#)  
[cso\\_get\\_meta](#), [2](#), [4](#), [9](#)  
[cso\\_get\\_toc](#), [2](#), [10](#)  
[cso\\_get\\_var\\_values](#), [2](#), [12](#)  
[cso\\_get\\_vars](#), [2](#), [11](#)  
[cso\\_search\\_toc](#), [2](#), [13](#)  
[csodata](#), [2](#)