BFS: An R package to Search and Download Swiss Federal Statistical Office Data

Félix Luginbühl[1] *(www.felixluginbuhl.com)*

1. Fachstelle für Statistik Kanton SG (August to October 2023).

Abstract

The BFS R package allows to search and download data from the Swiss Federal Statistical APIs in a dynamic and reproducible way.

Install and Load

```
install.packages("BFS")
#devtools::install_github("lgnbhl/BFS") # install from GitHub

library(BFS)
```

Search the Data Catalog

Display a list of all available datasets from the PXWeb data catalog with metadata in any language ("de", "fr", "it" or "en").

```
bfs_get_catalog_data(language = "en")
```

```
## # A tibble: 184 x 7
##   title          language publication_date    number_asset url_bfs
##   <chr>          <chr>    <dttm>                     <dbl> <chr>
## 1 Acknowledgme~ en       2023-06-22 08:30:00     25945442 https:~
## 2 Adoptions by~ en       2023-06-22 08:30:00     25945406 https:~
## 3 Deaths by in~ en       2023-06-22 08:30:00     25945423 https:~
## 4 Deaths by se~ en       2023-06-22 08:30:00     25945436 https:~
## 5 Deaths since~ en       2023-06-22 08:30:00     25945437 https:~
# i 2 more variables: url_px <chr>, catalog_date <dttm>
```

Download Data in Any Language

Get a dataset from the official PXWeb API by BFS number (FSO number).

```
bfs_get_data(number_bfs = "px-x-1502040100_131", language = "en")
```

```
## # A tibble: 18,060 x 5
##   Year    `ISCED Field`     Sex    `Level of study` `University students`
##   <chr>   <chr>             <chr>  <chr>                            <dbl>
## 1 1980/81 Education science Male   Master                             151
## 2 1980/81 Education science Male   Doctorate                          121
## 3 1980/81 Education science Female Master                             555
## 4 1980/81 Education science Female Doctorate                          306
## 5 1980/81 Education science Male   Master                             143
# i 18,055 more rows
```

Access all metadata information with `bfs_get_asset_metadata()`.

```
meta_students <- bfs_get_asset_metadata(number_asset = "24367729")
```

Query Specific Dimensions

Get variable and category code names using `bfs_get_metadata()`.

```
bfs_get_metadata(number_bfs = "px-x-1502040100_131", language = "en")
```

```
## # A tibble: 4 x 7
##   code         text         values valueTexts time  elimination title
##   <chr>        <chr>        <list> <list>     <lgl> <lgl>       <chr>
## 1 Jahr         Year         <chr>  <chr [43]> TRUE  NA          Univ…
# i 4 more rows
```

Access only specific dimensions of a dataset using the PXWeb API query.

```
bfs_get_data(
  number_bfs = "px-x-1502040100_131",
  language = "en",
  query = list(
    "Jahr" = c("40", "41"), # code values for "2020/21" and "2021/22"
    "ISCED Fach" = c("0"), # code value for "Education science"
    "Geschlecht" = c("*"), # use "*" to select all
    "Studienstufe" = c("2", "3"))) # code for "Master" and "Doctorate"
```

```
## # A tibble: 8 x 5
##   Year    `ISCED Field`     Sex    `Level of study` `University students`
##   <chr>   <chr>             <chr>  <chr>                            <dbl>
## 1 2020/21 Education science Male   Master                             151
## 2 2020/21 Education science Male   Doctorate                          121
## 3 2020/21 Education science Female Master                             555
## 4 2020/21 Education science Female Doctorate                          306
## 5 2021/22 Education science Male   Master                             143
```

Search and Download **BFS Data** in Any Language **with 1 Line of Code**

Access the full documentation: www.felixluginbuhl.com/BFS

Let's get in touch:

- **felix.luginbuhl@outlook.com**
- **linkedin.com/in/felixluginbuhl**
- **www.felixluginbuhl.com**

Swiss Geodata

Get the Geodata Catalog

Display geo-information catalog of the Swiss Official STAC API.

```
catalog_geodata <- bfs_get_catalog_geodata()

catalog_geodata
```

```
## # A tibble: 281 × 12
##   collection_id      type  href  title description created updated
##   <chr>              <chr> <chr> <chr> <chr>       <chr>   <chr>
## 1 ch.are.agglomera… API   http… Citi… "The list …" 2021-1… 2023-0…
## 2 ch.are.alpenkonv… API   http… Alpi… "The perim…" 2021-1… 2022-0…
## 3 ch.are.belastung… API   http… Load… "Passenger…" 2021-1… 2022-0…
## 4 ch.are.belastung… API   http… Load… "Passenger…" 2021-1… 2022-0…
## 5 ch.are.belastung… API   http… Load… "Vehicles …" 2021-1… 2022-0…
## 6 ch.are.belastung… API   http… Load… "Vehicles …" 2021-1… 2022-0…
## 7 ch.are.erreichba… API   http… Acce… "Accessibi…" 2021-1… 2022-0…
## 8 ch.are.erreichba… API   http… Acce… "Accessibi…" 2021-1… 2022-0…
## 9 ch.are.gemeindet… API   http… Typo… "The typol…" 2021-1… 2022-0…
## 10 ch.are.gueteklas… API   http… Publ… "The publi…" 2021-1… 2023-0…
## # i 271 more rows
## # i 3 more variables: provider_name <chr>, bbox <list>, inverval
```

Explore the Catalog

Download geodata

For example get information about the *Generalised borders G1* dataset.

```
geo_g1 <- "Generalised borders G1 and area with urban character"

catalog_geodata |>
  dplyr::filter(title == geo_g1)
```

```
## # A tibble: 1 × 12
##   collection_id      type  href  title description created updated
##   <chr>              <chr> <chr> <chr> <chr>       <chr>   <chr>
## 1 ch.bfs.generalisi… API   http… Gene… Administra… 2022-0… 2023-0…
## # i 3 more variables: provider_name <chr>, bbox <list>, inverval
```

Download geographic assets by collection id from the official STAC API.

```
coll_id <- "ch.bfs.generalisierte-grenzen_agglomerationen_g1"

bfs_download_geodata(collection_id = coll_id)
```

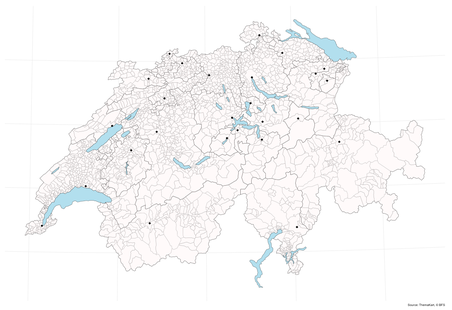
Cartographic base maps

You can get cartographic base maps from the ThemaKart project using `bfs_get_base_maps()`.

```
library(ggplot2)

switzerland <- bfs_get_base_maps(geom = "suis")
communes <- bfs_get_base_maps(geom = "polg", date = "20230101")
lakes <- bfs_get_base_maps(geom = "seen", category = "11")
districts <- bfs_get_base_maps(geom = "bezk")
cantons <- bfs_get_base_maps(geom = "kant")
cantons_capitals <- bfs_get_base_maps(
  geom = "stkt", type = "Pnts", category = "kk")

ggplot() +
  geom_sf(data = communes,
          fill = "snow", color = "grey45") +
  geom_sf(data = lakes,
          fill = "lightblue2", color = "black") +
  geom_sf(data = districts,
          fill = "transparent", color = "grey65") +
  geom_sf(data = cantons,
          fill = "transparent", color = "black") +
  geom_sf(data = cantons_capitals,
          shape = 18, size = 3) +
  theme_minimal() +
  theme(axis.text = element_blank()) +
  labs(caption = "Source: ThemaKart, © BFS")
```



A Use Case Example

- *Swiss City Statistics App* : choose an indicator, two cities, and have fun trying to guess which city has the highest indicator value.
  - Webpage: www.felixluginbuhl.com/applications/city-statistics