Package: inkaR (via r-universe)

November 2, 2026

Title Download and Analyze Spatial Development Data from 'INKAR'

Version 0.7.0

Description A professional R interface to download and analyze spatial development indicators from the 'BBSR' 'INKAR' (Indicators and Maps for Spatial and Urban Development) database. Features a bilingual interactive wizard, fuzzy and normalized indicator search, multi-indicator downloads with automatic tidy merging (long/wide), guaranteed consistent output schema, robust disk caching with automatic retry, and 'ggplot2' themes for regional mapping. Map boundaries come from 'Eurostat' 'GISCO' <https://ec.europa.eu/eurostat/web/gisco> and match the territorial status used by 'INKAR'.

URL https://github.com/ofurkancoban/inkaR, https://ofurkancoban.github.io/inkaR/

BugReports https://github.com/ofurkancoban/inkaR/issues

License MIT + file LICENSE

Encoding UTF-8

Language en-US

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports httr2, jsonlite, tibble, dplyr, rlang, utils, cli, tidyr

Suggests callr, stringdist, testthat (>= 3.0.0), withr, knitr, rmarkdown, sf, ggplot2, httptest2, pkgdown

VignetteBuilder knitr

Depends R (>= 4.1.0)

Config/testthat/edition 3

Config/pak/sysreqs libicu-dev libssl-dev

Repository https://ofurkancoban.r-universe.dev

Date/Publication 2026-11-02 14:29:56 UTC

RemoteUrl https://github.com/ofurkancoban/inkar

RemoteRef HEAD

RemoteSha c912a4c2f3357e6f8229ffaa47c32a33a9c349a7

Contents

inkaR-package . . . 2
clear_inkar_cache . . . 4
compare_districts . . . 4
compare_regions . . . 5
get_geographies . . . 5
get_indicators . . . 6
get_inkar_data . . . 6
get_inkar_geometry . . . 7
get_themes . . . 8
indicators . . . 9
inkaR . . . 10
inkar_shortcut . . . 11
inkar_trends . . . 12
plot_inkar . . . 12
search_indicators . . . 13
select_indicator . . . 14
select_level . . . 14
select_years . . . 15
theme_inkaR . . . 15
update_indicators . . . 16
view_indicators . . . 16

Index **17**

`inkaR-package` *inkaR: Access the INKAR Database of the BBSR*

Description

The inkaR package provides a user-friendly interface to access, search, and download statistical indicators from the INKAR (Indikatoren und Karten zur Raum- und Stadtentwicklung) database provided by the Federal Institute for Research on Building, Urban Affairs and Spatial Development (BBSR).

Main Functions

- `view_indicators()`: Opens the list of available indicators in a viewer. Supports German ("de") and English ("en") modes. Sorted by active status.
- `get_inkar_data()`: API wrapper to fetch data for a specific indicator. Supports auto-saving to CSV (`csv = TRUE`).

- `get_geographies()`: Helper to list available spatial levels (e.g., Kreise, Gemeinden) or regions within a level.
- `search_indicators()`: Search for indicators by text pattern.

Usage Workflow

1. **Explore**: Use `view_indicators()` to find the `Shortname` (e.g., "001") or `M_ID` of the desired indicator.
2. **Download**: Use `get_inkar_data("001", level="KRE", year=2021)` or ranges (`year=2010:2020`) to fetch the data.
3. **Export**: Add `csv = TRUE` to `get_inkar_data` to save the result immediately.

Author(s)

Maintainer: Omer Furkan Coban <oemer.furkan.coban@uni-oldenburg.de> (ORCID)

See Also

Useful links:

- https://github.com/ofurkancoban/inkaR
- https://ofurkancoban.github.io/inkaR/
- Report bugs at https://github.com/ofurkancoban/inkaR/issues

Examples

```
if (interactive()) {
  # 1. View available indicators (German)
  view_indicators()

  # 2. View in English
  view_indicators("en")
}


  # 3. Search for "GDP" (Bruttoinlandsprodukt)
  try(search_indicators("GDP", lang = "en"))

  # 4. Download data for GDP (011) for Districts (KRE)
  #    Note: You can use Shortnames ("011"), numeric M_IDs (11), or simple codes ("bip")
  try(data <- get_inkar_data("011", level = "KRE", year = 2021, lang = "de", csv = FALSE))

  # 5. Download and save directly as CSV
  try(get_inkar_data("011", csv = TRUE, export_dir = tempdir()))

  # 6. Download data for a year range
  try(get_inkar_data("011", level = "KRE", year = 2010:2020))
```

clear_inkar_cache *Clear INKAR Cache*

Description

Clears the persistent disk cache used for API responses (like time reference metadata).

Usage

```
clear_inkar_cache()
```

Value

No return value, called for side effects.

compare_districts *Filter Downloaded Data to Specific Districts*

Description

A specialized wrapper around `compare_regions()` to filter a data frame returned by `get_inkar_data()` to rows matching specific district names or IDs (Kennziffer).

Usage

```
compare_districts(data, districts, exact = FALSE)

compare_district(data, districts, exact = FALSE)
```

Arguments

`data`	A data frame returned by `get_inkar_data()`.
`districts`	Character/Numeric vector. District names or IDs (Kennziffer) to keep. Names match literally as parts of names. IDs match as a prefix (`"09"` keeps all of Bavaria) or exactly with leading zeros ignored (`1001` matches `"01001"`).
`exact`	Logical. If `TRUE`, require exact string match. Default `FALSE`.

Value

A filtered tibble.

Examples

```
df <- try(get_inkar_data("011", level = "KRE", year = 2021, lang = "en"))
if (is.data.frame(df) && nrow(df) > 0) compare_districts(df, c("Berlin", "Hamburg"))
```

compare_regions *Filter Downloaded Data to Specific Regions*

Description

Filters a data frame returned by `get_inkar_data()` to rows matching the supplied region names (partial, case-insensitive match by default).

Usage

```
compare_regions(data, regions, exact = FALSE)

compare_region(data, regions, exact = FALSE)
```

Arguments

`data`	A data frame returned by `get_inkar_data()`.
`regions`	Character vector. Region names or parts of names to keep, matched literally and case-insensitively (e.g. `"Frankfurt (Oder)"`).
`exact`	Logical. If `TRUE`, require exact string match. Default `FALSE`.

Value

A filtered tibble.

Examples

```
df <- try(get_inkar_data("011", level = "KRE", year = 2021, lang = "en"))
if (is.data.frame(df) && nrow(df) > 0) compare_regions(df, c("Berlin", "Hamburg"))
```

get_geographies *Get Available Geographies or Region List*

Description

Retrieves a list of available spatial levels (if geography is NULL) or a list of regions for a specific level (e.g., "KRE").

Usage

```
get_geographies(geography = NULL)
```

Arguments

`geography`	Character. Spatial level code (e.g. "KRE"). If NULL, returns all levels.

Value

A data frame with `ID` and `Name`.

`get_indicators` *List Available Indicators*

Description

Returns a data frame of available indicators with bilingual support.

Usage

```
get_indicators(lang = c("de", "en"))
```

Arguments

`lang`	Language code: "de" (German) or "en" (English).

Value

A tibble containing indicator IDs, names, and descriptions.

`get_inkar_data` *Download Data from INKAR*

Description

Retrieves statistical data for a given variable and spatial level. Automatically handles time reference lookup.

Usage

```
get_inkar_data(
  variable,
  level = "KRE",
  year = NULL,
  lang = c("de", "en"),
  format = c("long", "wide"),
  digits = NULL,
  csv = FALSE,
  export_dir = NULL
)
```

Arguments

variable	Character. The indicator ID (Shortname), e.g., "011".
level	Character. Spatial level code (e.g., "KRE" for Kreise).
year	Integer/Character vector. Specific year (e.g., 2021) or range (e.g., 2010:2020). If NULL, fetches all available years.
lang	Character. "de" (default) for German column names, "en" for English.
format	Character. "long" (default) for tidy format, "wide" for years as columns.
digits	Integer or NULL. If set, rounds numeric value column to this many decimal places. The INKAR API uses single-precision floats, so values like 8.45 may arrive as 8.44999...; digits = 2 corrects this. Default NULL keeps the raw API value.
csv	Logical. If TRUE, saves the data to a CSV file in the directory specified by export_dir.
export_dir	Character. Directory to save the CSV file if csv = TRUE. If NULL (default), it saves to the current working directory (". ").

Value

A tibble containing the data.

---

get_inkar_geometry *Get Administrative Boundaries Matching INKAR*

Description

Returns sf polygons for a spatial level with a Kennziffer column in the same format as get_inkar_data(), so the two can be joined directly. Boundaries come from Eurostat GISCO LAU (Gemeinden), whose territorial status matches INKAR. Higher levels are built by dissolving the Gemeinden; Gemeindeverbaende use the Destatis Gemeindeverzeichnis to assign each Gemeinde to its Verband.

Usage

```
get_inkar_geometry(
  level = c("KRE", "BND", "BLD", "GVB", "GEM"),
  refresh = FALSE
)
```

Arguments

level	Character. One of "BND", "BLD", "KRE", "GVB", "GEM". "ROR" is not available.
refresh	Logical. If TRUE, ignore the cache and download again.

Details

The GISCO file (about 43 MB) is downloaded once and the German polygons are cached. In interactive sessions the cache persists across sessions; see `clear_inkar_cache()`.

Source: (c) EuroGeographics for the administrative boundaries. Please keep this attribution in published maps.

Value

An `sf` object (CRS ETRS89-LAEA, EPSG:3035) with columns `Kennziffer` and `geometry` (plus `name` for `"GEM"`).

Examples

```
# Downloads about 43 MB on first use, so it only runs interactively
if (interactive() && requireNamespace("sf", quietly = TRUE)) {
  kreise <- get_inkar_geometry("KRE")
}
```

`get_themes` *List Available Indicator Themes*

Description

Returns the unique theme/domain values present in the local `indicators` dataset. Pass one of these values to the `theme` argument of `search_indicators()` or `view_indicators()` to narrow results.

Usage

```
get_themes()
```

Value

A sorted character vector of theme names.

Examples

```
get_themes()
```

`indicators` *INKAR Indicators Metadata*

Description

A comprehensive list of available indicators from the INKAR database. This dataset is used to lookup indicator IDs, names, and descriptions.

Usage

```
indicators
```

Format

A data frame with the following columns:

ID Short identifier from the INKAR indicator overview (e.g., "q_alo"), or the M_ID as text where INKAR publishes none

M_ID Numeric key the INKAR API expects (the wizard "Gruppe")

IndID Official INKAR indicator number from the indicator overview. It differs from M_ID for most indicators (e.g. 1101 vs 12) and is accepted by `get_inkar_data()` as well

Name_DE German name of the indicator

Short_DE Official German short name (e.g. "Arbeitslosenquote")

Name_EN English name (translated or placeholder)

Description_DE Detailed German description

Description_EN Detailed English description (missing for about a quarter of the indicators)

Theme Topic area of the INKAR wizard in German (e.g. "Arbeitslosigkeit"); see `get_themes()`

Active Logical. TRUE if the indicator is offered by the INKAR API. FALSE marks discontinued indicators kept for backward compatibility

Algorithmus Algorithm used (if any)

Anmerkungen Notes in German

Anmerkungen_EN Notes in English

Gemeinden Availability for Municipalities

Kreise Availability for Districts

Statistische Grundlagen Statistical basis (DE)

Stat_Grund_EN Statistical basis (EN)

Unit_DE Unit of measurement (DE)

Unit_EN Unit of measurement (EN)

Source

https://www.inkar.de/

inkaR *Download Data from INKAR (Interactive Alias)*

Description

A full-featured alias for `get_inkar_data()` with bilingual support and an interactive wizard when called without arguments (in interactive sessions). Call `inkaR("011")` to download directly, or `inkaR()` to open the wizard.

Usage

```
inkaR(
  variable = NULL,
  level = NULL,
  year = NULL,
  lang = c("de", "en"),
  digits = NULL,
  ...
)
```

Arguments

`variable`	Character. Indicator ID, shortname, or partial name. If `NULL` (default), opens an interactive selection menu (interactive sessions only).
`level`	Character. Spatial level code (e.g., `"KRE"` for Kreise). If `NULL` and `variable` is also `NULL`, an interactive level menu is shown.
`year`	Integer/Character vector or `"latest"`. Specific year (e.g. 2021) or range.
`lang`	Character. `"de"` (default) for German column names, `"en"` for English.
`digits`	Integer or NULL. Rounds the value column to this many decimal places. Useful to correct IEEE 754 single-precision artifacts from the API (e.g. 8.44999... -> 8.45).
`...`	Additional arguments passed to `get_inkar_data()`, such as `format` or `csv`.

Details

For a simpler English-first shortcut, see `inkar()`.

Value

A tibble containing the downloaded data, or `NULL` if selection was cancelled.

Examples

```
if (interactive()) {
  df <- inkaR()  # opens interactive menu
}


  try(df <- inkaR("bip", level = "KRE", year = 2021))
  try(df <- inkaR("Bruttoinlandsprodukt", level = "KRE"))
```

`inkar_shortcut` *Download INKAR Data (English Shortcut)*

Description

A convenience wrapper around `get_inkar_data()` with English output and `year = "latest"` as defaults. Equivalent to calling `get_inkar_data(variable, level, year = "latest", lang = "en")`.

Usage

```
inkar(
  variable,
  level = "KRE",
  year = "latest",
  lang = "en",
  digits = NULL,
  ...
)
```

Arguments

`variable`	Character. Indicator ID, short name, or partial name.
`level`	Character. Spatial level code (default `"KRE"`).
`year`	Integer/Character vector or `"latest"` (default). Year(s) to download.
`lang`	Character. Output language (default `"en"`).
`digits`	Integer or NULL. Rounds the value column to this many decimal places.
`...`	Additional arguments passed to `get_inkar_data()`.

Value

A tibble with English column names.

Examples

```
  try(df <- inkar("011", level = "KRE"))
  try(df <- inkar("011", level = "KRE", year = 2019:2021))
```

inkar_trends *Plot Time Series Trends for INKAR Indicators*

Description

Creates a ggplot2 line chart showing how indicator values change over time for selected regions. Input must be a long-format data frame from get_inkar_data().

Usage

```
inkar_trends(data, regions = NULL, title = NULL, mode = c("light", "dark"))
```

Arguments

data	A long-format data frame from get_inkar_data().
regions	Optional character vector. Region names (partial match) to include. If NULL, all regions are plotted.
title	Optional character. Custom plot title. Defaults to indicator name.
mode	Character. "light" (default) or "dark" theme.

Value

A ggplot2 object.

Examples

```
df <- try(get_inkar_data("011", level = "KRE", lang = "en"))
if (is.data.frame(df) && nrow(df) > 0) {
  inkar_trends(df, regions = c("Berlin", "Hamburg", "München"))
}
```

plot_inkar *Plot INKAR Data on German Maps*

Description

Automatically projects regional INKAR data onto administrative boundaries of Germany. Supports Bund (BND), Bundeslaender (BLD), Kreise (KRE), Gemeindeverbaende (GVB) and Gemeinden (GEM). Boundaries come from Eurostat GISCO with the same territorial status as INKAR (see get_inkar_geometry()); they are downloaded once and cached. Alternatively, a custom sf geometry can be provided, for example for Raumordnungsregionen (ROR).

Usage

```
plot_inkar(
  data,
  variable = NULL,
  year = NULL,
  mode = c("light", "dark"),
  highlight = NULL,
  breaks = c("equal", "quantile"),
  title = NULL,
  geom = NULL
)
```

Arguments

data	A data frame returned by `get_inkar_data()`.
variable	Character. For wide-format data with multiple indicators, specify which indicator column to plot.
year	Integer/Character. If the data contains multiple years, specify which year to plot. If NULL and multiple years exist, the most recent year is plotted.
mode	Character. "light" (default) or "dark" theme.
highlight	Character vector. Region names (partial match) to highlight; all other regions are shown at reduced opacity.
breaks	Character. Color scale break method: `"equal"` (default) or `"quantile"` for five quantile classes, each with its own color step.
title	Character. Custom plot title. Defaults to the indicator name.
geom	Optional `sf` object (spatial data frame) to use for plotting. If supplied, no boundaries are downloaded, and the data is merged directly with this object.

Value

A `ggplot2` object displaying the mapped data.

`search_indicators` *Search Indicators and Print Results*

Description

Search for indicators by keyword. Prints a formatted table and invisibly returns the matches so you can copy the ID for use in `inkaR()`.

Usage

```
search_indicators(pattern, lang = c("de", "en"), theme = NULL)
```

Arguments

pattern	Text to search in names, IDs and descriptions. Matched literally and case-insensitively, so characters such as (need no escaping.
lang	Language to search in ("de" or "en").
theme	Optional character. Filter to a specific theme/domain before searching. Use get_themes() to list available themes.

Value

A filtered tibble of indicators (invisibly).

---

select_indicator *Interactively Select an Indicator*

Description

Opens a GUI selection list (e.g., in RStudio) to browse and pick an indicator. For code-based workflows, use inkaR("name") or search_indicators() instead.

Usage

```
select_indicator(pattern = NULL, lang = c("de", "en"))
```

Arguments

pattern	Optional character. Pre-filter the list by a keyword or regex. If NULL (default), the full indicator list is shown.
lang	Language for names: "de" (default) or "en".

Value

Character. The selected indicator ID, or NULL if cancelled.

---

select_level *Interactively Select a Spatial Level*

Description

Provides an interactive console menu to choose an INKAR spatial level. If a variable ID is provided, it probes the live API to find which levels actually have data for that indicator.

Usage

```
select_level(variable = NULL)
```

Arguments

variable	Optional character. The indicator ID to probe available levels.

Value

Character. The selected level ID, e.g., "KRE".

---

select_years *Interactively Select Years*

Description

Probes the API for available years for a specific indicator and level, then allows the user to select one or more years.

Usage

```
select_years(variable, level)
```

Arguments

variable	Indicator ID.
level	Spatial level ID.

Value

Character vector of years.

---

theme_inkaR *Premium ggplot2 theme for inkaR*

Description

Premium ggplot2 theme for inkaR

Usage

```
theme_inkaR(mode = c("light", "dark"), base_size = 11)
```

Arguments

mode	Character. "light" or "dark".
base_size	Numeric. Base font size.

update_indicators *Refresh Local Indicators Metadata*

Description

Checks the INKAR API for indicators that are not in the local `indicators` dataset. The check is informational only; install a newer package version to add them to the local metadata. New indicators can still be downloaded by their numeric M_ID with `get_inkar_data()`.

Usage

```
update_indicators(lang = c("de", "en"))
```

Arguments

`lang`	Character. Language for messages: `"de"` or `"en"`. Only indicators for German spatial levels are compared; the API's European indicators (NUTS-2) are not supported by this package.

Value

Invisibly returns the local `indicators` tibble. Indicators the API offers but the local metadata lacks are attached as attribute `"new_indicators"` (`M_ID`, `Name`, `Bereich`); active local indicators the API no longer offers as `"removed_indicators"`.

view_indicators *View Indicators in RStudio Viewer*

Description

Opens the available indicators in the RStudio data viewer for easy filtering and searching.

Usage

```
view_indicators(lang = c("de", "en"), theme = NULL)
```

Arguments

`lang`	Language code: "de" (German) or "en" (English).
`theme`	Optional character. Filter to a specific theme/domain before opening the viewer. Use `get_themes()` to list available themes.

Value

Invokes `View()` on the data frame.

Index

∗ **datasets**
indicators, 9

clear_inkar_cache, 4
clear_inkar_cache(), *8*
compare_district *(*compare_districts*)*, 4
compare_districts, 4
compare_region *(*compare_regions*)*, 5
compare_regions, 5
compare_regions(), *4*

get_geographies, 5
get_geographies(), *3*
get_indicators, 6
get_inkar_data, 6
get_inkar_data(), *2*, *4, 5*, *7*, *9–12*, *16*
get_inkar_geometry, 7
get_inkar_geometry(), *12*
get_themes, 8
get_themes(), *9*, *14*, *16*

indicators, 9
inkaR, 10
inkar *(*inkar_shortcut*)*, 11
inkar(), *10*
inkaR-package, 2
inkar_shortcut, 11
inkar_trends, 12

plot_inkar, 12

search_indicators, 13
search_indicators(), *3*, *8*
select_indicator, 14
select_level, 14
select_years, 15

theme_inkaR, 15

update_indicators, 16

view_indicators, 16
view_indicators(), *2*, *8*