

# Package: CatastRoEus (via r-universe)

October 15, 2024

**Title** Interface to the 'INSPIRE' services of 'Catastro de Bizkaia',  
'Catastro de Gipuzkoa', 'Catastro de Araba'

**Version** 0.0.0.9000

**Description** Access public spatial data available under the 'INSPIRE'  
directive. Tools for downloading references, buildings and  
addresses of properties on Pais Vasco (Spain).

**License** MIT + file LICENSE

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.1

**URL** <https://github.com/rOpenSpain/CatastRoEus>

**BugReports** <https://github.com/rOpenSpain/CatastRoEus/issues>

**Depends** R (>= 3.6)

**Imports** tidygeocoder, tidyterra, rappdirs, sf (>= 1.0.0), stringi,  
tibble, xml2, png, curl, slippyath, terra, utils, dplyr,  
magrittr

**Suggests** CatastRo, CatastRoNav, ggplot2, knitr, mapSpain, rmarkdown,  
testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Repository** <https://ropenspain.r-universe.dev>

**RemoteUrl** <https://github.com/rOpenSpain/CatastRoEus>

**RemoteRef** HEAD

**RemoteSha** cfb9eed0c4e9f74dfe0695c1f66931ff8279523c

## Contents

|  |   |
|--|---|
| catreus_bizk_atom_get_addresses . . . . .        | 2 |
| catreus_bizk_atom_get_addresses_db_all . . . . . | 3 |
| catreus_bizk_atom_get_buildings . . . . .        | 4 |
| catreus_bizk_atom_get_buildings_db_all . . . . . | 6 |

|                                     |    |
|-------------------------------------|----|
| catreus_bizk_atom_get_parcel        | 7  |
| catreus_bizk_atom_get_parcel_db_all | 8  |
| catreus_clear_cache                 | 9  |
| catreus_set_cache_dir               | 10 |
| catreus_wfs_get_address_bbox        | 11 |
| catreus_wfs_get_buildings_bbox      | 13 |
| catreus_wfs_get_parcel_bbox         | 15 |
| catreus_wms_get_layer               | 17 |

## Index 21

---

catreus\_bizk\_atom\_get\_addresses

*ATOM INSPIRE: Download all the addresses of a municipality in Bizkaia*

---

### Description

Get the spatial data of all the addresses belonging to a single municipality using the INSPIRE ATOM service.

### Usage

```
catreus_bizk_atom_get_addresses(
  munic,
  cache = TRUE,
  update_cache = FALSE,
  cache_dir = NULL,
  verbose = FALSE
)
```

### Arguments

|              |  |
|--------------|--|
| munic        | Municipality to extract, It can be a part of a string or the addresses code. See <a href="#">catreus_bizk_atom_get_addresses_db_all</a> .          |
| cache        | A logical whether to do caching. Default is TRUE. See <b>About caching</b> section on <a href="#">catreus_set_cache_dir()</a> .                    |
| update_cache | A logical whether to update cache. Default is FALSE. When set to TRUE it would force a fresh download of the source file.                          |
| cache_dir    | A path to a cache directory. On missing value the function would store the cached files on a temporary dir (See <a href="#">base::tempdir()</a> ). |
| verbose      | Logical, displays information. Useful for debugging, default is FALSE.   |

### Value

A [sf](#) object.

## References

[INSPIREBIZKAIA](#)

## See Also

Other ATOM: [catreus\\_bizk\\_atom\\_get\\_addresses\\_db\\_all\(\)](#), [catreus\\_bizk\\_atom\\_get\\_buildings\(\)](#), [catreus\\_bizk\\_atom\\_get\\_buildings\\_db\\_all\(\)](#), [catreus\\_bizk\\_atom\\_get\\_parcel\(\)](#), [catreus\\_bizk\\_atom\\_get\\_parcel\\_db\\_all\(\)](#)

Other addresses: [catreus\\_bizk\\_atom\\_get\\_addresses\\_db\\_all\(\)](#)

## Examples

```
s <- catreus_bizk_atom_get_addresses("ABADIÑO")

library(ggplot2)

ggplot(s) +
  geom_sf() +
  labs(
    title = "Addresses Zoning",
    subtitle = "ABADIÑO"
  )
```

---

catreus\_bizk\_atom\_get\_addresses\_db\_all

*ATOM INSPIRE: Reference database for ATOM Addresses in Bizkaia*

---

## Description

Create a database containing the urls provided in the INSPIRE ATOM service for extracting Addresses.

## Usage

```
catreus_bizk_atom_get_addresses_db_all(
  cache = TRUE,
  update_cache = FALSE,
  cache_dir = NULL,
  verbose = FALSE
)
```

## Arguments

|              |   |
|--------------|---|
| cache        | A logical whether to do caching. Default is TRUE. See <b>About caching</b> section on <a href="#">catreus_set_cache_dir()</a> . |
| update_cache | A logical whether to update cache. Default is FALSE. When set to TRUE it would force a fresh download of the source file.       |

|           |   |
|-----------|---|
| cache_dir | A path to a cache directory. On missing value the function would store the cached files on a temporary dir (See <code>base::tempdir()</code> ). |
| verbose   | Logical, displays information. Useful for debugging, default is FALSE.  |

### Value

A `tibble` with the information requested:

- `munic`: Name of the municipality.
- `url`: url for downloading information of the corresponding municipality.
- `date`: Reference date of the data.

### Source

**INSPIREBIZKAIA**

### See Also

Other ATOM: `catreus_bizk_atom_get_addresses()`, `catreus_bizk_atom_get_buildings()`, `catreus_bizk_atom_get_buildings_db_all()`, `catreus_bizk_atom_get_parcel()`, `catreus_bizk_atom_get_parcel_db_all()`

Other addresses: `catreus_bizk_atom_get_addresses()`

### Examples

```
catreus_bizk_atom_get_addresses_db_all()
```

---

```
catreus_bizk_atom_get_buildings
```

*ATOM INSPIRE: Download all the buildings of a municipality in Bizkaia*

---

### Description

Get the spatial data of all the buildings belonging to a single municipality using the INSPIRE ATOM service.

### Usage

```
catreus_bizk_atom_get_buildings(  
  munic,  
  cache = TRUE,  
  update_cache = FALSE,  
  cache_dir = NULL,  
  verbose = FALSE  
)
```

## Arguments

|                           |  |
|---------------------------|--|
| <code>munic</code>        | Municipality to extract, It can be a part of a string or the buildings code. See <a href="#">catreus_bizk_atom_get_buildings_db_all</a> .          |
| <code>cache</code>        | A logical whether to do caching. Default is TRUE. See <b>About caching</b> section on <a href="#">catreus_set_cache_dir()</a> .                    |
| <code>update_cache</code> | A logical whether to update cache. Default is FALSE. When set to TRUE it would force a fresh download of the source file.                          |
| <code>cache_dir</code>    | A path to a cache directory. On missing value the function would store the cached files on a temporary dir (See <a href="#">base::tempdir()</a> ). |
| <code>verbose</code>      | Logical, displays information. Useful for debugging, default is FALSE.   |

## Value

A `sf` object.

## References

**INSPIREBIZKAIA**

## See Also

Other ATOM: [catreus\\_bizk\\_atom\\_get\\_addresses\(\)](#), [catreus\\_bizk\\_atom\\_get\\_addresses\\_db\\_all\(\)](#), [catreus\\_bizk\\_atom\\_get\\_buildings\\_db\\_all\(\)](#), [catreus\\_bizk\\_atom\\_get\\_parcel\(\)](#), [catreus\\_bizk\\_atom\\_get\\_parcel\\_db\\_all\(\)](#)

Other buildings: [catreus\\_bizk\\_atom\\_get\\_buildings\\_db\\_all\(\)](#)

## Examples

```
s <- catreus_bizk_atom_get_buildings("ABADIÑO")

library(ggplot2)

ggplot(s) +
  geom_sf() +
  labs(
    title = "Buildings Zoning",
    subtitle = "ABADIÑO"
  )
```

---

`catreus_bizk_atom_get_buildings_db_all`*ATOM INSPIRE: Reference database for ATOM Buildings in Bizkaia*

---

### Description

Create a database containing the urls provided in the INSPIRE ATOM service for extracting Buildings.

### Usage

```
catreus_bizk_atom_get_buildings_db_all(  
  cache = TRUE,  
  update_cache = FALSE,  
  cache_dir = NULL,  
  verbose = FALSE  
)
```

### Arguments

|                           |  |
|---------------------------|--|
| <code>cache</code>        | A logical whether to do caching. Default is TRUE. See <b>About caching</b> section on <a href="#">catreus_set_cache_dir()</a> .                    |
| <code>update_cache</code> | A logical whether to update cache. Default is FALSE. When set to TRUE it would force a fresh download of the source file.                          |
| <code>cache_dir</code>    | A path to a cache directory. On missing value the function would store the cached files on a temporary dir (See <a href="#">base::tempdir()</a> ). |
| <code>verbose</code>      | Logical, displays information. Useful for debugging, default is FALSE.   |

### Value

A [tibble](#) with the information requested:

- `munic`: Name of the municipality.
- `url`: url for downloading information of the corresponding municipality.
- `date`: Reference date of the data.

### Source

**INSPIREBIZKAIA**

### See Also

Other ATOM: [catreus\\_bizk\\_atom\\_get\\_addresses\(\)](#), [catreus\\_bizk\\_atom\\_get\\_addresses\\_db\\_all\(\)](#), [catreus\\_bizk\\_atom\\_get\\_buildings\(\)](#), [catreus\\_bizk\\_atom\\_get\\_parcel\(\)](#), [catreus\\_bizk\\_atom\\_get\\_parcel\\_db\\_all\(\)](#)  
Other buildings: [catreus\\_bizk\\_atom\\_get\\_buildings\(\)](#)

## Examples

```
catreus_bizk_atom_get_buildings_db_all()
```

---

```
catreus_bizk_atom_get_parcel
```

*ATOM INSPIRE: Download all the cadastral parcels of a municipality*

---

## Description

Get the spatial data of all the cadastral parcels belonging to a single municipality using the INSPIRE ATOM service.

## Usage

```
catreus_bizk_atom_get_parcel(  
    munic,  
    cache = TRUE,  
    update_cache = FALSE,  
    cache_dir = NULL,  
    verbose = FALSE  
)
```

## Arguments

|              |  |
|--------------|--|
| munic        | Municipality to extract, It can be a part of a string or the cadastral code. See <a href="#">catreus_bizk_atom_get_parcel_db_all</a> .             |
| cache        | A logical whether to do caching. Default is TRUE. See <b>About caching</b> section on <a href="#">catreus_set_cache_dir()</a> .                    |
| update_cache | A logical whether to update cache. Default is FALSE. When set to TRUE it would force a fresh download of the source file.                          |
| cache_dir    | A path to a cache directory. On missing value the function would store the cached files on a temporary dir (See <a href="#">base::tempdir()</a> ). |
| verbose      | Logical, displays information. Useful for debugging, default is FALSE.   |

## Value

A `sf` object.

## References

**INSPIREBIZKAIA**

**See Also**

Other ATOM: [catreus\\_bizk\\_atom\\_get\\_addresses\(\)](#), [catreus\\_bizk\\_atom\\_get\\_addresses\\_db\\_all\(\)](#), [catreus\\_bizk\\_atom\\_get\\_buildings\(\)](#), [catreus\\_bizk\\_atom\\_get\\_buildings\\_db\\_all\(\)](#), [catreus\\_bizk\\_atom\\_get\\_pa](#)

Other parcels: [catreus\\_bizk\\_atom\\_get\\_parcel\\_db\\_all\(\)](#)

**Examples**

```
s <- catreus_bizk_atom_get_parcel("ABANTO Y CIERVANA")

library(ggplot2)

ggplot(s) +
  geom_sf() +
  labs(
    title = "Cadastral Zoning",
    subtitle = "ABANTO Y CIERVANA"
  )
```

---

catreus\_bizk\_atom\_get\_parcel\_db\_all

*ATOM INSPIRE: Reference database for ATOM cadastral parcels*

---

**Description**

Create a database containing the urls provided in the INSPIRE ATOM service for extracting Cadastral Parcels.

**Usage**

```
catreus_bizk_atom_get_parcel_db_all(
  cache = TRUE,
  update_cache = FALSE,
  cache_dir = NULL,
  verbose = FALSE
)
```

**Arguments**

|              |  |
|--------------|--|
| cache        | A logical whether to do caching. Default is TRUE. See <b>About caching</b> section on <a href="#">catreus_set_cache_dir()</a> .                    |
| update_cache | A logical whether to update cache. Default is FALSE. When set to TRUE it would force a fresh download of the source file.                          |
| cache_dir    | A path to a cache directory. On missing value the function would store the cached files on a temporary dir (See <a href="#">base::tempdir()</a> ). |
| verbose      | Logical, displays information. Useful for debugging, default is FALSE.   |

**Value**

A [tibble](#) with the information requested:

- `munic`: Name of the municipality.
- `url`: url for downloading information of the corresponding municipality.
- `date`: Reference date of the data.

**Source**

**INSPIREBIZKAIA**

**See Also**

Other ATOM: [catreus\\_bizk\\_atom\\_get\\_addresses\(\)](#), [catreus\\_bizk\\_atom\\_get\\_addresses\\_db\\_all\(\)](#), [catreus\\_bizk\\_atom\\_get\\_buildings\(\)](#), [catreus\\_bizk\\_atom\\_get\\_buildings\\_db\\_all\(\)](#), [catreus\\_bizk\\_atom\\_get\\_pa](#)

Other parcels: [catreus\\_bizk\\_atom\\_get\\_parcel](#)s()

**Examples**

```
catreus_bizk_atom_get_parcel
```

s\_db\_all()

---

`catreus_clear_cache`    *Clear your **CatastRoEus** cache dir*

---

**Description**

**Use this function with caution.** This function would clear your cached data and configuration, specifically:

- Deletes the **CatastRoEus** config directory (`rappdirs::user_config_dir("CatastRoEus", "R")`).
- Deletes the `cache_dir` directory.
- Deletes the values on stored on `Sys.getenv("CATASTROEUS_CACHE_DIR")`.

**Usage**

```
catreus_clear_cache(config = FALSE, cached_data = TRUE, verbose = FALSE)
```

**Arguments**

|                          |   |
|--------------------------|---|
| <code>config</code>      | if TRUE, will delete the configuration folder of <b>CatastRoEus</b> .                   |
| <code>cached_data</code> | If this is set to TRUE, it will delete your <code>cache_dir</code> and all its content. |
| <code>verbose</code>     | Logical, displays information. Useful for debugging, default is FALSE.                  |

## Details

This is an overkill function that is intended to reset your status as if you would never have installed and/or used **CatastRoEus**.

## Value

Invisible. This function is called for its side effects.

## See Also

Other cache utilities: [catreus\\_set\\_cache\\_dir\(\)](#)

## Examples

```
# Don't run this! It would modify your current state
## Not run:
catreus_clear_cache(verbose = TRUE)

## End(Not run)

Sys.getenv("CATASTROEUS_CACHE_DIR")
```

---

`catreus_set_cache_dir` *Set your **CatastRoEus** cache dir*

---

## Description

This function will store your `cache_dir` path on your local machine and would load it for future sessions. Type `Sys.getenv("CATASTROEUS_CACHE_DIR")` to find your cached path.

Alternatively, you can store the `cache_dir` manually with the following options:

- Run `Sys.setenv(CATASTROEUS_CACHE_DIR = "cache_dir")`. You would need to run this command on each session (Similar to `install = FALSE`).
- Write this line on your `.Renviron` file: `CATASTROEUS_CACHE_DIR = "value_for_cache_dir"` (same behavior than `install = TRUE`). This would store your `cache_dir` permanently.

## Usage

```
catreus_set_cache_dir(
  cache_dir,
  overwrite = FALSE,
  install = FALSE,
  verbose = TRUE
)
```

**Arguments**

|           |  |
|-----------|--|
| cache_dir | A path to a cache directory. On missing value the function would store the cached files on a temporary dir (See <code>base::tempdir()</code> ).                        |
| overwrite | If this is set to TRUE, it will overwrite an existing CATASTROEUS_CACHE_DIR that you already have in local machine.  |
| install   | if TRUE, will install the key in your local machine for use in future sessions. Defaults to FALSE. If cache_dir is FALSE this parameter is set to FALSE automatically. |
| verbose   | Logical, displays information. Useful for debugging, default is FALSE.   |

**Value**

An (invisible) character with the path to your cache\_dir.

**About caching**

Sometimes cached files may be corrupt. On that case, try re-downloading the data setting `update_cache = TRUE`.

If you experience any problem on download, try to download the corresponding file by any other method and save it on your cache\_dir. Use the option `verbose = TRUE` for debugging the API query.

**See Also**

`rappdirs::user_config_dir()`

Other cache utilities: `catreus_clear_cache()`

**Examples**

```
# Don't run this! It would modify your current state
## Not run:
catreus_set_cache_dir(verbose = TRUE)

## End(Not run)

Sys.getenv("CATASTROEUS_CACHE_DIR")
```

---

catreus\_wfs\_get\_address\_bbox

*Retrieve Address Data in Pais Vasco Based on Bounding Box Coordinates*

---

## Description

This function fetches addresses data within a specified bounding box. It first checks if the provided coordinates are valid, determines the province based on these coordinates using reverse geocoding, and fetches address data for Bizkaia or Gipuzkoa accordingly. It supports transformations between geographical and UTM coordinate systems, handles multiple Coordinate Reference Systems (CRS), and manages different minimum counts for returned records.

## Usage

```
catreus_wfs_get_address_bbox(x, srs = NULL, verbose = FALSE, count = NULL)
```

## Arguments

|         |  |
|---------|--|
| x       | Bounding box coordinates or a spatial object, which could be: <ul style="list-style-type: none"><li>• A numeric vector of length 4 with the coordinates defining the bounding box: <code>c(latitude1, longitude1, latitude2, longitude2)</code>.</li><li>• An <code>sf/sfc</code> object from the <code>sf</code> package.</li></ul> |
| srs     | Spatial Reference System (SRS) or Coordinate Reference System (CRS) code to be used in the query. For best results, ensure the coordinates are in ETRS89 (EPSG:25830) or WGS84 (EPSG:4326) when using latitude and longitude.  |
| verbose | Logical; if TRUE, additional information about function operations is printed. Useful for debugging. Default is FALSE.   |
| count   | Integer specifying the maximum number of address records to return, or NULL to use default settings of the API service.  |

## Details

This function uses reverse geocoding to determine the province within the Basque Country from the coordinates provided. Based on the province, it delegates the data fetching to specific functions handling each province's data. It supports flexible input types and handles geographical coordinate transformations internally if needed.

## Value

Depending on the input and geographic location, this function may return:

- A message detailing the success of the query and the province fetched.
- A warning message if the bounding box spans multiple provinces.
- An error message if the coordinates are not within Pais Vasco.

## See Also

[st\\_bbox](#), which is used to manage spatial bounding boxes.

## Examples

```
# Define bounding box coordinates for an urban location in Gipuzkoa
coords_gipuzkoa <- c(582745.070132, 4795611.169048, 584249.337348, 4796830.604835)

# Fetch address data using the bounding box
addresses_gipuzkoa <- catreus_wfs_get_address_bbox(coords_gipuzkoa, srs = 25830)

library(ggplot2)
# Plot the buildings data
ggplot(addresses_gipuzkoa) +
  geom_sf() + ggtitle("Addresses Data for Gipuzkoa")

# Define bounding box coordinates for a location in Bizkaia
coords_bizkaia <- c(499307.414680, 4792958.784686, 502508.961032, 4796815.771278)

# Fetch address data using the bounding box
addresses_bizkaia <- catreus_wfs_get_address_bbox(coords_bizkaia, srs = 25830)

library(ggplot2)
# Plot the buildings data
ggplot(addresses_bizkaia) +
  geom_sf() + ggtitle("Addresses Data for Bizkaia")
```

---

catreus\_wfs\_get\_buildings\_bbox

*Retrieve Building Data in Pais Vasco Based on Bounding Box Coordinates*

---

## Description

This function fetches buildings data within a specified bounding box. It first checks if the provided coordinates are valid, determines the province based on these coordinates using reverse geocoding, and fetches buildings data for Bizkaia, Gipuzkoa, or Araba/Álava accordingly. It supports transformations between geographical and UTM coordinate systems, handles multiple Coordinate Reference Systems (CRS), and manages different minimum counts for returned records.

## Usage

```
catreus_wfs_get_buildings_bbox(x, srs = NULL, verbose = FALSE, count = NULL)
```

## Arguments

- x                    Bounding box coordinates or a spatial object, which could be:
- A numeric vector of length 4 with the coordinates defining the bounding box: `c(latitude1, longitude1, latitude2, longitude2)`.

|                      |   |
|----------------------|---|
|                      | <ul style="list-style-type: none"> <li>• An <code>sf/sfc</code> object from the <b>sf</b> package.</li> </ul>   |
| <code>srs</code>     | Spatial Reference System (SRS) or Coordinate Reference System (CRS) code to be used in the query. For best results, ensure the coordinates are in ETRS89 (EPSG:25830) or WGS84 (EPSG:4326) when using latitude and longitude. |
| <code>verbose</code> | Logical; if TRUE, additional information about function operations is printed. Useful for debugging. Default is FALSE.  |
| <code>count</code>   | Integer specifying the maximum number of building records to return. Default is 10.   |

### Details

This function uses reverse geocoding to determine the province within the Basque Country from the coordinates provided. Based on the province, it delegates the data fetching to specific functions handling each province's data. It supports flexible input types and handles geographical coordinate transformations internally if needed.

### Value

Depending on the input and geographic location, this function may return:

- An `sf` object containing building data within the specified `bbox`.
- A message indicating mismatched or out-of-region coordinates.

### See Also

[st\\_bbox](#), which is used to manage spatial bounding boxes.

### Examples

```
library(mapSpain)
library(dplyr)
library(sf)

# Define bounding box coordinates for a location in Gipuzkoa
coords_gipuzkoa <- c(580335.961264, 4795197.149650, 581482.214894, 4795985.065492)

# Fetch building data using the bounding box
buildings_gipuzkoa <- catreus_wfs_get_buildings_bbox(coords_gipuzkoa, srs = 25830, count = 10)

library(ggplot2)
# Plot the buildings data
ggplot(buildings_gipuzkoa) +
  geom_sf() + ggtitle("Building Data for Gipuzkoa")

# Define bounding box coordinates for a location in Bizkaia
bilbao <- esp_get_capimun(munic = "Bilbao") %>%
  st_transform(25830) %>%
  st_buffer(300)
buildings_bilbao <- catreus_wfs_get_buildings_bbox(bilbao)
```

```

ggplot(buildings_bilbao) + geom_sf() + ggtitle("Buildings Data for Bilbao")

# Define bounding box coordinates for a rural location in Araba/Álava
coords_alaba = c(525858.205755, 4742911.412803, 526701.543389, 4743398.976145)

# Fetch building data using the bounding box, requesting more features
buildings_alaba <- catreus_wfs_get_buildings_bbox(coords_alaba, srs = 25830)

library(ggplot2)
# Plot the buildings data for Araba/Álava
ggplot(buildings_alaba) + geom_sf() + ggtitle("Building Data for Alaba")

```

---

```
catreus_wfs_get_parcel_bbox
```

*Retrieve Parcel Data in Pais Vasco Based on Bounding Box Coordinates*

---

## Description

This function fetches parcel data within a specified bounding box. It first checks if the provided coordinates are valid, determines the province based on these coordinates using reverse geocoding, and fetches parcel data for Bizkaia, Gipuzkoa, or Araba/Álava accordingly. It supports transformations between geographical and UTM coordinate systems, handles multiple Coordinate Reference Systems (CRS), and manages different minimum counts for returned records.

## Usage

```
catreus_wfs_get_parcel_bbox(x, srs = NULL, verbose = FALSE, count = NULL)
```

## Arguments

|         |   |
|---------|---|
| x       | Bounding box coordinates or a spatial object, which could be: <ul style="list-style-type: none"> <li>• A numeric vector of length 4 with the coordinates defining the bounding box: <code>c(latitude1, longitude1, latitude2, longitude2)</code>.</li> <li>• An <code>sf/sfc</code> object from the <b>sf</b> package.</li> </ul> |
| srs     | Spatial Reference System (SRS) or Coordinate Reference System (CRS) code to be used in the query. For best results, ensure the coordinates are in ETRS89 (EPSG:25830) or WGS84 (EPSG:4326) when using latitude and longitude.   |
| verbose | Logical; if TRUE, additional information about function operations is printed. Useful for debugging. Default is FALSE.  |
| count   | Integer specifying the maximum number of parcel records to return. Default is 10.   |

## Details

The function uses reverse geocoding to identify the province within the Basque Country. It delegates data fetching to province-specific functions, which handle data extraction based on the provided geographic bounds. The function adapts dynamically to coordinate system inputs and provides options for detailed logging to facilitate debugging and verification of operations.

## Value

Depending on the input and geographic location, this function may return:

- A message detailing the success of the query and the province fetched.
- A warning message if the bounding box spans multiple provinces.
- An error message if the coordinates are not within Pais Vasco.

## See Also

[st\\_bbox](#), which is used to manage spatial bounding boxes.

## Examples

```
library(mapSpain)
library(dplyr)
library(sf)

# Define bounding box coordinates for an urban location in Gipuzkoa
coords_gipuzkoa <- c(582745.070132, 4795611.169048, 584249.337348, 4796830.604835)

# Fetch parcel data using the bounding box
parcels_gipuzkoa <- catreus_wfs_get_parcels_bbox(coords_gipuzkoa, srs = 25830)

library(ggplot2)
# Plot the parcels data
ggplot(parcels_gipuzkoa) +
  geom_sf() + ggtitle("Parcels Data for Gipuzkoa")

# Define bounding box coordinates for a location in Bizkaia
coords_bizkaia <- c(504218.816311, 4788948.595082, 505374.746026, 4789719.963173)

# Fetch parcel data using the bounding box
parcels_bizkaia <- catreus_wfs_get_parcels_bbox(coords_bizkaia, srs = 25830)

library(ggplot2)
# Plot the parcels data for Bizkaia
ggplot(parcels_bizkaia) +
  geom_sf() + ggtitle("Parcels Data for Bizkaia")

# Define bounding box coordinates for a location in Araba

# Fetch parcel data using the bounding box
```

```
vitoria <- esp_get_capimun(munic = "Vitoria") %>%
  st_transform(4326) %>%
  # Small buffer of 200 m
  st_buffer(300)

parcels_araba <- catreus_wfs_get_parcels_bbox(vitoria)
ggplot(parcels_araba) + geom_sf() + ggtitle("Parcels Data for Vitoria")
```

---

catreus\_wms\_get\_layer *WMS INSPIRE: Retrieve Map Images from the Basque Country (Pais Vasco) Based on Bounding Box Coordinates*

---

## Description

Fetches geotagged images from the Cadastre based on the provided bounding box coordinates. This function determines the province (Bizkaia, Gipuzkoa, or Araba/Álava) using reverse geocoding and fetches the appropriate data using specific WMS service functions for each province. It ensures the coordinates fall within a single province and handles different CRS inputs.

## Usage

```
catreus_wms_get_layer(
  x,
  srs = NULL,
  what = c("parcel", "admunit", "admbound", "building", "zoning", "address", "buother"),
  id = "change_id",
  styles = NULL,
  verbose = FALSE,
  crop = FALSE,
  options = list()
)
```

## Arguments

- |                   |   |
|-------------------|---|
| <code>x</code>    | <p>A numeric vector of bbox coordinates or an sf object defining the area to be covered.</p> <ul style="list-style-type: none"> <li>A numeric vector of length 4 with the coordinates defining the bounding box: <code>c(xmin, ymin, xmax, ymax)</code>.</li> </ul> |
| <code>srs</code>  | The spatial reference system to be used, typically an EPSG code (3857 or 25830).  |
| <code>what</code> | <p>Specifies the layer to be extracted. Options include:</p> <p>For Bizkaia:</p> <ul style="list-style-type: none"> <li>"parcel": CP.CadastralParcel</li> <li>"zoning": CP.CadastralZoning</li> </ul>   |

- "address": AD.Address
- "admunit": AU.AdministrativeUnit

For Gipuzkoa:

- "parcel": cp.CadastralParcel
- "zoning": cp.CadastralZoning
- "address": ad.Address
- "admunit": au.AdministrativeUnit
- "admbound": au.AdministrativeBoundary
- "building": bu.building
- "buother": bu.otherconstruction

For Araba:

- "parcel": CP.CadastralParcel
- "admunit": AU.AdministrativeUnit
- "admbound": AU.AdministrativeBoundary
- "building": BU.Building

|         |  |
|---------|--|
| id      | An identifier for the custom WMS service configuration.  |
| styles  | Specifies the style of the WMS layer, see <b>Styles</b> for options.   |
| verbose | Logical; if TRUE, prints additional information about function operations. Useful for debugging. Default is FALSE. |
| crop    | Logical; if TRUE, the returned image will be cropped to the bbox. Default is FALSE.                                |
| options | List; additional options for the WMS request.  |

## Details

This function uses reverse geocoding to determine the province within the Basque Country from the provided coordinates. Based on the identified province, it delegates the data fetching to specific functions handling each province's data. It supports flexible input types and handles geographical coordinate transformations internally if needed.

## Value

Returns a [SpatRaster](#) object with either RGB or RGBA layers depending on the data. Additionally, prints messages detailing the province detected and any potential issues with the provided coordinates.

## Styles

Various styles are available for each layer, which can enhance the visualization:

Bizkaia:

- "parcel": "CP.CadastralParcel.BoundariesOnly", "CP.CadastralParcel.ReferencePointOnly", "CP.CadastralParcel.LabelOnReferencePoint", "CP.CadastralParcel.Default", "inspire\_common:DEFAULT"
- "zoning": "CP.CadastralZoning.Default", "inspire\_common:DEFAULT"

- "address": "AD.Address.Default", "inspire\_common:DEFAULT"
- "admunit": "AU.AdministrativeUnit.Default", "inspire\_common:DEFAULT"

#### Gipuzkoa:

- "parcel": "CP.CadastralParcel.BoundariesOnly", "CP.CadastralParcel.ReferencePointOnly", "CP.CadastralParcel.LabelOnReferencePoint", "CP.CadastralParcel.Default", "CP.CadastralParcel.ELFCadastre"
- "zoning": "CP.CadastralZoning.Default", "CP.CadastralZoning.ELFCadastre"
- "address": "AD.Address.Default", "AD.Address.Number.ELFCadastre"
- "admunit": "AU.AdministrativeUnit.Default", "AU.AdministrativeUnit.ELFCadastre"
- "admbound": "AU.AdministrativeBoundary.Default", "AU.AdministrativeBoundary.ELFCadastre"
- "building": "BU.Building.Default", "BU.Building.ELFCadastre"
- "buother": "BU.OtherConstruction.Default", "BU.OtherConstruction.ELFCadastre"

#### Araba:

- "parcel": "INSPIRE\_CP:CP.CadastralParcel.Default", "INSPIRE\_CP:CP.CadastralParcel.ELFCadastre", "INSPIRE\_CP:CP.CadastralParcel.BoundariesOnly", "INSPIRE\_CP:CP.CadastralParcel.LabelOnReferencePoint", "INSPIRE\_CP:CP.CadastralParcel.ReferencePointOnly"
- "admunit": "INSPIRE\_AU:AU.AdministrativeUnit.Default"
- "admbound": "INSPIRE\_AU:AU.AdministrativeBoundary.Default"
- "building": "INSPIRE\_BU:BU.Building.Default"

#### See Also

[mapSpain::esp\\_getTiles\(\)](#), [mapSpain::esp\\_make\\_provider\(\)](#), [terra::RGB\(\)](#). For advanced plotting, see [terra::plotRGB\(\)](#) and [tidyterra::geom\\_spatraster\\_rgb\(\)](#).

#### Examples

```
library(mapSpain)
library(ggplot2)
library(terra)
library(tidyterra)

# Fetching a building layer using specific bounding box coordinates (Araba)
pic_bu <- catreus_wms_get_layer(c(-298730.238481,5288011.551711,-296360.690604,5289922.477418),
                               srs = 3857, what = "building", id = "layer_eus_1")

# Displaying the fetched raster image
ggplot() + geom_spatraster_rgb(data = pic_bu)

# Fetching an address layer using specific bounding box coordinates (Gipuzkoa)
pic_ad <- catreus_wms_get_layer(c(-221868.029226,5358914.061417,-220750.137688,5360872.760267),
                               srs= 3857, what = "address", id = "layer_eus_2")

# Displaying the fetched raster image
ggplot() + geom_spatraster_rgb(data = pic_ad)
```

```
# Fetching a zoning layer using specific bounding box coordinates (Bizkaia)
pic_zo <- catreus_wms_get_layer(c(-339724.372213,5356554.068169,-333915.158064,5361598.912035),
                               srs= 3857, what = "zoning", id = "layer_eus_3")

# Displaying the fetched raster image
ggplot() + geom_spatraster_rgb(data = pic_zo)
```

# Index

- \* **ATOM**
    - catreus\_bizk\_atom\_get\_addresses, [2](#)
    - catreus\_bizk\_atom\_get\_addresses\_db\_all, [3](#)
    - catreus\_bizk\_atom\_get\_buildings, [4](#)
    - catreus\_bizk\_atom\_get\_buildings\_db\_all, [6](#)
    - catreus\_bizk\_atom\_get\_parcel, [7](#)
    - catreus\_bizk\_atom\_get\_parcel\_db\_all, [8](#)
  - \* **INSPIRE**
    - catreus\_wms\_get\_layer, [17](#)
  - \* **WMS**
    - catreus\_wms\_get\_layer, [17](#)
  - \* **addresses**
    - catreus\_bizk\_atom\_get\_addresses, [2](#)
    - catreus\_bizk\_atom\_get\_addresses\_db\_all, [3](#)
  - \* **buildings**
    - catreus\_bizk\_atom\_get\_buildings, [4](#)
    - catreus\_bizk\_atom\_get\_buildings\_db\_all, [6](#)
  - \* **cache utilities**
    - catreus\_clear\_cache, [9](#)
    - catreus\_set\_cache\_dir, [10](#)
  - \* **parcels**
    - catreus\_bizk\_atom\_get\_parcel, [7](#)
    - catreus\_bizk\_atom\_get\_parcel\_db\_all, [8](#)
  - \* **spatial data layers from Basque Country (Pais Vasco)**
    - catreus\_wms\_get\_layer, [17](#)
- base::tempdir(), [2](#), [4-8](#), [11](#)
- catreus\_bizk\_atom\_get\_addresses, [2](#), [4-6](#), [8](#), [9](#)
- catreus\_bizk\_atom\_get\_addresses\_db\_all, [2](#), [3](#), [3](#), [5](#), [6](#), [8](#), [9](#)
- catreus\_bizk\_atom\_get\_buildings, [3](#), [4](#), [4](#), [6](#), [8](#), [9](#)
- catreus\_bizk\_atom\_get\_buildings\_db\_all, [3-5](#), [6](#), [8](#), [9](#)
- catreus\_bizk\_atom\_get\_parcel, [3-6](#), [7](#), [9](#)
- catreus\_bizk\_atom\_get\_parcel\_db\_all, [3-8](#), [8](#)
- catreus\_clear\_cache, [9](#), [11](#)
- catreus\_set\_cache\_dir, [10](#), [10](#)
- catreus\_set\_cache\_dir(), [2](#), [3](#), [5-8](#)
- catreus\_wfs\_get\_address\_bbox, [11](#)
- catreus\_wfs\_get\_buildings\_bbox, [13](#)
- catreus\_wfs\_get\_parcel\_bbox, [15](#)
- catreus\_wms\_get\_layer, [17](#)
- mapSpain::esp\_getTiles(), [19](#)
- mapSpain::esp\_make\_provider(), [19](#)
- rappdirs::user\_config\_dir(), [11](#)
- sf, [2](#), [5](#), [7](#)
- SpatRaster, [18](#)
- st\_bbox, [12](#), [14](#), [16](#)
- terra::plotRGB(), [19](#)
- terra::RGB(), [19](#)
- tibble, [4](#), [6](#), [9](#)
- tidyterra::geom\_spatraster\_rgb(), [19](#)