

Package: dtmapi (via r-universe)

October 9, 2026

Title Fetching Data from the 'Displacement Tracking Matrix'

Version 0.1.0.9000

Description Allows humanitarian community, academia, media, government, and non-governmental organizations to utilize the data collected by the 'Displacement Tracking Matrix' (<<https://dtm.iom.int>>), a unit in the International Organization for Migration. This also provides non-sensitive Internally Displaced Person figures, aggregated at the country, Admin 1 (states, provinces, or equivalent), and Admin 2 (smaller administrative areas) levels.

URL <https://github.com/Displacement-Tracking-Matrix/dtmapi-R>,
<https://displacement-tracking-matrix.github.io/dtmapi-R/>

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

Depends R (>= 4.1.0)

Imports htr2, testthat (>= 3.1.0), askpass

Suggests knitr, rmarkdown, withr

VignetteBuilder knitr

BugReports <https://github.com/Displacement-Tracking-Matrix/dtmapi-R/issues>

Config/roxygen2/version 8.0.0

Config/testthat/edition 3

Config/pak/sysreqs cmake make libuv1-dev libssl-dev

Repository <https://displacement-tracking-matrix.r-universe.dev>

Date/Publication 2026-10-09 16:26:50 UTC

RemoteUrl <https://github.com/displacement-tracking-matrix/dtmapi-r>

RemoteRef HEAD

RemoteSha 9a8b6ee27a7b4a95860612f134ca69d4a886fbc

Contents

baseline_get_admin0	2
baseline_get_admin1	3
baseline_get_admin2	4
baseline_get_countries	5
baseline_get_operations	6
baseline_get_subscription_key	6
baseline_set_subscription_key	7
hna_get_admin2	8
hna_get_catalog	9
hna_get_countries	9
hna_get_subscription_key	10
hna_set_subscription_key	10
Index	12

baseline_get_admin0	<i>Fetch IDP Admin0 Data</i>
---------------------	------------------------------

Description

Retrieve IDP data at Admin 0 level based on specified parameters. At least one of the following parameters must be provided: operation, admin0_name, or admin0_pcode.

Usage

```
baseline_get_admin0(  
    operation = NULL,  
    admin0_name = NULL,  
    admin0_pcode = NULL,  
    from_reporting_date = NULL,  
    to_reporting_date = NULL,  
    from_round_number = 0,  
    to_round_number = 0  
)
```

Arguments

- | | |
|---------------------|---|
| operation | Optional; Name of the DTM operation for which the data was collected. |
| admin0_name | Optional; Name of the country where the data was collected. |
| admin0_pcode | Optional; Country code (ISO 3166-1 alpha-3). |
| from_reporting_date | Optional; Start date for the reporting period (format: 'YYYY-MM-DD'). |
| to_reporting_date | Optional; End date for the reporting period (format: 'YYYY-MM-DD'). |

from_round_number
Optional; Starting round number for the data collection range.

to_round_number
Optional; Ending round number for the data collection range.

Value

A data frame containing the IDP Admin0 data matching the specified criteria.

Examples

```
# Fetch IDP data at Admin Level 0
idp_admin0_df <- baseline_get_admin0(admin0_name = "Ethiopia",
                                     from_round_number = 1,
                                     to_round_number = 10)

head(idp_admin0_df)
```

baseline_get_admin1	<i>Fetch IDP Admin1 Data</i>
---------------------	------------------------------

Description

Retrieve IDP data at Admin 1 level based on specified parameters. At least one of the following parameters must be provided: operation, admin0_name, or admin0_pcode.

Usage

```
baseline_get_admin1(
  operation = NULL,
  admin0_name = NULL,
  admin0_pcode = NULL,
  admin1_name = NULL,
  admin1_pcode = NULL,
  from_reporting_date = NULL,
  to_reporting_date = NULL,
  from_round_number = NULL,
  to_round_number = NULL
)
```

Arguments

operation	Optional; Name of the DTM operation for which the data was collected.
admin0_name	Optional; Name of the country where the data was collected.
admin0_pcode	Optional; Country code (ISO 3166-1 alpha-3).
admin1_name	Optional; Name of level 1 administrative boundaries.
admin1_pcode	Optional; Place code of level 1 administrative boundaries.

from_reporting_date
Optional; Start date for the reporting period (format: 'YYYY-MM-DD').

to_reporting_date
Optional; End date for the reporting period (format: 'YYYY-MM-DD').

from_round_number
Optional; Starting round number for the data collection range.

to_round_number
Optional; Ending round number for the data collection range.

Value

A data frame containing the IDP Admin1 data matching the specified criteria.

Examples

```
# Fetch IDP data at Admin Level 1
idp_admin1_df <- baseline_get_admin1(
  admin0_name = "Sudan",
  from_reporting_date = "2020-01-01",
  to_reporting_date = "2024-08-15"
)
head(idp_admin1_df)
```

baseline_get_admin2	<i>Fetch IDP Admin2 Data</i>
---------------------	------------------------------

Description

Retrieve IDP data at Admin 2 level based on specified parameters. At least one of the following parameters must be provided: operation, admin0_name, or admin0_pcode.

Usage

```
baseline_get_admin2(
  operation = NULL,
  admin0_name = NULL,
  admin0_pcode = NULL,
  admin1_name = NULL,
  admin1_pcode = NULL,
  admin2_name = NULL,
  admin2_pcode = NULL,
  from_reporting_date = NULL,
  to_reporting_date = NULL,
  from_round_number = 0,
  to_round_number = 0
)
```

Arguments

operation	Optional; Name of the DTM operation for which the data was collected.
admin0_name	Optional; Name of the country where the data was collected.
admin0_pcode	Optional; Country code (ISO 3166-1 alpha-3).
admin1_name	Optional; Name of level 1 administrative boundaries.
admin1_pcode	Optional; Place code of level 1 administrative boundaries.
admin2_name	Optional; Name of level 2 administrative boundaries.
admin2_pcode	Optional; Place code of level 2 administrative boundaries.
from_reporting_date	Optional; Start date for the reporting period (format: 'YYYY-MM-DD').
to_reporting_date	Optional; End date for the reporting period (format: 'YYYY-MM-DD').
from_round_number	Optional; Starting round number for the data collection range.
to_round_number	Optional; Ending round number for the data collection range.

Value

A data frame containing the IDP Admin2 data matching the specified criteria.

Examples

```
# Fetch IDP data at Admin Level 2
idp_admin2_df <- baseline_get_admin2(operation = "Yemen conflict", admin0_name = "Yemen")
head(idp_admin2_df)
```

baseline_get_countries

Fetch All Countries

Description

Retrieve all countries for which DTM data is publicly available through the API.

Usage

```
baseline_get_countries()
```

Value

A data frame containing the list of all countries.

Examples

```
countries_df <- baseline_get_countries()
head(countries_df)
```

```
baseline_get_operations
```

Fetch All Operations

Description

Retrieve all operations for which DTM data is publicly available through the API.

Usage

```
baseline_get_operations()
```

Value

A data frame containing the list of all operations.

Examples

```
# Fetch all operations
operations_df <- baseline_get_operations()
head(operations_df)
```

```
baseline_get_subscription_key
```

Retrieval of a Baseline API subscription key from the environment.

Description

The Baseline DTM API subscription key is returned, provided that it is available in the R session as an environment variable. Users will usually need to set the BASELINE_DTM_SUBSCRIPTION_KEY environment variable through a .Renv file (or other means) or by calling baseline_set_subscription_key().

Usage

```
baseline_get_subscription_key()
```

Details

On the other hand, if the TESTTHAT environment variable is true, indicating that unit tests are being run by the package maintainers, then the subscription key is returned through different means.

Value

A string representing a given subscription key for the DTM API.

Examples

```
## Not run:
# Generally, calling baseline_set_subscription_key() without the key as an
# argument is best, as the user can then be prompted to input the key without
# typing it directly into the console, making it more secure and less
# likely to be exposed.
baseline_set_subscription_key()

## End(Not run)
```

```
baseline_set_subscription_key
```

Set the user's Baseline API subscription key in order to make the API calls.

Description

The API subscription key will be stored as an environment variable named "BASELINE_DTM_SUBSCRIPTION_KEY".

Usage

```
baseline_set_subscription_key(..., key = NULL)
```

Arguments

...	Any argument here is ignored. It exists solely to discourage typing in the subscription key directly, e.g. <code>baseline_set_subscription_key("mysubscriptionkeyhere")</code> , since doing so will only result in the user being prompted to type the subscription key in a graphical user interface. Rather, if the user wishes to set the subscription key through the console / by programmatic means, then the user must explicitly pass the argument to the key parameter as a named parameter. E.g. <code>baseline_set_subscription_key(key = "mysubscriptionkeyhere")</code> .
key	Either NULL or a string representing the key. NULL is preferable: using it will prompt the user to type the subscription key in a graphical user interface that masks it.

Value

Nothing. Creates / overwrites an environment variable as a side effect.

Examples

```
## Not run:
# Generally, calling `baseline_set_subscription_key()` without the key as an
# argument is best, as the user can then be prompted to input the key without
# typing it directly into the console, making it more secure and less
# likely to exposed.
baseline_set_subscription_key()

## End(Not run)
```

hna_get_admin2	<i>Fetch HNA Admin2 Data</i>
----------------	------------------------------

Description

Retrieve Household Needs Assessment data at Admin 2 level based on specified parameters. At least one of the following parameters must be provided: admin0_name or admin0_pcode.

Usage

```
hna_get_admin2(
  admin0_name = NULL,
  admin0_pcode = NULL,
  population_group = NULL,
  year = NULL,
  page = NULL
)
```

Arguments

admin0_name	Optional; Name of the country where the data was collected.
admin0_pcode	Optional; Country code (ISO 3166-1 alpha-3).
population_group	Optional; A specific subpopulation (e.g. internally displaced persons).
year	Optional; Year of data collection.
page	Optional; Pagination, to fetch a certain chunk of the data.

Value

A list of one data frame and a sub-list. The former holds the requested data, and the latter is contains metadata on pagination.

Examples

```
# Fetch HNA data at Admin Level 2
hna_admin2 <- hna_get_admin2(admin0_name = "Mozambique", year = 2025)
head(hna_admin2)
```

hna_get_catalog	<i>Fetch Household Needs Assessment (HNA) Data Catalog</i>
-----------------	--

Description

Retrieve the HNA data catalog, which provides information on available indicators in the HNA API, including name, description, and data type.

Usage

```
hna_get_catalog()
```

Value

A data frame representing the data catalog for the HNA.

Examples

```
# Retrieve the HNA data catalog
operations_df <- hna_get_catalog()
head(operations_df)
```

hna_get_countries	<i>Fetch All Countries (HNA)</i>
-------------------	----------------------------------

Description

Retrieve all countries for which DTM Household Needs Assessment (HNA) data is publicly available through the API.

Usage

```
hna_get_countries()
```

Value

A data frame containing the list of all countries covered.

Examples

```
countries_df <- hna_get_countries()
head(countries_df)
```

`hna_get_subscription_key`*Retrieval of a HNA API subscription key from the environment.*

Description

The HNA DTM API subscription key is returned, provided that it is available in the R session as an environment variable. Users will usually need to set the HNA_DTM_SUBSCRIPTION_KEY environment variable through a .Renv file (or other means) or by calling `hna_set_subscription_key()`.

Usage

```
hna_get_subscription_key()
```

Details

On the other hand, if the TESTTHAT environment variable is true, indicating that unit tests are being run by the package maintainers, then the subscription key is returned through different means.

Value

A string representing a given subscription key for the DTM API.

Examples

```
## Not run:
# Generally, calling `hna_set_subscription_key()` without the key as an
# argument is best, as the user can then be prompted to input the key without
# typing it directly into the console, making it more secure and less
# likely to be exposed.
hna_set_subscription_key()

## End(Not run)
```

`hna_set_subscription_key`*Set the user's HNA API subscription key in order to make the API calls.*

Description

The API subscription key will be stored as an environment variable named "HNA_DTM_SUBSCRIPTION_KEY".

Usage

```
hna_set_subscription_key(..., key = NULL)
```

Arguments

...	Any argument here is ignored. It exists solely to discourage typing in the subscription key directly, e.g. <code>hna_set_subscription_key("mysubscriptionkeyhere")</code> , since doing so will only result in the user being prompted to type the subscription key in a graphical user interface. Rather, if the user wishes to set the subscription key through the console / by programmatic means, then the user must explicitly pass the argument to the key parameter as a named parameter. E.g. <code>hna_set_subscription_key(key = "mysubscriptionkeyhere")</code> .
key	Either NULL or a string representing the key. NULL is preferable: using it will prompt the user to type the subscription key in a graphical user interface that masks it.

Value

Nothing. Creates / overwrites an environment variable as a side effect.

Examples

```
## Not run:  
# Generally, calling hna_set_subscription_key() without the key as an  
# argument is best, as the user can then be prompted to input the key without  
# typing it directly into the console, making it more secure and less  
# likely to exposed.  
hna_set_subscription_key()  
  
## End(Not run)
```

Index

baseline_get_admin0, [2](#)
baseline_get_admin1, [3](#)
baseline_get_admin2, [4](#)
baseline_get_countries, [5](#)
baseline_get_operations, [6](#)
baseline_get_subscription_key, [6](#)
baseline_set_subscription_key, [7](#)

hna_get_admin2, [8](#)
hna_get_catalog, [9](#)
hna_get_countries, [9](#)
hna_get_subscription_key, [10](#)
hna_set_subscription_key, [10](#)