Package ‘bomrang’

January 20, 2020

Type Package

Title Australian Government Bureau of Meteorology (‘BOM’) Data Client

Version 0.7.0

Description Provides functions to interface with Australian Government Bureau of Meteorology (‘BOM’) data, fetching data and returning a tidy data frame of precis forecasts, historical and current weather data from stations, agriculture bulletin data, ‘BOM’ 0900 or 1500 weather bulletins and downloading and importing radar and satellite imagery files. Data (c) Australian Government Bureau of Meteorology Creative Commons (CC) Attribution 3.0 licence or Public Access Licence (PAL) as appropriate. See <http://www.bom.gov.au/other/copyright.shtml> for further details.

URL https://github.com/ropensci/bomrang,
https://docs.ropensci.org/bomrang/

BugReports https://github.com/ropensci/bomrang/issues

License MIT + file LICENSE

Depends R (>= 3.5.0)

Imports curl (>= 2.8.1), crayon, data.table (>= 1.10.4), dplyr (>= 0.7.0), foreign, httr (>= 1.2.1), hoardr, janitor (>= 1.0.0), jsonlite (>= 1.5), lubridate, magrittr (>= 1.5), raster, rgdal, readr (>= 1.1.1), rvest, tidyr (>= 0.6.3), tools, utils, xml2 (>= 1.1.1)

Encoding UTF-8

LazyData true

Suggests ASGS.foyer, covr, testthat, knitr, rmarkdown, ggplot2, ggthemes, grid, gridExtra, maps, mapproj, rappdirs, roxygen2 (>= 6.1.0)

Language en-AU

RoxygenNote 7.0.2

NeedsCompilation no

ByteCompile TRUE
VignetteBuilder  knitr

X-schema.org-applicationCategory  Tools

X-schema.org-keywords  bom, meteorological-data, weather-forecast, australia, weather, weather-data, meteorology, australia-bureau-of-meteorology

X-schema.org-isPartOf  https://ropensci.org

Author  Adam H. Sparks [aut, cre] (<https://orcid.org/0000-0002-0061-8359>),
Jonathan Carroll [aut] (<https://orcid.org/0000-0002-1404-5264>),
James Goldie [aut] (<https://orcid.org/0000-0002-5024-6207>),
Dean Marchiori [aut],
Paul Melloy [aut] (<https://orcid.org/0000-0003-4253-7167>),
Mark Padgham [aut, rev] (<https://orcid.org/0000-0003-2172-5265>),
Hugh Parsonage [aut] (<https://orcid.org/0000-0003-4055-0835>),
Keith Pembleton [aut] (<https://orcid.org/0000-0002-1896-4516>),
James Balamuta [ctb] (<https://orcid.org/0000-0003-2826-8458>),
Brooke Anderson [rev] (<https://orcid.org/0000-0002-5012-9035>)

Maintainer  Adam H. Sparks <adamhsparks@gmail.com>

Repository  CRAN

Date/Publication  2020-01-20 19:30:29 UTC

R topics documented:

bomrang ................................................................. 3
get_ag_bulletin ....................................................... 3
get_available_imagery ............................................... 5
get_available_radar .................................................. 6
get_coastal_forecast ............................................... 7
get_current_weather ............................................... 8
get_historical ......................................................... 10
get_precis_forecast ................................................ 12
get_radar_imagery ................................................... 13
get_satellite_imagery ............................................... 14
get_weather_bulletin ............................................... 16
manage_cache ....................................................... 17
parse_ag_bulletin ................................................... 18
parse_coastal_forecast ............................................. 20
parse_precis_forecast .............................................. 22
sweep_for_forecast_towns ......................................... 23
sweep_for_stations ................................................ 24
update_forecast_towns ............................................. 25
update_station_locations ......................................... 25

Index  27
bomrang  

---

### bomrang  

**Australian government Bureau of Meteorology (BOM) data client**

### Description

Provides functions to interface with Australian Government Bureau of Meteorology ('BOM') data, fetching data and returning a tidy data frame of precis forecasts, historical and current weather data from stations, agriculture bulletin data, 'BOM' 0900 or 1500 weather bulletins and downloading and importing radar and satellite imagery files. Data (c) Australian Government Bureau of Meteorology Creative Commons (CC) Attribution 3.0 licence or Public Access Licence (PAL) as appropriate. See <http://www.bom.gov.au/other/copyright.shtml> for further details.

### Author(s)

Adam H. Sparks and Jonathan Carroll and James Goldie and Dean Marchiori and Paul Melloy and Mark Padgham and Hugh Parsonage and Keith Pembleton

### See Also

**Useful links:**

- Development repository: https://github.com/ropensci/bomrang
- Static documentation: https://docs.ropensci.org/bomrang/
- Report bugs: https://github.com/ropensci/bomrang/issues

---

### get_ag_bulletin  

**Get BOM agriculture bulletin information for select stations**

### Description

Fetch the BOM agricultural bulletin information and return it in a data frame

### Usage

```
get_ag_bulletin(state = "AUS")
```

### Arguments

- **state**: Australian state or territory as full name or postal code. Fuzzy string matching via `agrep` is done. Defaults to "AUS" returning all state bulletins, see Details for more.
Details

Allowed state and territory postal codes, only one state per request or all using AUS.

ACT  Australian Capital Territory (will return NSW)
NSW  New South Wales
NT   Northern Territory
QLD  Queensland
SA   South Australia
TAS  Tasmania
VIC  Victoria
WA   Western Australia
AUS  Australia, returns forecast for all states, NT and ACT

Value

A data frame as a data.table object of Australia BOM agricultural bulletin information. For full details of fields and units returned see Appendix 3 in the bomrang vignette, use 
vignette("bomrang",package = "bomrang") to view.

Author(s)

Adam H. Sparks, <adamhsparks@gmail.com> and Paul Melloy <paul@melloy.com.au>

References

Agricultural observations are retrieved from the Australian Bureau of Meteorology (BOM) Weather Data Services Agriculture Bulletins,
and
Australian Bureau of Meteorology (BOM)) Weather Data Services Observation of Rainfall,
Station location and other metadata are sourced from the Australian Bureau of Meteorology (BOM) webpage, Bureau of Meteorology Site Numbers:

See Also

parse_ag_bulletin

Examples

ag_bulletin <- get_ag_bulletin(state = "QLD")
ag_bulletin
**get_available_imagery**  
*Get a listing of available BOM satellite GeoTIFF imagery*

**Description**
Fetch a listing of BOM 'GeoTIFF' satellite imagery from ftp://ftp.bom.gov.au/anon/gen/gms/ to determine which files are currently available for download. Files are available at ten minute update frequency with a 24 hour delete time. Useful to know the most recent files available and then specify in the `get_satellite_imagery` function.

**Usage**
```r
get_available_imagery(product_id = "all")
```

**Arguments**
- `product_id` Character. BOM product ID of interest for which a list of available images will be returned. Defaults to all images currently available.

**Details**
Valid BOM satellite Product IDs for 'GeoTIFF' files include:

- **IDE00420** AHI cloud cover only 2km FD GEOS GIS
- **IDE00421** AHI IR (Ch13) greyscale 2km FD GEOS GIS
- **IDE00422** AHI VIS (Ch3) greyscale 2km FD GEOS GIS
- **IDE00423** AHI IR (Ch13) Zehr 2km FD GEOS GIS
- **IDE00425** AHI VIS (true colour) / IR (Ch13 greyscale) composite 1km FD GEOS GIS
- **IDE00426** AHI VIS (true colour) / IR (Ch13 greyscale) composite 2km FD GEOS GIS
- **IDE00427** AHI WV (Ch8) 2km FD GEOS GIS
- **IDE00430** AHI cloud cover only 2km AUS equirect. GIS
- **IDE00431** AHI IR (Ch13) greyscale 2km AUS equirect. GIS
- **IDE00432** AHI VIS (Ch3) greyscale 2km AUS equirect. GIS
- **IDE00433** AHI IR (Ch13) Zehr 2km AUS equirect. GIS
- **IDE00435** AHI VIS (true colour) / IR (Ch13 greyscale) composite 1km AUS equirect. GIS
- **IDE00436** AHI VIS (true colour) / IR (Ch13 greyscale) composite 2km AUS equirect. GIS
- **IDE00437** AHI WV (Ch8) 2km AUS equirect. GIS
- **IDE00439** AHI VIS (Ch3) greyscale 0.5km AUS equirect. GIS

**Value**
A vector of all available files for the requested Product ID(s).
get_available_radar

**Author(s)**

Adam H. Sparks, <adamhsparks@gmail.com>

**References**


**Examples**

```r
# Check availability of AHI VIS (true colour) / IR (Ch13 greyscale) composite
# 1km FD GEOS GIS images
imagery <- get_available_imagery(product_id = "IDE00425")
```

---

**get_available_radar**  
*Get a listing of available BOM radar imagery*

**Description**

Fetch a listing of available BOM RADAR imagery from [ftp://ftp.bom.gov.au/anon/gen/radar/](ftp://ftp.bom.gov.au/anon/gen/radar/) to determine which files are currently available for download. The files available are the most recent RADAR imagery for each location, which are updated approximately every 6 to 10 minutes by the BOM.

**Usage**

```r
get_available_radar(radar_id = "all")
```

**Arguments**

- `radar_id`  
  Character. BOM radar ID of interest for which a list of available images will be returned. Defaults to all images currently available.

**Details**

Valid BOM RADAR ID for each location required.

**Value**

A data frame of all selected RADAR locations with location information and `product_ids`.

**Author(s)**

Dean Marchiori, <deanmarchiori@gmail.com>
get_coastal_forecast

References

Examples

# Check availability radar imagey for Wollongong (radar_id = 3)
imagery <- get_available_radar(radar_id = "3")

---

get_coastal_forecast Get BOM coastal waters forecast

Description
Fetch the BOM daily Coastal Waters Forecast and return a data frame of the forecast regions for a specified state or region.

Usage

get_coastal_forecast(state = "AUS")

Arguments

crate state Australian state or territory as full name or postal code. Fuzzy string matching via agrep is done. Defaults to "AUS" returning all state forecasts, see details for further information.

Details
Allowed state and territory postal codes, only one state per request or all using AUS.

ACT Australian Capital Territory (will return NSW)
NSW New South Wales
NT Northern Territory
QLD Queensland
SA South Australia
TAS Tasmania
VIC Victoria
WA Western Australia
AUS Australia, returns forecast for all states, NT and ACT
get_current_weather

Value

A data.table of an Australia BOM Coastal Waters Forecast. For full details of fields and units returned see Appendix 5 in the bomrang vignette, use vignette("bomrang",package = "bomrang") to view.

Author(s)

Dean Marchiori, <deanmarchiori@gmail.com> and Paul Melloy <paul@melloy.com.au>

References


See Also

parse_coastal_forecast()

Examples

coastal_forecast <- get_coastal_forecast(state = "NSW")

coastal_forecast

get_current_weather

Get current weather observations of a BOM station

Description

Get current weather observations of a BOM station

Usage

get_current_weather(
  station_name,
  strict = FALSE,
  latlon = NULL,
  emit_latlon_msg = TRUE
)

get_current_weather

Arguments

- **station_name**: The name of the weather station. Fuzzy string matching via `agrep` is done.
- **strict**: (logical) If TRUE, `station_name` must match the station name exactly, except that `station_name` need not be upper case. Note this may be different to `full_name` in the response. See Details.
- **latlon**: A length-2 numeric vector giving the decimal degree latitude and longitude (in that order), e.g., `latlon = c(-34,151)` for Sydney. When given instead of `station_name`, the nearest station (in this package) is used, with a message indicating the nearest such station. (See also `sweep_for_stations`.) Ignored if used in combination with `station_name`, with a warning.
- **emit_latlon_msg**: Logical. If TRUE (the default), and `latlon` is selected, a message is emitted before the table is returned indicating which station was actually used (i.e., which station was found to be nearest to the given coordinate).

Details

Station names are not consistently named within the Bureau, so the response may contain a different `full_name` to the one matched, even if `strict = TRUE`. For example, `get_current_weather("CASTLEMAINE PRISON")[["full_name"]][1]` is Castlemaine, not Castlemaine Prison.

Note that the column `local_date_time_full` is set to a POSIXct object in the local time of the user. For more details see "Appendix 1 - Output from get_current_weather()" in the bomrang vignette

vignette("bomrang",package = "bomrang")
for a complete list of fields and units.

Value

A `bomrang_tbl` object (extension of a `data.frame`) of requested BOM station’s current and prior 72hr data. For full details of fields and units returned, see Appendix 1 in the bomrang vignette, use vignette("bomrang",package = "bomrang") to view.

Author(s)

Hugh Parsonage. <hugh.parsonage@gmail.com>

References


get_historical

Obtain historical BOM data

Description

Retrieves daily observations for a given station.

Usage

get_historical(
  stationid = NULL,
  latlon = NULL,
  radius = NULL,
  type = c("rain", "min", "max", "solar")
)

Arguments

- **stationid**: BOM station 'ID'. See Details.
- **latlon**: Length-2 numeric vector of Latitude/Longitude. See Details.
- **radius**: Numeric value, distance (km) from latlon, must be numeric.
- **type**: Measurement type, either daily "rain", "min" (temp), "max" (temp), or "solar" (exposure). Partial matching is performed. If not specified returns the first matching type in the order listed.

Value

A bomrang_tbl object (extension of a data.frame) of historical observations for the chosen station/product type, with some subset of the following columns

<table>
<thead>
<tr>
<th><strong>Product_code</strong></th>
<th>BOM internal code.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Station_number</strong></td>
<td>BOM station ID.</td>
</tr>
<tr>
<td><strong>Year</strong></td>
<td>Year of observation (YYYY).</td>
</tr>
<tr>
<td><strong>Month</strong></td>
<td>Month of observation (1-12).</td>
</tr>
<tr>
<td><strong>Day</strong></td>
<td>Day of observation (1-31).</td>
</tr>
</tbody>
</table>
Min_temperature: Minimum daily recorded temperature (degrees C).
Max_temperature: Maximum daily recorded temperature (degrees C).
Accum_days_min: Accumulated number of days of minimum temperature.
Accum_days_max: Accumulated number of days of maximum temperature.
Rainfall: Daily recorded rainfall in mm.
Period: Period over which rainfall was measured.
Solar_exposure: Daily global solar exposure in MJ/m^2.
Quality: Y, N, or missing. Data which have not yet completed the routine quality control process are marked accordingly.

The following attributes are set on the data, and these are used to generate the header

- **site**: BOM station ID.
- **name**: BOM station name.
- **lat**: Latitude in decimal degrees.
- **lon**: Longitude in decimal degrees.
- **start**: Date observations start.
- **end**: Date observations end.
- **years**: Available number of years data.
- **percent**: Percent complete.
- **AWS**: Automated weather station?
- **type**: Measurement types available for the station.

**Caution**

Temperature data prior to 1910 should be used with extreme caution as many stations prior to that date were exposed in non-standard shelters. Some of which give readings which are several degrees warmer or cooler than those measured according to post-1910 standards.

Daily maximum temperatures usually occur in the afternoon and daily minimum temperatures overnight or near dawn. Occasionally, however, the lowest temperature in the 24 hours to prior to 9 AM can occur around 9 AM the previous day if the night was particularly warm.

Either **stationid** or **latlon** must be provided, but if both are, then **stationid** will be used as it is more reliable.

In some cases data is available back to the 1800s, so tens-of-thousands of daily records will be returned. Other stations will be newer and will return fewer observations.

**dplyr Compatibility**

The **bomrang_tbl** class is compatible with **dplyr** as long as the **bomrang** package is on the search path. Common functions (**filter, select, arrange, mutate, rename, arrange, slice, group_by**) are provided which mask the **dplyr** versions (but use those internally, maintaining attributes).

**Author(s)**

Jonathan Carroll, <rpkg@jcarroll.com.au>
**get_precis_forecast**

**Examples**

```r
get_historical(stationid = "023000", type = "max") ## ~48,000+ daily records
get_historical(latlon = c(-35.2809, 149.1300),
    type = "min") ## 3,500+ daily records
```

**Description**

Fetch the BOM daily précis forecast and return a data frame of the seven-day town forecasts for a specified state or territory.

**Usage**

```r
get_precis_forecast(state = "AUS")
```

**Arguments**

- `state` 
  Australian state or territory as full name or postal code. Fuzzy string matching via `agrep` is done. Defaults to "AUS" returning all state bulletins, see Details for more.

**Details**

Allowed state and territory postal codes, only one state per request or all using AUS.

- **ACT** Australian Capital Territory (will return NSW)
- **NSW** New South Wales
- **NT** Northern Territory
- **QLD** Queensland
- **SA** South Australia
- **TAS** Tasmania
- **VIC** Victoria
- **WA** Western Australia
- **AUS** Australia, returns forecast for all states, NT and ACT

**Value**

A `data.table` of an Australia BOM précis seven day forecasts for BOM selected towns. For full details of fields and units returned see Appendix 2 in the `bomrang` vignette, use `vignette("bomrang",package = "bomrang")` to view.
get_radar_imagery

Author(s)
Adam H. Sparks, <adamhsparks@gmail.com> and Keith Pembleton, <keith.pembleton@usq.edu.au> and Paul Melloy, <paul@melloy.com.au>

References

See Also
parse_precis_forecast

Examples

# get the short forecast for Queensland
BOM_forecast <- get_precis_forecast(state = "QLD")
BOM_forecast

get_radar_imagery

Get BOM radar imagery

Description
Fetch BOM radar imagery from ftp://ftp.bom.gov.au/anon/gen/radar/ and return a raster layer object. Files available are the most recent radar snapshot which are updated approximately every 6 to 10 minutes. Suggested to check file availability first by using get_available_radar.

Usage

get_radar_imagery(product_id, path = NULL, download_only = FALSE)

Arguments

product_id Character. BOM product ID to download and import as a raster object. Value is required.
path Character. A character string with the name where the downloaded file is saved. If not provided, the default value NULL is used which saves the file in a temp directory.
download_only Logical. Whether the radar image is loaded into the environment as a raster layer, or just downloaded.
get_satellite_imagery

**Details**

Valid BOM RADAR Product IDs for radar imagery can be obtained from `get_available_radar`.

**Value**

A raster layer based on the most recent `.gif` RADAR image snapshot published by the BOM. If `download_only = TRUE` there will be a ‘NULL’ return value with the download path printed in the console as a message.

**Author(s)**

Dean Marchiori, <deanmarchiori@gmail.com>

**References**

Australian Bureau of Meteorology (BOM) radar images  

**See Also**

`get_available_radar`

**Examples**

```r
# Fetch most recent radar image for Wollongong 256km radar
library(raster)
imagery <- get_radar_imagery(product_id = "IDR032")
plot(imagery)

# Save imagery to a local path
imagery <- get_radar_imagery(product_id = "IDR032", path = "image.gif")
```

---

**get_satellite_imagery**  
Get BOM Satellite GeoTIFF Imagery

**Description**


**Usage**

```r
get_satellite_imagery(product_id, scans = 1, cache = FALSE)
```
get_satellite_imagery

Arguments

product_id Character. BOM product ID to download in 'GeoTIFF' format and import as a stack object. A vector of values from get_available_imagery may be used here. Value is required.

scans Numeric. Number of scans to download, starting with most recent and progressing backwards, e.g., 1 - the most recent single scan available, 6 - the most recent hour available, 12 - the most recent 2 hours available, etc. Negating will return the oldest files first. Defaults to 1. Value is optional.

cache Logical. Store image files locally for later use? If FALSE, the downloaded files are removed when R session is closed. To take advantage of cached files in future sessions, use cache = TRUE. Defaults to FALSE. Value is optional.

Details

Valid BOM satellite Product IDs for use with product_id include:

IDE00420 AHI cloud cover only 2km FD GEOS GIS
IDE00421 AHI IR (Ch13) greyscale 2km FD GEOS GIS
IDE00422 AHI VIS (Ch3) greyscale 2km FD GEOS GIS
IDE00423 AHI IR (Ch13) Zehr 2km FD GEOS GIS
IDE00425 AHI VIS (true colour) / IR (Ch13 greyscale) composite 1km FD GEOS GIS
IDE00426 AHI VIS (true colour) / IR (Ch13 greyscale) composite 2km FD GEOS GIS
IDE00427 AHI WV (Ch8) 2km FD GEOS GIS
IDE00420 AHI cloud cover only 2km AUS equirect. GIS
IDE00421 AHI IR (Ch13) greyscale 2km AUS equirect. GIS
IDE00422 AHI VIS (Ch3) greyscale 2km AUS equirect. GIS
IDE00423 AHI IR (Ch13) Zehr 2km AUS equirect. GIS
IDE00425 AHI VIS (true colour) / IR (Ch13 greyscale) composite 1km AUS equirect. GIS
IDE00426 AHI VIS (true colour) / IR (Ch13 greyscale) composite 2km AUS equirect. GIS
IDE00427 AHI WV (Ch8) 2km AUS equirect. GIS
IDE00429 AHI VIS (Ch3) greyscale 0.5km AUS equirect. GIS

We cache using hoardr, find your cache folder by executing manage_cache$cache_path_get.

Value

A raster stack of GeoTIFF images with layers named by BOM Product ID, timestamp and band.

Author(s)

Adam H. Sparks, <adamhsparks@gmail.com>

References

Australian Bureau of Meteorology (BOM) high-definition satellite images
get_weather_bulletin

**See Also**

*get_available_imagery*  *manage_cache*

**Examples**

```r
# Fetch AHI VIS (true colour) / IR (Ch13 greyscale) composite 1km FD
# GEOS GIS raster stack for most recent single scan available
imagery <- get_satellite_imagery(product_id = "IDE00425", scans = 1)

# Get a list of available image files and use that to specify files for
# download, downloading the two most recent files available
avail <- get_available_imagery(product_id = "IDE00425")
imagery <- get_satellite_imagery(product_id = avail, scans = 2)
```

---

**get_weather_bulletin**  *Get BOM 0900 or 1500 weather bulletin*

**Description**

Fetch the daily BOM 0900 or 1500 weather bulletins and return a data frame for a specified state or territory.

**Usage**

```r
get_weather_bulletin(state = "qld", morning = TRUE)
```

**Arguments**

- **state**  
  Australian state or territory as full name or postal code. Fuzzy string matching via `agrep` is done.
- **morning**  
  If TRUE, return the 9am bulletin for the nominated state; otherwise return the 3pm bulletin.

**Details**

Allowed state and territory postal codes:

- **ACT**  
  Australian Capital Territory (will return NSW)
- **NSW**  
  New South Wales
- **NT**  
  Northern Territory
- **QLD**  
  Queensland
- **SA**  
  South Australia
- **TAS**  
  Tasmania
It is not possible to return weather bulletins for the entire country in a single call. Rainfall figures for the 9am bulletin are generally for the preceding 24 hours, while those for the 3pm bulletin are for the preceding 6 hours since 9am. Note that values are manually entered into the bulletins and sometimes contain typographical errors which may lead to warnings about "NAs introduced by coercion".

**Value**

Data frame as a data.table object of Australian 9am or 3pm weather observations for a state. For full details of fields and units returned see Appendix 4, "Appendix 4 - Output from get_weather_bulletin()" in the bomrang vignette, use vignette("bomrang",package = "bomrang") to view.

**Author(s)**

Mark Padgham, <mark.padgham@email.com>

**References**

Daily observation data come from Australian Bureau of Meteorology (BOM) website. The 3pm bulletin for Queensland is, for example, http://www.bom.gov.au/qld/observations/3pm_bulletin.shtml

**Examples**

```r
qld_weather <- get_weather_bulletin(state = "QLD", morning = FALSE)
qld_weather
```

---

**manage_cache**

Manage locally cached bomrang files

**Description**

Manage cached bomrang satellite imagery files with hoardr.

**Details**

The default cache directory is file.path(rappdirs::user_cache_dir(),"R/bomrang"), but you can set your own path using manage_cache$cache_path_set().

manage_cache$cache_delete only accepts one file name, while manage_cache$cache_delete_all does not accept any names, but deletes all files. For deleting many specific files, use manage_cache$cache_delete in an lapply type call.
Useful user functions

- `manage_cache$cache_path_get()` - get cache path
- `manage_cache$cache_path_set()` - set cache path
- `manage_cache$list()` - returns a character vector of full path file names
- `manage_cache$files()` - returns file objects with metadata
- `manage_cache$details()` - returns files with details
- `manage_cache$delete()` - delete specific files
- `manage_cache$delete_all()` - delete all files, returns nothing

Examples

```r
## Not run:

# list files in cache
imagery <- get_satellite_imagery(product_id = "IDE00425",
                               scans = 1,
                               cache = TRUE)
manage_cache$list()

# delete certain database files
manage_cache$delete("file path")
manage_cache$list()

# delete all files in cache
manage_cache$delete_all()
manage_cache$list()

# set a different cache path from the default
manage_cache$cache_path_set("~/tmp")

## End(Not run)
```

**parse_ag_bulletin**

Parse local BOM agriculture bulletin XML file(s) for select stations

**Description**

Parse local BOM agriculture bulletin XML file(s) and return a data frame for a specified state or territory or all Australia.

**Usage**

`parse_ag_bulletin(state, filepath)`
**parse_ag_bulletin**

**Arguments**

- **state**
  - Required value of an Australian state or territory as full name or postal code. Fuzzy string matching via `agrep` is done.

- **filepath**
  - A string providing the directory location of the précis file(s) to parse. See Details for more.

**Details**

Allowed state and territory postal codes, only one state per request or all using `AUS`.

- **ACT** Australian Capital Territory (will return NSW)
- **NSW** New South Wales
- **NT** Northern Territory
- **QLD** Queensland
- **SA** South Australia
- **TAS** Tasmania
- **VIC** Victoria
- **WA** Western Australia
- **AUS** Australia, returns forecast for all states, NT and ACT

The `filepath` argument will only accept a directory where files are located for parsing. DO NOT supply the full path including the file name. This function will only parse the requested state or all of Australia in the same fashion as `get_precis_forecast()`, provided that the files are all present in the directory.

**Value**

A `data.table` of Australia BOM agricultural bulletin information. For full details of fields and units returned see Appendix 3 in the `bomrang` vignette, use `vignette("bomrang", package = "bomrang")` to view.

**Author(s)**

Adam H. Sparks, <adamhsparks@gmail.com> and Paul Melloy <paul@melloy.com.au>

**References**


and


parse_coastal_forecast

Parse local BOM coastal waters forecast XML files

Description

Parse local BOM daily coastal waters forecast XML file(s) and return a data frame for a specified state or territory or all Australia.

Usage

parse_coastal_forecast(state = "AUS", filepath)

Arguments

state Required value of an Australian state or territory as full name or postal code. Fuzzy string matching via agrep is done.
filepath A string providing the directory location of the coastal forecast file(s) to parse. See Details for more.

Details

Allowed state and territory postal codes, only one state per request or all using AUS.

ACT Australian Capital Territory (will return NSW)
NSW New South Wales
NT Northern Territory
QLD Queensland  
SA South Australia  
TAS Tasmania  
VIC Victoria  
WA Western Australia  
AUS Australia, returns forecast for all states, NT and ACT

The `filepath` argument will only accept a directory where files are located for parsing. DO NOT supply the full path including the file name. This function will only parse the requested state or all of Australia in the same fashion as `get_coastal_forecast()`, provided that the files are all present in the directory.

Value

A `data.table` of an Australia BOM Coastal Waters Forecast. For full details of fields and units returned see Appendix 5 in the `bomrang` vignette, use `vignette("bomrang",package = "bomrang")` to view.

Author(s)

Dean Marchiori, <deanmarchiori@gmail.com> and Paul Melloy <paul@melloy.com.au>

References


See Also

`get_coastal_forecast`

Examples

```r
# parse the coastal forecast for Queensland

destfile = file.path(tempdir()),  
mode = "wb")

coastal_forecast <- parse_coastal_forecast(state = "QLD",  
                                        filepath = tempdir())

coastal_forecast
```
parse_precis_forecast  Parse local BOM daily précis forecast XML file(s) for select towns

Description

Parse local BOM daily précis forecast XML file(s) and return a data frame of the seven-day town forecasts for a specified state or territory or all Australia.

Usage

parse_precis_forecast(state, filepath)

Arguments

state  Required value of an Australian state or territory as full name or postal code. Fuzzy string matching via agrep is done.
filepath  A string providing the directory location of the précis file(s) to parse. See Details for more.

Details

Allowed state and territory postal codes, only one state per request or all using AUS.

ACT  Australian Capital Territory (will return NSW)
NSW  New South Wales
NT  Northern Territory
QLD  Queensland
SA  South Australia
TAS  Tasmania
VIC  Victoria
WA  Western Australia
AUS  Australia, returns forecast for all states, NT and ACT

The filepath argument will only accept a directory where files are located for parsing. DO NOT supply the full path including the file name. This function will only parse the requested state or all of Australia in the same fashion as `get_precis_forecast()`, provided that the files are all present in the directory.

Value

A data table of Australia BOM précis seven day forecasts for BOM selected towns. For full details of fields and units returned see Appendix 2 in the bomrang vignette, use vignette("bomrang",package = "bomrang") to view.
sweep_for_forecast_towns

Author(s)
Adam H. Sparks, <adamhsparks@gmail.com> and Keith Pembleton, <keith.pembleton@usq.edu.au> and Paul Melloy, <paul@melloy.com.au>

References
Forecast data come from Australian Bureau of Meteorology (BOM) Weather Data Services
Location data and other metadata for towns come from the BOM anonymous FTP server with spatial data

See Also
get_precis_forecast

Examples

# parse the short forecast for Queensland
# download to tempfile() using basename() to keep original name
destfile = file.path(tempdir(),
mode = "wb")
BOM_forecast <- parse_precis_forecast(state = "QLD",
filepath = tempdir())
BOM_forecast

sweep_for_forecast_towns

Find nearest BOM forecast towns

Description
Find nearest BOM forecast towns

Usage
sweep_for_forecast_towns(latlon = c(-35.3, 149.2))
sweep_for_stations

Arguments

latlon  A length-2 numeric vector. By default, Canberra (approximately).

Value

A data.table of all forecast towns (in this package) sorted by distance from latlon, ascending.

Author(s)

Hugh Parsonage, <hugh.parsonage@gmail.com> and James Goldie, <me@rensa.co>

---

sweep_for_stations  Find nearest BOM weather stations

Description

Find nearest BOM weather stations

Usage

sweep_for_stations(latlon = c(-35.3, 149.2))

Arguments

latlon  A length-2 numeric vector. By default, Canberra (approximately).

Value

A data.table of all weather stations (in this package) sorted by distance from latlon, ascending.

Author(s)

Hugh Parsonage, <hugh.parsonage@gmail.com>
update_forecast_towns

Description

Download the latest select forecast towns from the BOM server and update internal database of précis forecast town names and AAC codes used by `get_precis_forecast`. There is no need to use this unless you know that a forecast town exists in a more current version of the BOM précis forecast town name database that is not available in the database distributed with `bomrang`. In fact, for reproducibility purposes, users are discouraged from using this function.

Usage

```r
update_forecast_towns()
```

Value

Updated database of BOM précis forecast towns

Author(s)

Adam H. Sparks, <adamhsparks@gmail.com>

References


Examples

```r
## Not run:
update_forecast_towns()
```

## End(Not run)

update_station_locations

Update internal databases with latest BOM station metadata

Description

Download the latest station locations and metadata and update internal databases that support the use of `get_current_weather` `get_ag_bulletin` and `get_historical`. There is no need to use this unless you know that a station exists in BOM’s database that is not available in the databases distributed with `bomrang`. In fact, for reproducibility purposes, users are discouraged from using this function.
Usage

update_station_locations()

Details

If ASGS.foyer is installed locally, this function will automatically check and correct any invalid state values for stations located in Australia. If ASGS.foyer is not installed, the function will update the internal database without validating the state values for stations by reported longitude/latitude location.

Value

Updated internal databases of BOM station locations and JSON URLs

Author(s)

Adam H. Sparks, <adamhsparks@gmail.com>

References

Station location and other metadata are sourced from the Australian Bureau of Meteorology (BOM) webpage, Bureau of Meteorology Site Numbers:

Examples

```r
## Not run:
update_station_locations()

## End(Not run)
```
Index

agrep, 3, 7, 9, 12, 16, 19, 20, 22
arrange, 11
bomrang, 3
bomrang-package (bomrang), 3
data.frame, 9, 10
data.table, 4, 8, 12, 17, 19, 21, 22, 24
dplyr, 11
filter, 11
get_ag_bulletin, 3, 25
get_available_imagery, 5, 14–16
get_available_radar, 6, 13, 14
get_coastal_forecast, 7
get_current_weather, 8, 25
get_historical, 10, 25
get_precis_forecast, 12, 25
get_radar_imagery, 13
get_satellite_imagery, 5, 14
get_weather_bulletin, 16
group_by, 11
lapply, 17
manage_cache, 16, 17
mutate, 11
parse_ag_bulletin, 18
parse_coastal_forecast, 20
parse_precis_forecast, 22
raster, 13
rename, 11
select, 11
slice, 11
stack, 14, 15
sweep_for_forecast_towns, 23
sweep_for_stations, 9, 24
update_forecast_towns, 25
update_station_locations, 25