

# Package: tidygeoRSS (via r-universe)

September 4, 2024

**Type** Package

**Title** Tidy GeoRSS

**Version** 0.0.1

**Author** Robert Myles McDonnell

**Maintainer** Robert Myles McDonnell <robertmylesmcdonnell@gmail.com>

**Description** In order to easily integrate geoRSS data into analysis, 'tidygeoRSS' parses 'geo' feeds and returns tidy simple features data frames.

**URL** <https://github.com/RobertMyles/tidygeoRSS>

**BugReports** <https://github.com/RobertMyles/tidygeoRSS/issues>

**Encoding** UTF-8

**License** MIT + file LICENSE

**Depends** R (>= 3.1.0)

**Imports** xml2 (>= 1.3.1), httr (>= 1.4.1), anytime (>= 0.3.7), dplyr (>= 1.0.0), tidyRSS (>= 2.0.2), jsonlite (>= 1.6.1), strex (>= 1.2.0), stringr (>= 1.4.0), magrittr (>= 1.5), purrr (>= 0.3.3), sf (>= 0.9.1), rlang (>= 0.4.6)

**Suggests** httpptest, knitr, rmarkdown, testthat (>= 2.1.0)

**RoxygenNote** 7.1.0

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

**RemoteUrl** <https://github.com/robertmyles/tidygeorss>

**RemoteRef** HEAD

**RemoteSha** 94006565750fdb9e7345c9918b785d3a2282e32

## Contents

tidygeo	2
Index	3

---

`tidygeo`*Extract a tidy data frame from geoRSS, geo-Atom and geoJSON feeds*

---

## Description

Extract a tidy data frame from geoRSS, geo-Atom and geoJSON feeds

## Usage

```
tidygeo(  
  feed,  
  config = list(),  
  clean_tags = TRUE,  
  list = FALSE,  
  parse_dates = TRUE  
)
```

## Arguments

<code>feed</code>	character, the url for the feed that you want to parse, e.g. "https://earthquake.usgs.gov/earthquakes/feed/
<code>config</code>	Arguments passed off to <code>httr::GET()</code> .
<code>clean_tags</code>	default TRUE. Clean HTML tags from summary and content?
<code>list</code>	default FALSE. Return metadata and entry information in separate dataframes? These will be combined in a named list.
<code>parse_dates</code>	logical, default TRUE. If TRUE, tidyRSS will attempt to parse columns that contain datetime values, although this may fail, see note.

## Value

A data frame of class "sf", "tbl\_df", "tbl" and "data.frame".

## Note

`tidygeo()` attempts to parse columns that should contain dates. This can fail, as can be seen in tidyRSS [here](#). If you need lower-level control over the parsing of dates, it's better to leave `parse_dates` equal to FALSE and then parse these yourself.

## Examples

```
## Not run: tidygeo("https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_hour.atom")
```

# Index

tidygeo, [2](#)