Package 'XML2R'

June 4, 2024

Title Easier XML Data Collection

Version 0.0.8

Author Carson Sievert <cpsievert1@gmail.com>

Maintainer Carson Sievert <cpsievert1@gmail.com>

Description Helpers for transforming
XML content into number of tables while preserving parent to child
relationships.

License GPL (>= 2)

Depends R (>= 2.15.1)

Imports XML, httr, plyr

URL https://github.com/cpsievert/XML2R

BugReports https://github.com/cpsievert/XML2R/issues

RoxygenNote 7.3.1

NeedsCompilation no

Repository CRAN

Date/Publication 2024-06-04 21:20:02 UTC

Contents

add_key *Add a key to connect parents to descendants*

Description

This function creates a mapping from parent observations to it's descendants (which useful for merging/joining tables). Either an existing value in the parent observation can be `recycled` to it's descendants or a new column will be created (if `recycle` is missing).

Usage

```
add_key(obs, parent, recycle, key.name, quiet = FALSE)
```

Arguments

`obs`	list. Should be the output from listsToObs.
`parent`	character string. Should be present in the names of `obs`.
`recycle`	character string that matches a variable name among `parent` observations.
`key.name`	The desired column name of the newly generated key.
`quiet`	logical. Include message about the keys being generated?

Value

A list of observations.

collapse_obs *Collapse a list of observations into a list of tables.*

Description

This function aggregates all observations with a similar name into a common table. Note that observations with a particular name don't need consistent variables (any missing information is filled with NAs).

Usage

```
collapse_obs(obs)
```

Arguments

`obs`	list of observations.

Value

Returns list with one element for each relevant XML node. Each element contains a matrix.

docsToNodes *Parse XML Documents into XML Nodes*

Description

Essentially a recursive call to getNodeSet.

Usage

```
docsToNodes(docs, xpath)
```

Arguments

docs	XML documents
xpath	xpath expression

listsToObs *Flatten nested list into a list of observations*

Description

This function flattens the nested list into a list of "observations" (that is, a list of matrices with one row). The names of the list that is returned reflects the XML ancestory of each observation.

Usage

```
listsToObs(l, urls, append.value = TRUE, as.equiv = TRUE, url.map = TRUE)
```

Arguments

l	list. Should be the output from nodesToList.
urls	character vector the same length as l. Each element should map element of l to an XML file.
append.value	logical. Should the XML value be appended to the observation?
as.equiv	logical. Should observations from two different files (but the same ancestory) have the same name returned?
url.map	logical. If TRUE, the 'url_key' column will contain a condensed url identifier (for each observation) and full urls will be stored in the "url_map" element. If FALSE, the full urls are included (for each observation) as a 'url' column and no "url_map" is included.

Value

A list where each element reflects one "observation".

nodesToList — *Coerce XML Nodes into a list with both attributes and values*

Description

Essentially a recursive call to xmlToList.

Usage

```
nodesToList(nodes)
```

Arguments

nodes	A collection of XML nodes. Should be the output from docsToNodes.

Value

A nested list with a structure that resembles the XML structure

re_name — *Rename rows of a list*

Description

Sometimes, certain nodes in an XML ancestory may want to be neglected before any keys are created (see add_key) or observations are aggregated (see collapse). This function takes a list of "observations" (that is, a list of matrices with one row) and alters the names of that list. Note that any information lost from changing names is saved in a new column whose name is specified by `diff.name`.

Usage

```
re_name(obs, namez, equiv, diff.name = "diff_name", rename.as, quiet = FALSE)
```

Arguments

obs	list. Should be the output from XML2Obs (or listsToObs).
namez	must be equivalent to `names(obs)`. Intended use is to avoid unneccessarily repeating that operation.
equiv	character vector with the appropriate (unique) names that should be regarded "equivalent".
diff.name	character string used for naming the variable that is appended to any observations whose name was overwritten. The value for this variable is the difference in from the original name and the overwritten name.
rename.as	character string to override naming of observations that are renamed.
quiet	logical. Include message about how observations are being renamed?

Value

A list of "observations".

`urlsToDocs` *Parse XML Files into XML Documents*

Description

Essentially a recursive call to xmlParse.

Usage

```
urlsToDocs(urls, local = FALSE, quiet = FALSE, ...)
```

Arguments

`urls`	character vector. Either urls that point to an XML file online or a local XML file name.
`local`	logical. Should urls be treated as paths to local files?
`quiet`	logical. Print file name currently being parsed?
`...`	arguments passed along to 'httr::GET'

`XML2Obs` *Parse XML files into a list of "observations"*

Description

This function takes a collection of urls that point to XML files and coerces the relevant info into a list of observations. An "observation" is defined as a matrix with one row. An observation can also be thought of as a single instance of XML attributes (and value) for a particular level in the XML hierarchy. The names of the list reflect the XML node ancestory for which each observation was extracted from.

Usage

```
XML2Obs(
  urls,
  xpath,
  append.value = TRUE,
  as.equiv = TRUE,
  url.map = FALSE,
  local = FALSE,
  quiet = FALSE,
  ...
)
```

Arguments

urls	character vector. Either urls that point to an XML file online or a local XML file name.
xpath	XML XPath expression that is passed to getNodeSet. If missing, the entire root and all descendents are captured and returned (ie, tables = "/").
append.value	logical. Should the XML value be appended for relevant observations?
as.equiv	logical. Should observations from two different files (but the same ancestory) have the same name returned?
url.map	logical. If TRUE, the 'url_key' column will contain a condensed url identifier (for each observation) and full urls will be stored in the "url_map" element. If FALSE, the full urls are included (for each observation) as a 'url' column and no "url_map" is included.
local	logical. Should urls be treated as paths to local files?
quiet	logical. Print file name currently being parsed?
...	arguments passed along to 'httr::GET'

Details

It's worth noting that a "url_key" column is appended to each observation to help track the origin of each observation. The value of the "url_key" column is not the actual file name, but a simplified identifier to avoid unnecessarily repeating long file names for each observation. For this reason, an addition element (named "url_map") is added to the list of observations in case the actual file named want to be used.

Value

A list of "observations" and (possibly) the "url_map" element.

See Also

urlsToDocs, docsToNodes, nodesToList, listsToObs

Examples

```
## Not run:
urls <- c("http://gd2.mlb.com/components/game/mlb/year_2013/mobile/346180.xml",
          "http://gd2.mlb.com/components/game/mlb/year_2013/mobile/346188.xml")
obs <- XML2Obs(urls)
table(names(obs))

# parses local files as well
players <- system.file("extdata", "players.xml", package = "XML2R")
obs2 <- XML2Obs(players, local = TRUE)
table(names(obs2))

## End(Not run)
```

XML2R *Parse XML files into (a list of) matrices or data frame(s)*

Description

This function is an experimental wrapper around XML2Obs. One should only use this function over XML2Obs if keys already exist in the XML data and ancestory doesn't need to be altered.

Usage

```
XML2R(urls, xpath, df = FALSE)
```

Arguments

urls	character vector or list of urls that point to an XML file (or anything readable by xmlParse).
xpath	XML XPath expression that is passed to getNodeSet. If missing, the entire root and all descendents are captured and returned (ie, tables = "/").
df	logical. Should matrices be coerced into data frames?

Value

Returns list with one element for each relevant XML node. Each element contains a matrix by default.

See Also

urlsToDocs, docsToNodes, nodesToList, listsToObs

Examples

```
## Not run:
urls2 <- c("http://gd2.mlb.com/components/game/mlb/year_2013/mobile/346180.xml",
           "http://gd2.mlb.com/components/game/mlb/year_2013/mobile/346188.xml")
dat3 <- XML2R(urls2)

cens <- "http://www.census.gov/developers/data/sf1.xml"
census <- XML2R(cens)

## End(Not run)
```

Index