

# Package: neotoma2 (via r-universe)

October 13, 2024

**Title** Working with the Neotoma Paleocology Database

**Date** 2024-07-23

**Version** 1.0.4

**URL** <https://github.com/NeotomaDB/neotoma2>

**BugReports** <https://github.com/NeotomaDB/neotoma2/issues>

**Description** Access and manipulation of data using the Neotoma Paleocology Database. <<https://api.neotomadb.org/api-docs/>>.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.1

**Suggests** testthat, knitr, pkgdown, rmarkdown, Bchron, ggplot2, covr

**Imports** rlang, magrittr, sf, geojsonsf, purrr, lubridate, httr, jsonlite, methods, dplyr, gtools, leaflet, assertthat, stringr, progress, wk, uuid, tidyr

**VignetteBuilder** knitr

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

**RemoteUrl** <https://github.com/neotomadb/neotoma2>

**RemoteRef** HEAD

**RemoteSha** 110b3e3607ce8bfed60abc8baa6fb72d40c18bea

## Contents

add_chronology . . . . .	7
add_chronology,collunit,chronology,data.frame-method . . . . .	7
as.data.frame,authors-method . . . . .	8
as.data.frame,chronologies-method . . . . .	9
as.data.frame,chronology-method . . . . .	9
as.data.frame,collunit-method . . . . .	10
as.data.frame,collunits-method . . . . .	10

as.data.frame,contact-method . . . . .	11
as.data.frame,contacts-method . . . . .	11
as.data.frame,dataset-method . . . . .	12
as.data.frame,datasets-method . . . . .	12
as.data.frame,publication-method . . . . .	13
as.data.frame,publications-method . . . . .	13
as.data.frame,site-method . . . . .	14
as.data.frame,sites-method . . . . .	14
as.data.frame,specimen-method . . . . .	15
as.data.frame,specimens-method . . . . .	15
as.list,sites-method . . . . .	16
author-class . . . . .	16
authors-class . . . . .	17
build_chron . . . . .	17
build_collunits . . . . .	18
build_dataset . . . . .	18
build_sample . . . . .	19
build_sites . . . . .	19
build_specimen . . . . .	20
c,chronologies-method . . . . .	20
c,collunits-method . . . . .	21
c,contact-method . . . . .	21
c,contacts-method . . . . .	22
c,datasets-method . . . . .	22
c,missingOrNULL-method . . . . .	23
c,publications-method . . . . .	23
c,sites-method . . . . .	24
c,specimens-method . . . . .	24
check_args . . . . .	25
check_contacts . . . . .	25
check_contacts.contacts . . . . .	26
chroncontrols . . . . .	26
chroncontrols,site-method . . . . .	27
chroncontrols,sites-method . . . . .	27
chronologies . . . . .	28
chronologies,collunit-method . . . . .	28
chronologies,collunits-method . . . . .	29
chronologies,site-method . . . . .	29
chronologies,sites-method . . . . .	30
chronologies-class . . . . .	30
chronology-class . . . . .	30
cite_data . . . . .	31
cite_data,site-method . . . . .	31
cite_data,sites-method . . . . .	32
collunit-class . . . . .	32
collunits . . . . .	33
collunits,site-method . . . . .	33
collunits,sites-method . . . . .	34

collunits-class . . . . .	34
contact-class . . . . .	34
contacts-class . . . . .	35
coordinates . . . . .	35
coordinates,sites-method . . . . .	36
dataset-class . . . . .	36
datasets . . . . .	37
datasets,collunit-method . . . . .	37
datasets,collunits-method . . . . .	38
datasets,site-method . . . . .	38
datasets,sites-method . . . . .	39
datasets-class . . . . .	39
doi . . . . .	39
doi,publication-method . . . . .	40
doi,site-method . . . . .	40
doi,sites-method . . . . .	41
filter . . . . .	41
filter.sites . . . . .	43
getids . . . . .	44
getids.collunit . . . . .	45
getids.collunits . . . . .	45
getids.site . . . . .	46
getids.sites . . . . .	46
get_contacts . . . . .	47
get_contacts.default . . . . .	47
get_contacts.numeric . . . . .	48
get_datasets . . . . .	49
get_datasets.default . . . . .	50
get_datasets.numeric . . . . .	51
get_datasets.site . . . . .	52
get_datasets.sites . . . . .	52
get_downloads . . . . .	53
get_downloads.character . . . . .	55
get_downloads.numeric . . . . .	55
get_downloads.sites . . . . .	56
get_manual . . . . .	56
get_publications . . . . .	57
get_publications.default . . . . .	58
get_publications.numeric . . . . .	58
get_publications.publication . . . . .	59
get_publications.publications . . . . .	60
get_sites . . . . .	61
get_sites.default . . . . .	63
get_sites.numeric . . . . .	64
get_sites.sites . . . . .	64
get_specimens . . . . .	65
get_specimens.default . . . . .	66
get_specimens.numeric . . . . .	66

get_specimens.sites . . . . .	67
get_stats . . . . .	68
get_table . . . . .	69
length,chronologies-method . . . . .	69
length,collunits-method . . . . .	70
length,datasets-method . . . . .	70
length,publications-method . . . . .	71
length,samples-method . . . . .	71
length,sites-method . . . . .	72
length,specimens-method . . . . .	72
missingOrNULL-class . . . . .	73
names,collunit-method . . . . .	73
names,contact-method . . . . .	73
names,dataset-method . . . . .	74
names,publication-method . . . . .	74
names,publications-method . . . . .	75
names,site-method . . . . .	75
names,specimen-method . . . . .	76
newURL . . . . .	76
parseURL . . . . .	77
parse_site . . . . .	77
pingNeotoma . . . . .	78
plot,sites,ANY-method . . . . .	78
plot,sites-method . . . . .	79
plotLeaflet . . . . .	79
plotLeaflet,site-method . . . . .	80
plotLeaflet,sites-method . . . . .	80
publication-class . . . . .	81
publications-class . . . . .	81
repositories-class . . . . .	82
repository-class . . . . .	82
sample-class . . . . .	82
samples . . . . .	83
samples,collunit-method . . . . .	83
samples,collunits-method . . . . .	84
samples,site-method . . . . .	84
samples,sites-method . . . . .	85
samples-class . . . . .	85
selectMatch . . . . .	86
selectMatch,publication,logical-method . . . . .	86
selectMatch,publication,numeric-method . . . . .	87
set_chronology . . . . .	87
set_collunit . . . . .	88
set_contact . . . . .	90
set_dataset . . . . .	91
set_default . . . . .	92
set_default,chronologies-method . . . . .	93
set_publications . . . . .	93

set_sample . . . . .	95
set_server . . . . .	96
set_site . . . . .	97
show,collunit-method . . . . .	98
show,collunits-method . . . . .	98
show,contact-method . . . . .	99
show,contacts-method . . . . .	99
show,dataset-method . . . . .	100
show,datasets-method . . . . .	100
show,publication-method . . . . .	101
show,publications-method . . . . .	101
show,site-method . . . . .	102
show,sites-method . . . . .	102
show,specimen-method . . . . .	103
show,specimens-method . . . . .	103
showMatch . . . . .	104
showMatch,publication-method . . . . .	104
site-class . . . . .	105
sites-class . . . . .	105
specimen-class . . . . .	105
specimens . . . . .	106
specimens,collunit-method . . . . .	106
specimens,collunits-method . . . . .	107
specimens,site-method . . . . .	107
specimens,sites-method . . . . .	108
specimens-class . . . . .	108
summary,sites-method . . . . .	109
taxa . . . . .	109
taxa,collunit-method . . . . .	110
taxa,collunits-method . . . . .	110
taxa,site-method . . . . .	111
taxa,sites-method . . . . .	111
toJSON . . . . .	112
toJSON,sites-method . . . . .	113
toWide . . . . .	113
write.csv,chronologies-method . . . . .	115
write.csv,collunits-method . . . . .	115
write.csv,datasets-method . . . . .	116
write.csv,sites-method . . . . .	116
write.csv,specimens-method . . . . .	117
[,collunits,numeric,ANY-method . . . . .	117
[,collunits,numeric-method . . . . .	118
[,datasets,numeric,ANY-method . . . . .	118
[,datasets,numeric-method . . . . .	119
[,site,character,ANY-method . . . . .	119
[,site,character-method . . . . .	120
[,site,numeric,ANY-method . . . . .	120
[,site,numeric-method . . . . .	121

[,sites,numeric,ANY-method . . . . .	121
[,sites,numeric-method . . . . .	122
[,specimens,numeric,ANY-method . . . . .	122
[,specimens,numeric-method . . . . .	123
[<-,collunit,character,ANY,ANY-method . . . . .	123
[<-,collunit,numeric,ANY,ANY-method . . . . .	124
[<-,dataset,character,ANY,ANY-method . . . . .	124
[<-,dataset,numeric,ANY,ANY-method . . . . .	125
[<-,site,character,ANY,ANY-method . . . . .	125
[<-,site,numeric,ANY,ANY-method . . . . .	126
[<-,specimen,character,ANY,ANY-method . . . . .	126
[<-,specimen,numeric,ANY,ANY-method . . . . .	127
[[,chronologies,numeric,ANY-method . . . . .	127
[[,chronologies,numeric-method . . . . .	128
[[,collunits,numeric,ANY-method . . . . .	128
[[,collunits,numeric-method . . . . .	129
[[,contacts,numeric,ANY-method . . . . .	129
[[,contacts,numeric-method . . . . .	130
[[,datasets,numeric,ANY-method . . . . .	130
[[,datasets,numeric-method . . . . .	131
[[,publications,numeric,ANY-method . . . . .	131
[[,publications,numeric-method . . . . .	132
[[,samples,numeric,ANY-method . . . . .	132
[[,samples,numeric-method . . . . .	133
[[,sites,numeric,ANY-method . . . . .	133
[[,sites,numeric-method . . . . .	134
[[,specimens,numeric,ANY-method . . . . .	134
[[,specimens,numeric-method . . . . .	135
[[<-,collunits,ANY,ANY,ANY-method . . . . .	135
[[<-,datasets,ANY,ANY,ANY-method . . . . .	136
[[<-,publications,ANY,ANY,ANY-method . . . . .	136
[[<-,sites,ANY,ANY,ANY-method . . . . .	137
[[<-,specimens,ANY,ANY,ANY-method . . . . .	137
\$.chronologies-method . . . . .	138
\$.chronology-method . . . . .	138
\$.collunit-method . . . . .	139
\$.collunits-method . . . . .	139
\$.contact-method . . . . .	140
\$.contacts-method . . . . .	140
\$.dataset-method . . . . .	141
\$.datasets-method . . . . .	141
\$.publication-method . . . . .	142
\$.sample-method . . . . .	142
\$.samples-method . . . . .	143
\$.site-method . . . . .	143
\$.sites-method . . . . .	144
\$.specimen-method . . . . .	144
\$.specimens-method . . . . .	145

\$<-,chronology-method . . . . .	145
\$<-,collunit-method . . . . .	146
\$<-,dataset-method . . . . .	146
\$<-,site-method . . . . .	147
\$<-,specimen-method . . . . .	147

**Index** **148**

add\_chronology *Add a new chronology to a collection unit.*

**Description**

Add a new chronology to a collection unit.

**Usage**

```
add_chronology(object, x, y)
```

**Arguments**

object	A collectionunit object
x	A chronology object
y	A data.frame of sample ages

**Value**

chronology object defined by user,

add\_chronology,collunit,chronology,data.frame-method  
*Add a new chronology into an existing collectionunit.*

**Description**

Given a collunit, add a new chronology object to the unit with both the chronology metadata and the age information (as y)

**Usage**

```
## S4 method for signature 'collunit,chronology,data.frame'
add_chronology(object, x, y)
```

**Arguments**

object	A collection unit object
x	A chronology object generated using set_chronology()
y	A data.frame of sample ages, with required columns: "analysisunitid", "age", "agetype", "ageolder", and "ageyounger".

**Details**

When undertaking analysis we may wish to add a new chronology to existing records within Neotoma. To do this we must first build the chronology, but also link it to existing analysis units within the collection unit. For examples from this function, see the [Complex Workflows](#) documentation online.

**Value**

chronologies with new added chronology

---

as.data.frame,authors-method

*Convert a publication author to a data.frame*

---

**Description**

Convert a publication author to a data.frame

**Usage**

```
## S4 method for signature 'authors'
as.data.frame(x)
```

**Arguments**

x	An author
---	-----------

**Value**

data.frame with publications metadata



---

as.data.frame,chronologies-method  
*as.data.frame chronologies*

---

### Description

Convert all slots within each chronology within a chronologies object to a data.frame.

### Usage

```
## S4 method for signature 'chronologies'  
as.data.frame(x)
```

### Arguments

x                    chronologies object

### Value

data.frame with chronologies metadata

---

as.data.frame,chronology-method  
*Create a data.frame from a chronology object.*

---

### Description

Convert all slots within a chronology to a data.frame.

### Usage

```
## S4 method for signature 'chronology'  
as.data.frame(x)
```

### Arguments

x                    chronology object

### Value

data.frame

---

`as.data.frame,collunit-method`  
*as.data.frame site*

---

**Description**

show as dataframe

**Usage**

```
## S4 method for signature 'collunit'  
as.data.frame(x)
```

**Arguments**

x                    site object

**Value**

data.frame object with a collection units metadata

---

`as.data.frame,collunits-method`  
*as.data.frame collunits*

---

**Description**

show as dataframe as prep to save as csv

**Usage**

```
## S4 method for signature 'collunits'  
as.data.frame(x)
```

**Arguments**

x                    collunits object

**Value**

data.frame of multiple collection units metadata.

---

as.data.frame,contact-method

*Transform a contacts object to a data.frame()*

---

### Description

Transform a contacts object to a data.frame()

### Usage

```
## S4 method for signature 'contact'  
as.data.frame(x)
```

### Arguments

x                    A contact object.

### Value

data.frame object with contact metadata

---

as.data.frame,contacts-method

*Transform a contacts object to a data.frame()*

---

### Description

Transform a contacts object to a data.frame()

### Usage

```
## S4 method for signature 'contacts'  
as.data.frame(x)
```

### Arguments

x                    A contacts object.

### Value

data.frame object with multiple contacts metadata

as.data.frame,dataset-method  
*as.data.frame dataset*

---

**Description**

show as dataframe as prep to save as csv

**Usage**

```
## S4 method for signature 'dataset'  
as.data.frame(x)
```

**Arguments**

x                    dataset object

**Value**

data.frame with dataset metadata

---

as.data.frame,datasets-method  
*as.data.frame datasets*

---

**Description**

show as dataframe as prep to save as csv

**Usage**

```
## S4 method for signature 'datasets'  
as.data.frame(x)
```

**Arguments**

x                    datasets object

**Value**

data.frame with datasets metadata

---

as.data.frame,publication-method  
*Convert a publication to a data.frame*

---

**Description**

Convert a publication to a data.frame

**Usage**

```
## S4 method for signature 'publication'  
as.data.frame(x)
```

**Arguments**

x                    A publication object.

**Value**

data.frame with publications' metadata.

---

as.data.frame,publications-method  
*Convert publications to a data.frame*

---

**Description**

Convert publications to a data.frame

**Usage**

```
## S4 method for signature 'publications'  
as.data.frame(x)
```

**Arguments**

x                    A publications object.

**Value**

data.frame with publications' metadata.

---

`as.data.frame,site-method`  
*as.data.frame site*

---

**Description**

show as dataframe as prep to save as csv

**Usage**

```
## S4 method for signature 'site'  
as.data.frame(x)
```

**Arguments**

x                    site object

**Value**

data.frame object with site metadata

---

`as.data.frame,sites-method`  
*as.data.frame sites*

---

**Description**

shows object as data.frame

**Usage**

```
## S4 method for signature 'sites'  
as.data.frame(x)
```

**Arguments**

x                    sites object

**Value**

data.frame object with sites metadata

---

*as.data.frame,specimen-method*  
*as.data.frame specimen*

---

**Description**

show as data.frame

**Usage**

```
## S4 method for signature 'specimen'  
as.data.frame(x)
```

**Arguments**

x                   specimen object

**Value**

data.frame with specimen metadata

---

*as.data.frame,specimens-method*  
*as.data.frame specimens*

---

**Description**

show as data.frame

**Usage**

```
## S4 method for signature 'specimens'  
as.data.frame(x)
```

**Arguments**

x                   specimens object

**Value**

data.frame with specimens metadata

---

as.list, sites-method    *as.list sites*

---

### Description

show as dataframe as prep to save as csv

### Usage

```
## S4 method for signature 'sites'
as.list(x)
```

### Arguments

x                    sites object

### Value

list object with sites metadata

---

author-class                    *An S4 class for the authors of a Neotoma publication.*

---

### Description

This class combines the S4 class contact with a numeric author order. This allows us to reuse contact objects, and to assign the authorship order within a publication. The full set of authors for a publication are represented by the authors object.

### Value

object of class author

### Examples

```
{
simon <- new("contact", familyname = "Goring", givennames = "Simon J.")
firstauthor <- new("author", author = simon, order = 1)
}
```



---

authors-class                    *An S4 class for a set of Neotoma author objects.*

---

### Description

The S4 authors are a set of individual author objects that are then associated with a single S4 publication class.

### Value

object of class authors

### Examples

```
{
  simon <- new("contact", familyname = "Goring", givennames = "Simon J.")
  socorro <- new("contact", familyname = "Dominguez", givennames = "Socorro")
  first_author <- new("author", author = simon, order = 1)
  second_author <- new("author", author = socorro, order = 2)
  author_list <- new("authors", authors = list(first_author, second_author))
}
```

---

build\_chron                    *build\_chron*

---

### Description

A helper function to build a new chronology object from the Neotoma API response.

### Usage

```
build_chron(x)
```

### Arguments

x                    A chronology element from the API JSON output.

### Details

This function is an internal function called from `build_collunit()` to help support the translation between the JSON representation of data in the API and the R implementation.

### Value

A single chronology object.

### Author(s)

Socorro Dominguez

---

build\_collunits      *Build a collection unit from the API response*

---

**Description**

Build a collection unit from the API response

**Usage**

```
build_collunits(x)
```

**Arguments**

x                      The structured JSON from a Neotoma API v2.0 response that returns a collection unit in any form.

**Value**

An simple collunit object

---

build\_dataset      *Build a dataset object from a JSON list representation.*

---

**Description**

Helper function to build a dataset from the API JSON response.

**Usage**

```
build_dataset(x)
```

**Arguments**

x                      a JSON dataset object passed from the Neotoma API.

**Value**

A simple dataset object.

**Author(s)**

Socorro Dominguez

---

build_sample	<i>Build a samples data.frame from Neotoma API JSON</i>
--------------	---

---

**Description**

Helper function to build a sample from the API input (list formatted) coming from the Neotoma API.

**Usage**

```
build_sample(x)
```

**Arguments**

x                    sample list

**Value**

A simple sample object

**Author(s)**

Socorro Dominguez <s.dominguez@ht-data.com>

---

build_sites	<i>Build a site from the Neotoma API response.</i>
-------------	--

---

**Description**

Build a site from the Neotoma API response.

**Usage**

```
build_sites(x)
```

**Arguments**

x                    A list returned from the Neotoma API data section.

**Value**

A simple site object

build\_specimen      *Build a specimen objects.*

---

**Description**

A helper function to build a specimen object from a list returned by the Neotoma API call. The function is not exported, but called from the get\_specimens() call.

**Usage**

```
build_specimen(x)
```

**Arguments**

x                    specimen list

**Value**

A simple specimen object

---

c,chronologies-method    *c Method - Combine chronologies objects*

---

**Description**

c Method - Combine chronologies objects

**Usage**

```
## S4 method for signature 'chronologies'  
c(x, y)
```

**Arguments**

x                    chronologies object 1  
y                    chronologies object 2

**Value**

concatenated chronologies

---

*c, collunits-method*      *c Method - Combine collunits objects*

---

**Description**

*c* Method - Combine collunits objects

**Usage**

```
## S4 method for signature 'collunits'  
c(x, y)
```

**Arguments**

x	collunits object 1
y	collunits object 2

**Value**

concatenated collection units without duplicates

---

*c, contact-method*      *c Method - Combine contacts objects*

---

**Description**

*c* Method - Combine contacts objects

**Usage**

```
## S4 method for signature 'contact'  
c(x, y)
```

**Arguments**

x	contacts object 1
y	contacts object 2

**Value**

contacts concatenated object

---

c,contacts-method      *c Method - Combine contacts objects*

---

**Description**

c Method - Combine contacts objects

**Usage**

```
## S4 method for signature 'contacts'  
c(x, y)
```

**Arguments**

x                      contacts object 1  
y                      contacts object 2

**Value**

concatenated and clean objects

---

c,datasets-method      *c Method - Combine datasets objects*

---

**Description**

c Method - Combine datasets objects

**Usage**

```
## S4 method for signature 'datasets'  
c(x, y)
```

**Arguments**

x                      datasets object 1  
y                      datasets object 2

**Value**

concatenated datasets object

---

c,missingOrNULL-method  
*c Method for NULL values*

---

### Description

c Method for NULL values  
c Method for NULL values

### Usage

```
## S4 method for signature 'missingOrNULL'  
c(x = "missingOrNULL", y)
```

```
## S4 method for signature 'missingOrNULL'  
c(x = "missingOrNULL", y)
```

### Arguments

x	NULL object
y	sites/datasets object

### Value

concatenated collunits object  
list of concatenated items when the first object is NULL

---

c,publications-method *Combine publication objects.*

---

### Description

Combine publication objects.

### Usage

```
## S4 method for signature 'publications'  
c(x, y)
```

### Arguments

x	A publications object.
y	A publications object

### Value

concatenated publications object

---

*c, sites-method*      *c Method - Combine sites objects*

---

**Description**

*c Method - Combine sites objects*

**Usage**

```
## S4 method for signature 'sites'  
c(x, y)
```

**Arguments**

x	sites object 1
y	sites object 2

**Value**

concatenated and cleaned sites object

---

*c, specimens-method*      *c Method - Combine specimens objects*

---

**Description**

*c Method - Combine specimens objects*

**Usage**

```
## S4 method for signature 'specimens'  
c(x, y)
```

**Arguments**

x	specimens object 1
y	specimens object 2

**Value**

concatenated specimens object



---

check_args	<i>check_args</i>
------------	-------------------

---

**Description**

Internal function to check passed arguments.

**Usage**

```
check_args(c1)
```

**Arguments**

c1	called arguments. Arguments are going to be called by match_call inside: <a href="#">get_sites</a> <a href="#">get_datasets</a> <a href="#">get_downloads</a>
----	--

**Value**

A list with two components:

flag	Returns a 0 if everything's fine, a 1 if there's a problem.
message	A list of error messages.

**Author(s)**

Socorro Dominguez

**References**

Neotoma Project Website: <https://www.neotomadb.org/>

---

check_contacts	<i>Check contact information for a record against Neotoma contributors</i>
----------------	--

---

**Description**

Uses the Neotoma API to search and access information about individuals who have contributed to the data in the Neotoma Paleoecology Database

**Usage**

```
check_contacts(x, ...)
```

**Arguments**

x	contacts A contacts object associated with a set of names.
...	Additional parameters associated with the call.

**Value**

contacts object

---

check\_contacts.contacts

*Get contact information for Neotoma contributors*

---

**Description**

Get contact information for Neotoma contributors

**Usage**

```
## S3 method for class 'contacts'
check_contacts(x, similarity = 0.5, ...)
```

**Arguments**

x	contacts A contacts object associated with a set of names.
similarity	The similarity score between matched records (from 0 - 1).
...	Additional parameters associated with the call.

**Value**

contacts object

---

chroncontrols

*chroncontrols*

---

**Description**

Show the samples table

**Usage**

```
chroncontrols(x)
```

**Arguments**

x	Sites object to extract chroncontrols table from
---	--

**Value**

data.frame with chroncontrols information

---

chroncontrols,site-method

*Recover information about the chron controls for a collectionunit.*

---

### Description

For a site that includes collection units with chronologies return the chronological controls that are used in building the chronology.

### Usage

```
## S4 method for signature 'site'  
chroncontrols(x)
```

### Arguments

x                    site object

### Value

data.frame with chronological controls

---

chroncontrols,sites-method

*Recover information about the chron controls for a collectionunit.*

---

### Description

For all sites that includes collection units with chronologies return the chronological controls that are used in building the chronology.

### Usage

```
## S4 method for signature 'sites'  
chroncontrols(x)
```

### Arguments

x                    sites object

### Value

data.frame with chronological controls

---

chronologies	<i>Obtain the chronology from a record or multiple records.</i>
--------------	---

---

**Description**

Obtain the chronology from a record or multiple records.

**Usage**

```
chronologies(x)
```

**Arguments**

x                    sites object that contains chronologies

**Value**

chronologies object with all chronologies used.

---

chronologies,collunit-method	<i>Extract chronologies from a collunit object.</i>
------------------------------	---

---

**Description**

Extract chronologies from a collunit object.

**Usage**

```
## S4 method for signature 'collunit'  
chronologies(x)
```

**Arguments**

x                    A collunit object

**Value**

chronologies from a collunit object

---

chronologies,collunits-method

*Extract chronologies from a collunits object.*

---

**Description**

Extract chronologies from a collunits object.

**Usage**

```
## S4 method for signature 'collunits'  
chronologies(x)
```

**Arguments**

x                    A collunits object

**Value**

chronologies from a collunits object

---

chronologies,site-method

*Extract chronologies from a site object.*

---

**Description**

Extract chronologies from a site object.

**Usage**

```
## S4 method for signature 'site'  
chronologies(x)
```

**Arguments**

x                    A site object

**Value**

chronologies from a site object

---

chronologies, sites-method

*Extract chronologies from a sites object.*

---

### **Description**

Extract chronologies from a sites object.

### **Usage**

```
## S4 method for signature 'sites'
chronologies(x)
```

### **Arguments**

x                    A sites object

### **Value**

chronologies from a sites object

---

chronologies-class    *S4 class for chronologies information*

---

### **Description**

The grouped class for chronologies from the Neotoma Paleocology Database.

### **Value**

object of class chronologies

---

chronology-class      *S4 class for chronologies information*

---

### **Description**

The class for chronologies from the Neotoma Paleocology Database. A single collection unit may have one or more chronology. These individual chronology classes are then grouped into an S4 chronologies class.

### **Value**

object of class chronology

---

cite_data	<i>Generate a data citation from a Neotoma2 object.</i>
-----------	---

---

**Description**

The function, applied to a data object with a valid dataset, will return a properly formatted data citation for the record.

**Usage**

```
cite_data(x)
```

**Arguments**

x                    Object with DOIs associated to it.

**Value**

data.frame with citation data

---

cite_data,site-method	<i>Obtain data citations from a single record.</i>
-----------------------	--

---

**Description**

Given complete dataset objects in Neotoma (must have used `get_datasets()` or `get_downloads()`), return a formatted citation for the record, including the dataset DOI.

**Usage**

```
## S4 method for signature 'site'  
cite_data(x)
```

**Arguments**

x                    sites object

**Value**

data.frame object with citation information.

**Examples**

```
ds <- get_datasets(1)  
cite_data(ds)
```

---

cite\_data, sites-method

*Obtain data citations from multiple records.*

---

### Description

Given complete dataset objects in Neotoma (must have used `get_datasets()` or `get_downloads()`), return a formatted citation for the record, including the dataset DOI.

### Usage

```
## S4 method for signature 'sites'  
cite_data(x)
```

### Arguments

x                    sites object

### Value

data.frame object with citation information.

### Examples

```
{  
ds <- get_datasets(1)  
cite_data(ds)  
}
```

---

collunit-class

*S4 class for collection units information.*

---

### Description

A collection unit represents a collection event from within a site. For example, a lake sediment core, or a single dig site within an archaeological site.

### Value

object of class collunit



---

collunits	<i>Extract collection units from a sites object</i>
-----------	---

---

**Description**

Extract collection units from a sites object

**Usage**

```
collunits(object)
```

**Arguments**

object	A sites object
--------	----------------

**Value**

collunits detail from a sites object

---

collunits,site-method	<i>Extract collunits from a site object.</i>
-----------------------	--

---

**Description**

Extract collunits from a site object.

**Usage**

```
## S4 method for signature 'site'  
collunits(object)
```

**Arguments**

object	A site object
--------	---------------

**Value**

collunits from a site object

---

collunits, sites-method

*Extract collunits from a sites object.*

---

### Description

Extract collunits from a sites object.

### Usage

```
## S4 method for signature 'sites'
collunits(object)
```

### Arguments

object            A sites object

### Value

collunits from a sites object

---

collunits-class

*An S4 class for Neotoma Collection Units*

---

### Description

Holds Collection unit information from the Neotoma Paleocology Database. @returns object of class collunits

---

contact-class

*An S4 class for Neotoma contacts*

---

### Description

The object that contains the contact information for an individual, along with associated metadata.

### Value

object of class contact

### Examples

```
new("contact", familyname = "Goring", givennames = "Simon J.")
```

---

contacts-class	<i>An S4 class for multi-contact information from the Neotoma Paleocology Database.</i>
----------------	---

---

**Description**

An unordered list of individual S4 contact objects.

**Value**

object of class contacts

**Examples**

```
{
# Create two contact objects and associate them within a contacts object.
simon <- new("contact", familyname = "Goring", givennames = "Simon J.")
socorro <- new("contact", familyname = "Dominguez", givennames = "Socorro")
packagers <- new("contacts", contacts = list(simon, socorro))
packagers
}
```

---

coordinates	<i>Obtain coordinates from a sites object.</i>
-------------	--

---

**Description**

Obtain coordinates from a sites object.

**Usage**

```
coordinates(obj, ...)
```

**Arguments**

obj	A sites object
...	Additional parameters associated with the call.

**Value**

dataframe with coordinate values

---

coordinates, sites-method

*Return the latitude and longitude of sites*

---

### Description

Return the latitude and longitude of sites

### Usage

```
## S4 method for signature 'sites'  
coordinates(obj, ...)
```

### Arguments

obj            A sites object  
...            Additional parameters associated with the call.

### Value

data.frame object with site coordinates.

---

dataset-class

*S4 class for dataset information*

---

### Description

The standard object class for datasets from the Neotoma Paleoecology Database.

### Value

object of class dataset

---

datasets	<i>Extract datasets from a sites object.</i>
----------	--

---

**Description**

If the sites object contains datasets, then the datasets will be returned. If the sites object does not contain datasets then the user can apply `get_datasets()` to the object.

**Usage**

```
datasets(object)
```

**Arguments**

object	A sites object
--------	----------------

**Value**

datasets object specific to the metadata contained in datasets

---

datasets, collunit-method	<i>Extract datasets from a collunit object.</i>
---------------------------	---

---

**Description**

Extract datasets from a collunit object.

**Usage**

```
## S4 method for signature 'collunit'  
datasets(object)
```

**Arguments**

object	A collunit object
--------	-------------------

**Value**

datasets from a collunit object

---

datasets,collunits-method

*Extract datasets from a collunits object.*

---

**Description**

Extract datasets from a collunits object.

**Usage**

```
## S4 method for signature 'collunits'  
datasets(object)
```

**Arguments**

object            A collunits object

**Value**

datasets from a collunits object

---

datasets,site-method    *Extract datasets from a site object.*

---

**Description**

Extract datasets from a site object.

**Usage**

```
## S4 method for signature 'site'  
datasets(object)
```

**Arguments**

object            A site object

**Value**

datasets from a site object

---

datasets,sites-method *Extract datasets from a sites object.*

---

**Description**

Extract datasets from a sites object.

**Usage**

```
## S4 method for signature 'sites'
datasets(object)
```

**Arguments**

object            A sites object

**Value**

datasets from a sites object

---

datasets-class        *S4 class for datasets information*

---

**Description**

The grouped class for datasets from the Neotoma Paleoecology Database.

**Value**

object of class datasets

---

doi                    *Obtain the DOI for publications or datasets.*

---

**Description**

Obtain the DOI for publications or datasets.

**Usage**

```
doi(x)
```

**Arguments**

x                    Object with DOIs associated to it.

**Value**

doi object with DOI information

---

doi,publication-method

*Get a publication DOI.*

---

**Description**

Get a publication DOI.

**Usage**

```
## S4 method for signature 'publication'
doi(x)
```

**Arguments**

x                    A publication object.

**Value**

DOI from a publication

---

doi,site-method

*Obtain dataset DOIs from records.*

---

**Description**

Given complete dataset objects in Neotoma (must have used `get_datasets()` or `get_downloads()`), return the dataset DOI for the record.

**Usage**

```
## S4 method for signature 'site'
doi(x)
```

**Arguments**

x                    a Neotoma2 site object

**Value**

data.frame object with DOIs information.



**Examples**

```
{
  ds <- get_datasets(1)
  doi(ds)
}
```

---

doi,sites-method	<i>Obtain dataset DOIs from records.</i>
------------------	--

---

**Description**

Given complete dataset objects in Neotoma (must have used `get_datasets()` or `get_downloads()`), return the dataset DOI for the record.

**Usage**

```
## S4 method for signature 'sites'
doi(x)
```

**Arguments**

x                    a Neotoma2 site object

**Value**

data.frame object with DOIs information.

**Examples**

```
{
  ds <- get_datasets(1)
  doi(ds)
}
```

---

filter	<i>Apply a filter for Neotoma sites objects.</i>
--------	--

---

**Description**

The `filter` function takes a `sites` object and allows a user to filter on a number of properties. Since a `sites` object is a nested object (it contains collection units, datasets, samples, etc.) the degree to which filtering occurs depends on the amount of data contained within the `sites` object. Filtering parameters include:

- `siteid` A numeric site identifier from the Neotoma Database.
- `sitename` The character string sitename.

- `lat` A numeric latitude value.
- `long` A numeric longitude value.
- `altitude` The elevation of the site. Note that some sites do not include elevation information. For these an NA value appears, and they would be removed when using an elevation filter.
- `datasetid` A numeric datasetid from Neotoma.
- `database` A character string naming the constituent database from which the dataset is drawn.
- `datasettype` A character string representing one of the many dataset types within Neotoma.
- `age_range_old` A dataset-level parameter indicating the oldest date covered by the dataset chronology.
- `age_range_young` A dataset-level parameter indicating the youngest date covered by the dataset chronology.
- `notes` Free-form dataset notes provided by the dataset PI(s), analysts or data stewards.
- `collectionunitid` A numeric collection unit identifier from Neotoma.
- `handle` A character string identifying the collection unit. These are often shorter form names (originally a default 8 character length).
- `collectionunitname` A character string identifying the collection unit name.
- `colldate` The date on which the collection unit was sampled. Many of these are empty.
- `location` A free-form character string indicating the location of the collection unit within the site.
- `waterdepth` A numeric depth at which the core was obtained.
- `collunittype` A character string for the collection unit type.
- `collectiondevice` A fixed vocabulary term for the collection device.
- `depositionalenvironment` A fixed vocabulary name for the depositional environment.

### Usage

```
filter(x, ...)
```

### Arguments

<code>x</code>	A site, dataset or download.
<code>...</code>	arguments to filter by.

### Value

filtered sites object

---

 filter.sites

 Apply a filter for Neotoma sites objects.
 

---

### Description

The filter function takes a sites object and allows a user to filter on a number of properties. Since a sites object is a nested object (it contains collection units, datasets, samples, etc.) the degree to which filtering occurs depends on the amount of data contained within the sites object. Filtering parameters include:

- siteid A numeric site identifier from the Neotoma Database.
- sitename The character string sitename.
- lat A numeric latitude value.
- long A numeric longitude value.
- altitude The elevation of the site. Note that some sites do not include elevation information. For these an NA value appears, and they would be removed when using an elevation filter.
- datasetid A numeric datasetid from Neotoma.
- database A character string naming the constituent database from which the dataset is drawn.
- datasettype A character string representing one of the many dataset types within Neotoma.
- age\_range\_old A dataset-level parameter indicating the oldest date covered by the dataset chronology.
- age\_range\_young A dataset-level parameter indicating the youngest date covered by the dataset chronology.
- notes Free-form dataset notes provided by the dataset PI(s), analysts or data stewards.
- collectionunitid A numeric collection unit identifier from Neotoma.
- handle A character string identifying the collection unit. These are often shorter form names (originally a default 8 character length).
- collectionunitname A character string identifying the collection unit name.
- colldate The date on which the collection unit was sampled. Many of these are empty.
- location A free-form character string indicating the location of the collection unit within the site.
- waterdepth A numeric depth at which the core was obtained.
- collunittype A character string for the collection unit type.
- collectiondevice A fixed vocabulary term for the collection device.
- depositionalenvironment A fixed vocabulary name for the depositional environment.

### Usage

```
## S3 method for class 'sites'
filter(x, ...)
```

**Arguments**

x                    A sites object.  
 ...                 arguments to filter by.

**Value**

filtered sites object

**Examples**

```
# Download 100 sites, but only keep the sites that are close to sea level.
some_sites <- get_sites(sitename = "Lake%", limit = 3)
site_subset <- some_sites %>% filter(altitude < 100)
# Download 100 sites, get all associated datasets, but keep only
# sites/datasets that are of datasettype "pollen":
sites <- get_sites(limit = 1) %>%
  get_datasets(all_data = TRUE)
pollen_subset <- sites %>% filter(datasettype == "pollen")
```

---

getids

*Get object IDs*

---

**Description**

This function parses a site object, from site to dataset level and returns a data.frame that contains the site, collectionunit and dataset IDs for each element within the site.

**Usage**

```
getids(x, order = TRUE)
```

**Arguments**

x                    A Neotoma2 sites object.  
 order               sort items by siteid, collunitid, datasetid

**Value**

data.frame containing siteid, datasetid, and collunitid

---

getids.collunit	<i>Get object IDs from a single collectionunit.</i>
-----------------	---

---

**Description**

From a collectionunit object, return the collectionunit and dataset ids.

**Usage**

```
## S3 method for class 'collunit'  
getids(x, order = TRUE)
```

**Arguments**

x	A Neotoma2 collunit object.
order	sort items by siteid, collunitid, datasetid

**Value**

data.frame containing siteid, datasetid, and collunitid

**Examples**

```
{  
  marion <- get_sites(sitename = "Marion Lake")  
  collunitids <- getids(collunits(marion)[[1]])  
}
```

---

getids.collunits	<i>Get object IDs from collectionunits.</i>
------------------	---

---

**Description**

From a set of collectionunit objects, return the collectionunit and dataset ids.

**Usage**

```
## S3 method for class 'collunits'  
getids(x, order = TRUE)
```

**Arguments**

x	A Neotoma2 collunits object.
order	sort items by siteid, collunitid, datasetid

**Value**

data.frame containing siteid, datasetid, and collunitid

**Examples**

```
{
marion <- get_sites(sitename = "Marion Lake")
collunitids <- getids(collunits(marion))
}
```

---

getids.site	<i>Get object IDs from a site object.</i>
-------------	---

---

**Description**

Get object IDs from a site object.

**Usage**

```
## S3 method for class 'site'
getids(x, order = TRUE)
```

**Arguments**

x	A Neotoma2 site object.
order	sort items by siteid, collunitid, datasetid

**Value**

data.frame containing siteid, datasetid, and collunitid

---

getids.sites	<i>Get object IDs from sites</i>
--------------	----------------------------------

---

**Description**

Get object IDs from sites

**Usage**

```
## S3 method for class 'sites'
getids(x, order = TRUE)
```

**Arguments**

x                    A Neotoma2 sites object.  
 order                sort items by siteid, collunitid, datasetid

**Value**

data.frame containing siteid, datasetid, and collunitid

---

get\_contacts                    *Get contact information for Neotoma contributors*

---

**Description**

Uses the Neotoma API to search and access information about individuals who have contributed to the data in the Neotoma Paleoecology Database

**Usage**

```
get_contacts(x = NA, ...)
```

**Arguments**

x                    integer A contact ID  
 ...                    (contactname) A full or partial name for an individual contributor to the database.  
                       (familyname) The full or partial last name for an individual contributor to the  
                       database. (status) The current status of the contributor (active or retired)

**Value**

contacts object

---

get\_contacts.default                    *Get contact information for Neotoma contributors*

---

**Description**

Uses the Neotoma API to search and access information about individuals who have contributed to the data in the Neotoma Paleoecology Database

**Usage**

```
## Default S3 method:  

get_contacts(x, ...)
```

**Arguments**

x integer A contact ID

... (contactname) A full or partial name for an individual contributor to the database.  
 (familyname) The full or partial last name for an individual contributor to the database. (status) The current status of the contributor (active or retired)

**Value**

contacts object

---

*get\_contacts.numeric* *Get contact information for Neotoma contributors*

---

**Description**

Uses the Neotoma API to search and access information about individuals who have contributed to the data in the Neotoma Paleocology Database

**Usage**

```
## S3 method for class 'numeric'
get_contacts(x, ...)
```

**Arguments**

x integer A contact ID

... (contactname) A full or partial name for an individual contributor to the database.  
 (familyname) The full or partial last name for an individual contributor to the database. (status) The current status of the contributor (active or retired)

**Value**

contacts object



---

get_datasets	<i>get_datasets</i>
--------------	---------------------

---

## Description

The `get_datasets()` function is a wrapper for the Neotoma datasets API endpoint. The function takes parameters defined by the user and returns dataset information supplied by the Neotoma Paleocological Database. The user may define all or none of the possible fields.

## Usage

```
get_datasets(x = NA, ...)
```

## Arguments

<code>x</code>	A single datasetid, or a vector of unique dataset ids.
<code>...</code>	accepted arguments, see details for more information.

## Details

A dataset is an element nested within `neotoma2` site objects. The `get_datasets()` call returns a list of individual site objects with `collunits` (collection units) that contain valid, matching dataset elements. So, `get_sites()` returns only site metadata. `get_datasets()` returns site metadata, plus metadata about the individual datasets present at that site. The `get_datasets()` function searches for each site within Neotoma that matches the query parameters, and returns them as a `sites` object, a list of site objects, plus returns all the additional metadata for the datasets at that site. The `get_datasets()` command wraps the Neotoma API ([api.neotomadb.org](http://api.neotomadb.org)) call for datasets. The call itself uses a SQL query which accepts any one of the following parameters:

- `siteid` The unique site ID (integer) in Neotoma. Can be passed as a vector of site IDs.
- `sitename` The site name, or approximate match using the `%` wildcard.
- `database` The constituent database for the record. See `get_table("constituentdatabases")`
- `datasettype` Neotoma contains data for a number of dataset types. This returns a subset of data types. For a complete list of available dataset types, run `neotoma2::get_table('datasettypes')`
- `altmin` The minimum altitude range for site elevation (in meters).
- `altmax` The maximum altitude range for site elevation (in meters).
- `datasetid` The unique dataset ID (integer) in Neotoma. Can be passed as a vector of dataset IDs.
- `doi` The dataset DOI for a dataset contained within a site. Can be passed as a vector of DOIs.
- `gpuid` The geopolitical name or identifier containing a site. Can be passed as a vector of names.
- `keywords` Keywords for samples within a set of sites. For example "modern" indicates a sample within the record uses the keyword "modern".
- `contacts` Contact names or IDs associated with a site.
- `ageyoung` A minimum spanning age for the record, in years before radiocarbon present (1950).

- ageold A maximum spanning age for the record, in years before radiocarbon present (1950).
- ageof An age which must be contained within the range of sample ages for a site.
- taxa The names of taxa which must be present within samples in a record.
- all\_data The API only downloads the first 25 records of the query. For the complete records, use all\_data=TRUE

### Value

The function returns either a single item of class "try-error" describing the reason for failure (either misdefined parameters or an error from the Neotoma API), or list of site objects, each containing one or more collunit objects, with fully populated datasets elements.

### Author(s)

Socorro Dominguez <s.dominguez@ht-data.com>

### Examples

```
# To find all datasets with a min altitude of 12 and a max altitude of 25:
sites_12to25 <- get_datasets(altmin=12, altmax=25)
# To find all datasets in Brazil
brazil <- '{"type": "Polygon",
"coordinates": [[
  [-73.125, -9.102096738726443],
  [-56.953125, -33.137551192346145],
  [-36.5625, -7.710991655433217],
  [-68.203125, 13.923403897723347],
  [-73.125, -9.102096738726443]]]}'
brazil_datasets <- get_datasets(loc = brazil[1], limit=2)
# To obtain the dataset metadata:
datasets(brazil_datasets)
# There is insufficient metadata at this point to obtain information
# about taxa present at the site. We must use get_downloads() to
# obtain the full set of sample information:
# This fails: taxa(brazil_datasets)
```

---

get\_datasets.default *Get Dataset Default*

---

### Description

Get Dataset Default

### Usage

```
## Default S3 method:
get_datasets(x, ...)
```

**Arguments**

x                    Use a single number to extract site information  
...                    accepted arguments, see details for more information.

**Value**

sites object with full metadata up to the dataset level

**Examples**

```
{
# To find all datasets with a min altitude of 12 and a max altitude of 25:
sites_12to25 <- get_datasets(altmin=12, altmax=25, limit=2)
# To find all datasets in Brazil
brazil <- '{"type": "Polygon",
"coordinates": [[
[-73.125, -9.102096738726443],
[-56.953125, -33.137551192346145],
[-36.5625, -7.710991655433217],
[-68.203125, 13.923403897723347],
[-73.125, -9.102096738726443]]]}'
brazil_datasets <- get_datasets(loc = brazil[1], limit=2)
}
```

---

get\_datasets.numeric    *Get Dataset Numeric*

---

**Description**

Get Dataset Numeric

**Usage**

```
## S3 method for class 'numeric'
get_datasets(x, ...)
```

**Arguments**

x                    Use a single number to extract site information  
...                    Additional parameters to get\_datasets

**Value**

sites object with full metadata up to the dataset level

**Examples**

```
allds <- get_datasets(1:3)
```

get\_datasets.site      *Get Dataset from a site object.*

---

**Description**

Get Dataset from a site object.

**Usage**

```
## S3 method for class 'site'  
get_datasets(x, ...)
```

**Arguments**

x                      An object of class site.  
...                    additional arguments accepted by get\_datasets()

**Value**

sites object with full metadata up to the dataset level

**Examples**

```
random_sites <- get_sites(1)  
alllds <- get_datasets(random_sites, limit=3)
```

---

get\_datasets.sites      *Get Dataset from a sites object.*

---

**Description**

Get Dataset from a sites object.

**Usage**

```
## S3 method for class 'sites'  
get_datasets(x, ...)
```

**Arguments**

x                      An object of class sites.  
...                    additional arguments accepted by get\_datasets()

**Value**

sites object with full metadata up to the dataset level

**Examples**

```
random_sites <- get_sites(1)
alllds <- get_datasets(random_sites, limit=3)
```

---

<code>get_downloads</code>	<i>get_downloads</i>
----------------------------	----------------------

---

**Description**

Information for Fossil Datasets

**Usage**

```
get_downloads(x = NA, verbose = TRUE, ...)
```

**Arguments**

<code>x</code>	Use a single number to extract site information
<code>verbose</code>	Status bar of items being downloaded
<code>...</code>	accepted arguments: sites, datasets

**Details**

The `get_downloads()` command wraps the Neotoma API ([api.neotomadb.org](http://api.neotomadb.org)) call for downloads. The call itself uses a SQL query which accepts any one of the following parameters:

- `datasetid` The unique dataset ID (integer) in Neotoma. Can be passed as a vector of dataset IDs.
- `all_data` The API only downloads the first 25 records of the query. For the complete records, use `all_data=TRUE`

**Value**

The function returns either a single item of class "try-error" describing the reason for failure (either misdefined parameters or an error from the Neotoma API), or a table of sites, with rows corresponding to the number of individual sites and datasets returned by the Neotoma API. Each "site" object contains 6 parameters that can be accessed as well:

<code>siteid</code>	site ID number
<code>sitename</code>	site's name
<code>location</code>	sf object that describes site's location

description  
 collunits        limited information on collunits

Each "collection unit" embedded in the "sites" object contains 6 parameters that can be accessed as well:

collunitid       collection unit ID number  
 handle           collection unit's handle  
 collunitname    collection unit's name  
 colldate        date in collection unit  
 substrate        substrate  
 location        sf object that describes site's location  
 datasets        detailed information regarding dataset

Each "dataset" nested in the "collection unit" contains the following detail of information:

datasetid        dataset ID number  
 datasetname     site's name  
 datasettype     type of data found  
 location        sf object that describes site's location  
 notes           notes on the dataset  
 taxa table      taxa table  
 pi list         P.I. info  
 analyst         analyst info  
 metadata        dataset metadata

### Author(s)

Socorro Dominguez <s.dominguez@ht-data.com>

### Examples

```
# To find the downloads object of dataset 24:
downloads24 <- get_downloads(24)

# To find all downloads in Brazil
brazil <- '{"type": "Polygon",
"coordinates": [[
[-73.125, -9.102096738726443],
[-56.953125, -33.137551192346145],
[-36.5625, -7.710991655433217],
[-68.203125, 13.923403897723347],
[-73.125, -9.102096738726443]]]}'
brazil_datasets <- get_datasets(loc = brazil[1])
brazil_downloads <- get_downloads(brazil_datasets)
```

---

```
get_downloads.character  
  get_downloads JSON
```

---

**Description**

get\_downloads JSON

**Usage**

```
## S3 method for class 'character'  
get_downloads(x, verbose = TRUE, ...)
```

**Arguments**

x	sites object
verbose	Should text be printed during the download process?
...	arguments in ellipse form

**Value**

The function returns either a single item of class "try-error" describing the reason for failure (either misdefined parameters or an error from the Neotoma API), or a table of sites, with rows corresponding to the number of individual sites and datasets returned by the Neotoma API.

---

```
get_downloads.numeric  get_downloads
```

---

**Description**

get\_downloads

**Usage**

```
## S3 method for class 'numeric'  
get_downloads(x, verbose = TRUE, ...)
```

**Arguments**

x	Use a single number to extract site information
verbose	Should text be printed during the download process?
...	arguments in ellipse form

**Value**

The function returns either a single item of class "try-error" describing the reason for failure (either misdefined parameters or an error from the Neotoma API), or a table of sites, with rows corresponding to the number of individual sites and datasets returned by the Neotoma API.

---

get\_downloads.sites     *get\_downloads sites*

---

**Description**

get\_downloads sites

**Usage**

```
## S3 method for class 'sites'
get_downloads(x, verbose = TRUE, ...)
```

**Arguments**

x	sites object
verbose	Should text be printed during the download process?
...	arguments in ellipse form

**Value**

The function returns either a single item of class "try-error" describing the reason for failure (either misdefined parameters or an error from the Neotoma API), or a table of sites, with rows corresponding to the number of individual sites and datasets returned by the Neotoma API.

---

get\_manual                     *get\_manual*

---

**Description**

Open up the Neotoma manual homepage.

**Usage**

```
get_manual()
```

**Value**

NULL side effect for opening browser with the manual



**Author(s)**

Simon Goring <goring@wisc.edu>

**Examples**

```
{
# This call does not work from `source()` calls or in testing.
# interactive() just lets us know you are interacting with the console:
if (interactive()) {
  get_manual()
}
}
```

---

get\_publications      *Get publication information for Neotoma records*

---

**Description**

Uses the Neotoma API to search and access information about publications associated with data in the Neotoma Paleocology Database

**Usage**

```
get_publications(x = NA, ...)
```

**Arguments**

x	integer A contact ID
...	publicationid The unique numeric identifier associated with a publication in Neotoma. datasetid A unique identifier for a Neotoma dataset that is associated with a publication. familyname The full or partial last name for an individual author. pubtype The publication type, from get_tables("publicationtypes"). year The year the publication was released. search A plain text search string used to search the citation.

**Value**

publications object

**Examples**

```
# How old are the papers in Neotoma that include the term "mammut"?
mammoth_papers <- get_publications(search="mammut") %>%
  as.data.frame()
hist(as.numeric(mammoth_papers$year))
```

---

```
get_publications.default
```

*Get publication information from Neotoma*

---

### Description

Get publication information from Neotoma

### Usage

```
## Default S3 method:
get_publications(...)
```

### Arguments

```
...      publicationid The unique numeric identifier associated with a publication in
          Neotoma. datasetid A unique identifier for a Neotoma dataset that is associ-
          ated with a publication. familyname The full or partial last name for an individ-
          ual author. pubtype The publication type, from get_tables("publicationtypes").
          year The year the publication was released. search A plain text search string
          used to search the citation.
```

### Value

publications object

### Examples

```
# How old are the papers in Neotoma that include the term "mammut"?
mammoth_papers <- get_publications(search="mammut") %>%
  as.data.frame()
hist(as.numeric(mammoth_papers$year))
```

---

```
get_publications.numeric
```

*Get publications using their unique identifier.*

---

### Description

Get publications using their unique identifier.

### Usage

```
## S3 method for class 'numeric'
get_publications(x, ...)
```

**Arguments**

x integer A contact ID  
 ... publicationid The unique numeric identifier associated with a publication in Neotoma. datasetid A unique identifier for a Neotoma dataset that is associated with a publication. familyname The full or partial last name for an individual author. pubtype The publication type, from get\_tables("publicationtypes"). year The year the publication was released. search A plain text search string used to search the citation.

**Value**

publications object

**Examples**

```
{
# We want the paper identified in Neotoma as 666:
get_publications(666)
}
```

---

get\_publications.publication

*Update information for a publications object.*

---

**Description**

This works for records without publicationids. We assume that data with publicationids is correct.

**Usage**

```
## S3 method for class 'publication'
get_publications(x, ...)
```

**Arguments**

x integer A publication  
 ... publicationid The unique numeric identifier associated with a publication in Neotoma. datasetid A unique identifier for a Neotoma dataset that is associated with a publication. familyname The full or partial last name for an individual author. pubtype The publication type, from get\_tables("publicationtypes"). year The year the publication was released. search A plain text search string used to search the citation.

**Value**

updated publication object

**Examples**

```
# Take a publication object and purposely degrade the metadata:
bad_pub <- get_publications(666)
# Note this only changes the reported year, not the citation string.
bad_pub[[1]]@year <- "1923"
bad_pub[[1]]@publicationid <- as.character(NA)
updated_pubs <- get_publications(bad_pub[[1]])
attr(updated_pubs, "matches")
# we see the proper citation in the record:
updated_pubs <- attr(updated_pubs, "matches")[[3]]
```

---

```
get_publications.publications
```

*Update metadata for a set of publication objects.*

---

**Description**

Update metadata for a set of publication objects.

**Usage**

```
## S3 method for class 'publications'
get_publications(x, ...)
```

**Arguments**

x	integer A publication
...	publicationid The unique numeric identifier associated with a publication in Neotoma. datasetid A unique identifier for a Neotoma dataset that is associated with a publication. familyname The full or partial last name for an individual author. pubtype The publication type, from get_tables("publicationtypes"). year The year the publication was released. search A plain text search string used to search the citation.

**Value**

publications object

**Examples**

```
# Take a publication object and purposely degrade the metadata:
bad_pub <- get_publications(c(666, 667, 668))
# Note this only changes the reported year, not the citation string.
bad_pub[[1]]@year <- "1923"
bad_pub[[1]]@publicationid <- as.character(NA)
updated_pubs <- get_publications(bad_pub)
# Only the first publication object has any matches. It's the only one
```

```
# that is missing its publicaitonid.
attr(updated_pubs[[1]], "matches")
attr(updated_pubs[[2]], "matches")
# we see the proper citation in the record:
updated_pubs[[1]] <- attr(updated_pubs[[1]], "matches")[[1]]
```

---

get\_sites

*get\_sites*


---

## Description

The `get_sites()` function is a wrapper for the Neotoma sites API endpoint. The function takes parameters defined by the user and returns a list of site information supplied by the Neotoma Paleocological Database. The user may define all or none of the possible fields.

## Usage

```
get_sites(x = NA, ...)
```

## Arguments

<code>x</code>	Use a single integer or vector of integers representing unique Neotoma site identifiers (siteids) to extract site information.
<code>...</code>	accepted arguments, see details for more information.

## Details

A site object in Neotoma is a physical location at which one or more collection units are located. Each collection unit may have one or more datasets within it, defined by the dataset type. The `get_sites()` function searches for each site within Neotoma that matches the query parameters, and returns them as a `sites` object, a list of site objects. The `get_sites()` command wraps the Neotoma API ([api.neotomadb.org](http://api.neotomadb.org)) call for sites. The call itself uses a SQL query which accepts any one of the following parameters:

- `siteid` The unique site ID (integer) in Neotoma. Can be passed as a vector of site IDs.
- `sitename` The site name, or approximate match using the `%` wildcard.
- `database` The constituent database for the record. See `get_table("constituentdatabases")`
- `altmin` The minimum altitude range for site elevation (in meters).
- `altmax` The maximum altitude range for site elevation (in meters).
- `datasetid` The unique dataset ID (integer) in Neotoma. Can be passed as a vector of dataset IDs.
- `datasettype` Neotoma contains data for a number of datasettypes. This returns a subset of data types. For a complete list of available datasettypes, run `neotoma2::get_table('datasettypes')`
- `doi` The dataset DOI for a dataset contained within a site. Can be passed as a vector of DOIs.

- `gpid` The geopolitical name or identifier containing a site. Can be passed as a vector of names.
- `keywords` Keywords for samples within a set of sites. For example "modern" indicates a sample within the record uses the keyword "modern".
- `contacts` Contact names or IDs associated with a site.
- `ageyoung` A minimum spanning age for the record, in years before radiocarbon present (1950).
- `ageold` A maximum spanning age for the record, in years before radiocarbon present (1950).
- `ageof` An age which must be contained within the range of sample ages for a site.
- `taxa` The names of taxa which must be present within samples in a record.
- `all_data` The API only downloads the first 25 records of the query. For the complete records, use `all_data=TRUE` This call will then return a data object that contains site metadata for one or more sites, along with limited metadata describing the collection units and datasets located at that site.

### Value

The function returns either a single item of class "try-error" describing the reason for failure (either misdefined parameters or an error from the Neotoma API), or a table of sites, with rows corresponding to the number of individual sites returned by the Neotoma API. Each "site" object contains 6 parameters that can be accessed as well: `siteid`, `sitename`, `location`, `altitude`, `description`, limited collection units information.

- `loc` An `sf` object that describes site's location.
- `description`
- `collunits` limited information on collunits

### Author(s)

Socorro Dominguez <s.dominguez@ht-data.com>

### Examples

```
## Find all sites with a min altitude of 12m and a max altitude of 25m
## By default returns only 25 sites (default limit is 25):
sites_12to25 <- get_sites(altmin=12, altmax=25)
## Return all sites, using a minimum altitude of 2500m (returns >500 sites):
sites_2500 <- get_sites(altmin=2500, all_data = TRUE)
## To find all sites that contain the string "Alex%"
alex_sites <- get_sites(sitename="Alex%")
## To find sites in Brazil (again with default 25 records)
brazil <- '{"type": "Polygon",
"coordinates": [[
[-73.125, -9.102096738726443],
[-56.953125, -33.137551192346145],
[-36.5625, -7.710991655433217],
[-68.203125, 13.923403897723347],
[-73.125, -9.102096738726443]]]}'
brazil_sites <- get_sites(loc = brazil[1])
```

```
# Finding all sites with Liliaceae pollen in 1000 year bins:
lilysites <- c()
for (i in seq(0, 10000, by = 1000)) {
  lily <- get_sites(taxa=c("Liliaceae"),
                   ageyoung = i - 500,
                   ageold = i + 500,
                   all_data = TRUE)
  lilysites <- c(lilysites, length(lily))
}
plot(x = seq(0, 10000, by = 1000), y = lilysites, type = 'b')
```

---

get\_sites.default      *get\_sites*

---

## Description

get\_sites

## Usage

```
## Default S3 method:
get_sites(...)
```

## Arguments

...                    One of a set of possible query parameters discussed in details.

## Value

The function returns either a single item of class "try-error" describing the reason for failure (either misdefined parameters or an error from the Neotoma API), or a table of sites, with rows corresponding to the number of individual sites returned by the Neotoma API. Each "site" object contains 6 parameters that can be accessed as well: siteid, sitename, location, altitude, description, limited collection units information.

- loc An sf object that describes site's location.
- collunits limited information on collunits

## Author(s)

Socorro Dominguez <s.dominguez@ht-data.com>

---

get\_sites.numeric      *Get Site Information for Fossil Sites*

---

### Description

Get Site Information for Fossil Sites

### Usage

```
## S3 method for class 'numeric'
get_sites(x, ...)
```

### Arguments

x	The numeric site ID from Neotoma
...	accepted arguments if numeric all_data

### Value

The function returns either a single item of class "try-error" describing the reason for failure (either misdefined parameters or an error from the Neotoma API), or a table of sites, with rows corresponding to the number of individual sites returned by the Neotoma API. Each "site" object contains 6 parameters that can be accessed as well: siteid, sitename, location, altitude, description, limited collection units information.

- loc An sf object that describes site's location.
- collunits limited information on collunits

### Examples

```
{
## Find all sites by numeric siteid:
sites <- get_sites(seq(1,3))
}
```

---

get\_sites.sites      *Get Site Information for Fossil Sites from a Set of Sites*

---

### Description

Get Site Information for Fossil Sites from a Set of Sites

### Usage

```
## S3 method for class 'sites'
get_sites(x, ...)
```



**Arguments**

x                    The numeric site ID from Neotoma  
 ...                  accepted arguments if numeric all\_data

**Value**

The function returns either a single item of class "try-error" describing the reason for failure (either misdefined parameters or an error from the Neotoma API), or a table of sites, with rows corresponding to the number of individual sites returned by the Neotoma API. Each "site" object contains 6 parameters that can be accessed as well: siteid, sitename, location, altitude, description, limited collection units information.

- loc An sf object that describes site's location.
- collunits limited information on collunits

**Examples**

```
## Find all sites using a set of prior sites:
char_sites <- get_sites(taxa = "charcoal")
pollen_coloc <- get_sites(char_sites, datasettype = "pollen")
char_coloc <- char_sites %>% filter(siteid %in% getids(pollen_coloc)$siteid)
pol_char <- c(pollen_coloc, char_coloc)
```

---

get_specimens	<i>get_specimens</i>
---------------	----------------------

---

**Description**

Information for Specimens

**Usage**

```
get_specimens(x = NA, ...)
```

**Arguments**

x                    Use a single specimenid  
 ...                  Additional terms passed to get\_specimens, most common datasetid

**Value**

The function returns a specimens list

**Author(s)**

Socorro Dominguez <s.dominguez@ht-data.com>

**Examples**

```
# To find specimen with ID 7:
my_specimens <- get_specimens(7)
# To find specimens in datasetid 41610
my_specimens2 <- get_specimens(datasetid = 41610)
```

---

get\_specimens.default *Get Specimen datasetid*

---

**Description**

Get Specimen datasetid

**Usage**

```
## Default S3 method:
get_specimens(...)
```

**Arguments**

... Pass argument datasetid and the corresponding datasetid

**Value**

The function returns a specimens list

**Examples**

```
{
# To find specimens in datasetid 41610
my_specimens <- get_specimens(datasetid = 41610)
}
```

---

get\_specimens.numeric *Get Specimen Numeric*

---

**Description**

Get Specimen Numeric

**Usage**

```
## S3 method for class 'numeric'
get_specimens(x, ...)
```

**Arguments**

- x                    Use a single number to extract site information
- ...                  Additional terms passed to *get\_specimens*.

**Value**

The function returns a specimens list

**Examples**

```
{  
## To find specimen with ID 7  
my_specimens <- get_specimens(7)  
}
```

---

*get\_specimens.sites*    *Get Specimen Sites*

---

**Description**

Get Specimen Sites

**Usage**

```
## S3 method for class 'sites'  
get_specimens(x, ...)
```

**Arguments**

- x                    Use a single number to extract site information
- ...                  Other possible parameters such as *datasetid*

**Value**

The function returns a specimens list

**Examples**

```
# To find specimen with ID 7:  
my_site <- get_sites(13296)  
# To find specimens in `my_site`  
my_specimens <- get_specimens(my_site)
```

---

get_stats	<i>get_stats</i>
-----------	------------------

---

### Description

Returns a count of sites, datasets, publications and other objects added to Neotoma during the requested time period.

### Usage

```
get_stats(start, end)
```

### Arguments

start	The starting month (from present == 0) for which to generate the summary.
end	The ending month (from present == 0) for which to generate the summary.

### Details

This function returns summaries about the data holdings within Neotoma using the existing Neotoma API's summary endpoint. This can provide information about recent uploads (the number of new sites uploaded within the last month, for example), or can be used to provide information about the overall number of sites/datasets (using an arbitrarily high value for end).

### Value

data.frame with summary statistics

### Author(s)

Socorro Dominguez <s.dominguez@ht-data.com>

Simon Goring <goring@wisc.edu>

### Examples

```
last_month <- get_stats(start = 0, end = 1)
```

---

get_table	<i>Get table record from Neotoma</i>
-----------	--------------------------------------

---

**Description**

Call Neotoma and return a table (with limits & offsets for large tables)

**Usage**

```
get_table(x, limit = 25, offset = 0)
```

**Arguments**

x	Table name (consult <a href="https://open.neotomadb.org/dbschema/">https://open.neotomadb.org/dbschema/</a> for a complete list of table names.
limit	Default 25 records
offset	Default 0.

**Value**

selected table values from the Database

**Examples**

```
{
# Returns only the first 25 specimen records.
someSpec <- get_table('specimens')
}
```

---

length,chronologies-method	<i>Length Method chronologies</i>
----------------------------	-----------------------------------

---

**Description**

Length Method chronologies

**Usage**

```
## S4 method for signature 'chronologies'
length(x)
```

**Arguments**

x	chronologies object
---	---------------------

**Value**

integer describing length

---

length,collunits-method

*Length Method collunits*

---

**Description**

Length Method collunits

**Usage**

```
## S4 method for signature 'collunits'  
length(x)
```

**Arguments**

x collunits object

**Value**

length of a collunits object

---

length,datasets-method

*Length Method datasets*

---

**Description**

Length Method datasets

**Usage**

```
## S4 method for signature 'datasets'  
length(x)
```

**Arguments**

x datasets object

**Value**

int that showcases the length of a datasets object

---

length,publications-method

*Get the number of publications in a publications object.*

---

**Description**

Get the number of publications in a publications object.

**Usage**

```
## S4 method for signature 'publications'  
length(x)
```

**Arguments**

x                    A publications object.

**Value**

int of the length of the publications object

---

length,samples-method *Length Method samples*

---

**Description**

Length Method samples

**Usage**

```
## S4 method for signature 'samples'  
length(x)
```

**Arguments**

x                    samples object

**Value**

int representing the length of samples object

length, sites-method    *Length Method Sites*

---

**Description**

Length Method Sites

**Usage**

```
## S4 method for signature 'sites'  
length(x)
```

**Arguments**

x                    sites object

**Value**

int with the length of sites object

---

length, specimens-method    *Length Method specimens*

---

**Description**

Length Method specimens

**Usage**

```
## S4 method for signature 'specimens'  
length(x)
```

**Arguments**

x                    specimens object

**Value**

int with length of specimens object



---

missingOrNULL-class    *c Method - Combine objects, including NULL*

---

**Description**

c Method - Combine objects, including NULL  
 c Method - Combine objects, including NULL

---

names,collunit-method    *Get slot names*

---

**Description**

Get all names for named elements within a collunit object.

**Usage**

```
## S4 method for signature 'collunit'
names(x)
```

**Arguments**

x                      A collection unit object.

**Value**

NULL. Shows the names of the slots

---

names,contact-method    *Get names of contacts slots*

---

**Description**

Get names of contacts slots

**Usage**

```
## S4 method for signature 'contact'
names(x)
```

**Arguments**

x                      A contact object.

**Value**

names of slots

names,dataset-method *Get slot names*

---

**Description**

Get all names for named elements within a dataset object.

**Usage**

```
## S4 method for signature 'dataset'  
names(x)
```

**Arguments**

x                    A dataset object.

**Value**

list with all names of dataset slots

---

names,publication-method  
*Get slot names for a publication object.*

---

**Description**

Get slot names for a publication object.

**Usage**

```
## S4 method for signature 'publication'  
names(x)
```

**Arguments**

x                    A publication object.

**Value**

string with publication slots' names

---

names,publications-method

*Get slot names for a publication object.*

---

### **Description**

Get slot names for a publication object.

### **Usage**

```
## S4 method for signature 'publications'  
names(x)
```

### **Arguments**

x                    A publications object.

### **Value**

string with publications slots' names

---

names,site-method

*Get slot names*

---

### **Description**

Get all names for named elements within a site object.

### **Usage**

```
## S4 method for signature 'site'  
names(x)
```

### **Arguments**

x                    A site object.

### **Value**

names of the slots of a site object

---

names,specimen-method *Get slot names*

---

### Description

Get all names for named elements within a specimen object.

### Usage

```
## S4 method for signature 'specimen'
names(x)
```

### Arguments

x                    A specimen object.

### Value

names of the slots of a site object

---

newURL                    *Format API call to Neotoma from call arguments*

---

### Description

Take a set of arguments from the Neotoma2 package and produce the appropriate URL to the Neotoma v2.0 API. This is an internal function used by parseURL().

### Usage

```
newURL(baseurl, args, ...)
```

### Arguments

baseurl                The base URL for the Neotoma API  
args                    The set of query arguments to be passed to the API  
...                     Any additional arguments to be passed to the function.

### Value

A properly formatted URL.

---

parseURL	<i>parseURL</i>
----------	-----------------

---

**Description**

An internal helper function used to connect to the Neotoma API in a standard manner, and to provide basic validation of any response.

**Usage**

```
parseURL(x, use = "neotoma", all_data = FALSE, ...)
```

**Arguments**

x	The HTTP/S path for the particular API call.
use	Uses the Neotoma server by default ("neotoma"), but supports either the development API server ("dev") or a local server ("local").
all_data	If TRUE return all possible API calls
...	Any query parameters passed from the calling function.

**Value**

list with cleaned and parsed data from HTTP request

**Author(s)**

Socorro Dominguez <s.dominguez@ht-data.com>  
Simon Goring <goring@wisc.edu>

---

parse_site	<i>parse_site</i>
------------	-------------------

---

**Description**

An internal helper function to parse the API result into a site object.

**Usage**

```
parse_site(result)
```

**Arguments**

result	A JSON object from the API.
--------	-----------------------------

**Value**

A Neotoma2 site object.

---

pingNeotoma                      *pingNeotoma*

---

**Description**

A quick function to test whether or not the Neotoma Database API is currently running.

**Usage**

```
pingNeotoma(server = "neotoma")
```

**Arguments**

server                      One of localhost:PORT (where PORT is a valid numeric port), neotoma or dev.

**Value**

A valid HTTP status code or returns an error if a connection is refused.

**Examples**

```
{
  test_connection <- pingNeotoma("neotoma")
}
```

---

*plot,sites,ANY-method*    *Plot site coordinates using a basic plot.*

---

**Description**

Plot site coordinates using a basic plot.

**Usage**

```
## S4 method for signature 'sites,ANY'
plot(x, y, ...)
```

**Arguments**

x                              sites object  
y                                ANY  
...                              Additional parameters associated with the call.

**Value**

plot object with site coordinates.

---

plot,sites-method	<i>Plot site coordinates using a basic plot.</i>
-------------------	--

---

**Description**

Plot site coordinates using a basic plot.

**Usage**

```
## S4 method for signature 'sites'  
plot(x, y, ...)
```

**Arguments**

x	sites object
y	ANY
...	Additional parameters associated with the call.

**Value**

plot object with site coordinates.

---

plotLeaflet	<i>plotLeaflet</i>
-------------	--------------------

---

**Description**

Plot sites on a leaflet map

**Usage**

```
plotLeaflet(object)
```

**Arguments**

object	Sites object to plot
--------	----------------------

**Value**

leaflet map with site markers

**Author(s)**

Socorro Dominguez <s.dominguez@ht-data.com>

---

plotLeaflet,site-method  
*plotLeaflet*

---

**Description**

Plot a site on a leaflet map

**Usage**

```
## S4 method for signature 'site'  
plotLeaflet(object)
```

**Arguments**

object            Site object to plot

**Value**

leaflet map

**Examples**

```
modernSites <- get_sites(keyword = "Modern")  
plotLeaflet(modernSites[[1]])
```

---

plotLeaflet,sites-method  
*plotLeaflet*

---

**Description**

Plot sites on a leaflet map

**Usage**

```
## S4 method for signature 'sites'  
plotLeaflet(object)
```

**Arguments**

object            Sites object to plot

**Value**

leaflet map



**Examples**

```
# Note that by default the limit for queries is 25 records:
modernSites <- get_sites(keyword = "Modern")
plotLeaflet(modernSites)
```

---

publication-class	<i>An S4 class for a single Neotoma publication.</i>
-------------------	--

---

**Description**

A publication is linked to an individual Neotoma dataset object. They are grouped using an S4 publications class.

**Value**

object of class publication

**Examples**

```
{
simon <- new("contact", familyname = "Goring", givennames = "Simon J.")
socorro <- new("contact", familyname = "Dominguez", givennames = "Socorro")
first_author <- new("author", author = simon, order = 1)
second_author <- new("author", author = socorro, order = 2)
author_list <- new("authors", authors = list(first_author, second_author))
pub <- new("publication",
  articletitle = "Top or bottom: Best toast spreading surfaces.",
  journal = "Peanut Butter Science",
  year = "2022",
  volume = "2",
  author = author_list)
}
```

---

publications-class	<i>An S4 class for multi-publication information from the Neotoma Paleocology Database. This S4 class allows a single dataset to have one or more publication classes associated with it.</i>
--------------------	---

---

**Description**

An S4 class for multi-publication information from the Neotoma Paleocology Database. This S4 class allows a single dataset to have one or more publication classes associated with it.

**Value**

object of class publications

---

repositories-class      *S4 class for repositories information*

---

**Description**

The grouped class for repositories from the Neotoma Paleoecology Database.

**Value**

object of class repositories

---

repository-class      *S4 class for repository information*

---

**Description**

The standard object class for repository from the Neotoma Paleoecology Database.

**Value**

object of class repository

---

sample-class      *S4 class for dataset information*

---

**Description**

The standard object class for samples from the Neotoma Paleoecology Database.

**Value**

object of class sample

---

samples	<i>Obtain samples from a record or multiple records.</i>
---------	--

---

**Description**

Obtain samples from a record or multiple records.

**Usage**

```
samples(x)
```

**Arguments**

x                    sites object

**Value**

data.frame with record information at sample level

---

samples, collunit-method	
	<i>samples</i>

---

**Description**

Obtain elements from collunit

**Usage**

```
## S4 method for signature 'collunit'  
samples(x)
```

**Arguments**

x                    collunit object

**Value**

data.frame with sample records

---

samples,collunits-method

*Get samples from a collectionunit or set of collection units:*

---

**Description**

Obtain elements from collunits

**Usage**

```
## S4 method for signature 'collunits'  
samples(x)
```

**Arguments**

x                    collunits object

**Value**

data.frame with sample records

---

samples,site-method    *samples*

---

**Description**

Obtain elements on the samples level

**Usage**

```
## S4 method for signature 'site'  
samples(x)
```

**Arguments**

x                    site object

**Value**

data.frame with sample records

**Examples**

```
marion <- get_sites(sitename = "Marion Lake") %>%  
  get_datasets() %>%  
  filter(datasettype == "pollen") %>%  
  get_downloads()  
pollen <- samples(marion)
```

---

samples, sites-method    *samples*

---

**Description**

Obtain all samples within a sites object

**Usage**

```
## S4 method for signature 'sites'  
samples(x)
```

**Arguments**

x                    sites object

**Value**

data.frame with sample records

**Examples**

```
{  
dw <- get_downloads(1)  
pollen <- samples(dw)  
}
```

---

samples-class                    *S4 class for the set of samples*

---

**Description**

The grouped class for samples from the Neotoma Paleoecology Database.

**Value**

object of class samples

---

selectMatch	<i>Select the best match for an object.</i>
-------------	---

---

**Description**

Select the best match for an object.

**Usage**

```
selectMatch(x, n)
```

**Arguments**

x	object
n	n elements that are a best match

**Value**

attr Select the match between a local record and a Neotoma match

---

selectMatch,publication,logical-method	<i>Select the best match (between a local record and a Neotoma match)</i>
--	---

---

**Description**

Select the best match (between a local record and a Neotoma match)

**Usage**

```
## S4 method for signature 'publication,logical'
selectMatch(x, n)
```

**Arguments**

x	A publication object
n	The match number (in the case an NA is returned).

**Value**

the best match to the selected publication.

---

```
selectMatch,publication,numeric-method
```

*Select the best match (between a local record and a Neotoma match)*

---

**Description**

Select the best match (between a local record and a Neotoma match)

**Usage**

```
## S4 method for signature 'publication,numeric'
selectMatch(x, n)
```

**Arguments**

x	A publication object
n	The match number.

**Value**

the best match to the selected publication.

---

```
set_chronology
```

*set chronology information for a new record.*

---

**Description**

Create a new chronology for a record. Within Neotoma all chronologies have unique numeric identifiers. Within R, because of the need to use the identifiers across objects, and because we want to avoid conflicts between naming systems, a universally unique identifier (UUID) is created for the object ID.

Function to create new chronology objects for personal analysis. The new object will not be uploaded to the database.

**Usage**

```
set_chronology(
  x = NA,
  chronologyid = NA_integer_,
  notes = NA_character_,
  contact = list(),
  agemodel = NA_character_,
  ageboundolder = NA_integer_,
  ageboundyounger = NA_integer_,
  isdefault = NA_integer_,
```

```

dateprepared = as.Date(character(0)),
modelagetype = NA_character_,
chronologyname = NA_character_,
chroncontrols = data.frame(0)
)

```

### Arguments

x	Object to be set as a chronology
chronologyid	An optional value. Will be assigned a unique identifier if not provided.
notes	Additional notes about the chronology. For more modern models, often the function call to Bacon or Bchron is added here.
contact	A contacts object, identifying the individual(s) who created the chronology
agemodel	A string representing the age model name, for example "Crummy linear interpolation".
ageboundolder	The ageboundolder is assigned the oldest sample age rounded up to the nearest 10
ageboundyounger	The ageboundyounger is assigned the oldest sample age rounded up to the nearest 10
isdefault	Defines whether the model is the default for the collection unit for a particular model age type.
dateprepared	The date at which the age model was prepared.
modelagetype	The age type for the model. For validation, the models should be one of the valid Neotoma agetypes: <a href="https://api.neotomadb.org/v2.0/data/dbtables?table=agetypes">https://api.neotomadb.org/v2.0/data/dbtables?table=agetypes</a>
chronologyname	A valid name for the chronology.
chroncontrols	A data.frame containing the chronological controls for the age model.

### Value

chronology object

---

set\_collunit

*set Site Information for Fossil Sites*

---

### Description

Function to create new collection unit objects for personal analysis. The new object will not be uploaded to the database.



**Usage**

```

set_collunit(
  x = NA,
  collectionunitid = NA_integer_,
  notes = NA_character_,
  handle = NA_character_,
  colldate = as.Date(character(1)),
  location = NA_character_,
  waterdepth = NA_integer_,
  gpslocation = st_as_sf(st_sfc()),
  collunittype = NA_character_,
  collectiondevice = NA_character_,
  collectionunitname = NA_character_,
  depositionalenvironment = NA_character_,
  datasets = new("datasets"),
  chronologies = new("chronologies"),
  defaultchronology = NA_integer_
)

```

**Arguments**

x	object to be set as collunit
collectionunitid	collection unit identifier
notes	notes
handle	handle
colldate	collection date
location	location of the collection unit
waterdepth	depth at where the sample is taken
gpslocation	location with GPS
collunittype	type of collection unit
collectiondevice	device used to collect the sample
collectionunitname	name of the collection unit
depositionalenvironment	depositional environment
datasets	datasets that the collection unit has
chronologies	chronologies taken from the collection unit
defaultchronology	best chronology model identifier to be used with this collection unit

**Value**

collunit object

**Examples**

```
{
# Create a collunit
my_collunit <- set_collunit(notes = "my lake")
}
```

---

set_contact	<i>Set contact information for a new record.</i>
-------------	--

---

**Description**

Create a new contact for a record. Within Neotoma all chronologies have unique numeric identifiers. Within R, because of the need to use the identifiers across objects, and because we want to avoid conflicts between naming systems, a universally unique identifier (UUID) is created for the object ID. This is not to be updated to the database.

**Usage**

```
set_contact(
  x = NA,
  contactid = NA_integer_,
  familyname = NA_character_,
  leadinginitials = NA_character_,
  givennames = NA_character_,
  suffix = NA_character_,
  ORCID = NA_character_,
  title = NA_character_,
  institution = NA_character_,
  email = NA_character_,
  phone = NA_character_,
  contactstatus = NA_character_,
  fax = NA_character_,
  url = NA_character_,
  address = NA_character_,
  notes = NA_character_
)
```

**Arguments**

x	Object to be set as a contact
contactid	An arbitrary Contact identification number.
familyname	Family or surname name of a person.
leadinginitials	Leading initials for given or forenames without spaces (e.g. G.G.).
givennames	Given or forenames of a person (e.g. George Gaylord). Initials with spaces are used if full given names are not known (e.g. G. G).

suffix	Suffix of a person's name (e.g. Jr., III).
ORCID	A unique ORCID (see <a href="https://orcid.org">https://orcid.org</a> ).
title	A person's title (e.g. Dr., Prof., Prof. Dr).
institution	The institution where an individual works.
email	An individuals email address
phone	Their phone number
contactstatus	Are they "active" or "retired"?
fax	Do people still use fax machines?
url	Their homepage
address	A physical address
notes	Notes about the individual

**Value**

contact object

---

set_dataset	<i>set Site Information for Fossil Sites</i>
-------------	--

---

**Description**

Function to create new dataset objects for personal analysis. The new object will not be uploaded to the database.

**Usage**

```
set_dataset(
  x = NA,
  datasetid = NA_integer_,
  database = NA_character_,
  doi = NA,
  datasettype = NA_character_,
  age_range_old = NA_integer_,
  age_range_young = NA_integer_,
  notes = NA_character_,
  pi_list = NA,
  samples = new("samples")
)
```

**Arguments**

x	object to be set as dataset,
datasetid	dataset identifier
database	dataset where the dataset came from
doi	DOI
datasettype	type the dataset belongs to
age_range_old	age range old
age_range_young	age range young
notes	notes
pi_list	pi list
samples	taxa objects

**Value**

dataset object

**Examples**

```
{
# Create a dataset
my_dataset <- set_dataset(database = "EPD",
                          datasettype = "pollen",
                          notes = "my lake")
}
```

---

set\_default

*Set the default chronology within a collectionunit.*

---

**Description**

Set the default chronology within a collectionunit.

**Usage**

```
set_default(x, n)
```

**Arguments**

x	A chronologies object.
n	The particular chronology to be used as the default.

**Value**

sites object with new default chronology

---

`set_default,chronologies-method`*Change the default age model for a record.*

---

**Description**

Change the default age model for a record.

**Usage**

```
## S4 method for signature 'chronologies'  
set_default(x, n)
```

**Arguments**

x	A chronologies object.
n	The particular chronology to be used as the default.

**Value**

chronologies object with a new defaulted chronology

---

`set_publications`      *Create a new publication (or publication set)*

---

**Description**

A function to create new publication objects by hand.

**Usage**

```
set_publications(  
  publicationid = NA_integer_,  
  publicationtypeid = NA_integer_,  
  publicationtype = NA_character_,  
  year = NA_character_,  
  citation = NA_character_,  
  articletitle = NA_character_,  
  journal = NA_character_,  
  volume = NA_character_,  
  issue = NA_character_,  
  pages = NA_character_,  
  citationnumber = NA_character_,  
  doi = NA_character_,  
  booktitle = NA_character_,
```

```

numvolumes = NA_character_,
edition = NA_character_,
volumetitle = NA_character_,
seriestitle = NA_character_,
seriesvolume = NA_character_,
publisher = NA_character_,
url = NA_character_,
city = NA_character_,
state = NA_character_,
country = NA_character_,
originallanguage = NA_character_,
notes = NA_character_,
author = NULL
)

```

### Arguments

publicationid	ID of publication
publicationtypeid	ID of kind of publication
publicationtype	A text string identifying the publication type within the Neotoma database.
year	The year of publication.
citation	A full text citation for the article.
articletitle	The title of the article.
journal	The journal in which the article was published.
volume	The journal volume.
issue	The journal issue.
pages	The pages of the journal.
citationnumber	How many times has the paper been cited?
doi	A DOI for the record.
booktitle	The title of the book (if the publication is a book)
numvolumes	The number of book volumes (if a series)
edition	The book edition.
volumetitle	The title of the volume (in a published series)
seriestitle	The title of the series.
seriesvolume	The series volume.
publisher	The publisher.
url	Publication URL
city	City of publication.
state	State of publication.
country	Country of publication.

originallanguage	Original language of publication.
notes	Publication notes.
author	name of the author of publication.

**Value**

publication object

---

set_sample	<i>set Sample Information</i>
------------	-------------------------------

---

**Description**

Function to create new samples objects for analysis. The new object will not be uploaded to the database.

**Usage**

```
set_sample(
  x = NA,
  ages = list(),
  igsn = NA_character_,
  datum = data.frame(),
  depth = NA_integer_,
  sampleid = NA_integer_,
  thickness = NA_integer_,
  samplename = NA_character_,
  sampleanalyst = list(),
  analysisunitid = NA_integer_,
  analysisunitname = NA_character_
)
```

**Arguments**

x	Object to be set as a sample
ages	ages
igsn	IGSN character
datum	dataframe of datum
depth	integer representing depth
sampleid	ID for sample
thickness	thickness of core
samplename	sample's name
sampleanalyst	Analyst's contact name
analysisunitid	Which analysis unit it is
analysisunitname	Analysis Unit's name

**Value**

sample object

**Examples**

```
{  
# Set an empty sample  
my_sample <- set_sample()  
}
```

---

set\_server

*Set Neotoma API Source or Server*

---

**Description**

Choose to pull Neotoma data from the main Neotoma server, the development server or from a local instance of the API.

**Usage**

```
set_server(server = "neotoma")
```

**Arguments**

server            One of local (when the API is running locally on port 3005), neotoma or dev.

**Value**

NULL modifies how to talk to the API (local, dev, server)

**Examples**

```
# The user is running the API locally using the node/express API  
# cloned from github: https://github.com/NeotomaDB/api\_nodetest  
set_server(server = "local")  
# The user switches back to the remote api server.  
set_server(server = "neotoma")
```



---

 set\_site

*set Site Information for Fossil Sites*


---

## Description

set Site Information for Fossil Sites

## Usage

```
set_site(
  x = NA,
  siteid = NA_integer_,
  sitename = NA_character_,
  geography = st_as_sf(st_sfc()),
  altitude = NA_integer_,
  geopolitical = list(),
  area = NA_integer_,
  notes = NA_character_,
  description = NA_character_,
  collunits = new("collunits")
)
```

## Arguments

x	Object to be set as a site
siteid	The unique site id for a site. If this site is new to Neotoma then leave the ID as NA (the default).
sitename	Actual site name as a character string.
geography	An sf object representing the site location, either as a polygon or point.
altitude	altitude/elevation of the site.
geopolitical	The geopolitical unit in which the site is located.
area	The area of the site or depositional basin in <i>ha</i> . Can be calculated from the polygon.
notes	additional information of the site
description	Function to create new site objects for personal analysis. The new object will not be uploaded to the database.
collunits	Collection units in the site

## Value

site object

**Examples**

```
{
# Create a site called "My Lake", to
x = sf::st_as_sf(sf::st_sfc(sf::st_point(c(5,5))))
my_site <- set_site(sitename = "My Lake",
                    geography = x,
                    description = "my lake",
                    altitude = 30)
}
```

---

show,collunit-method *Show the collection unit information*

---

**Description**

Show the collection unit information

**Usage**

```
## S4 method for signature 'collunit'
show(object)
```

**Arguments**

object            collunit object

**Value**

null used for side effects. Printing a data.frame

---

show,collunits-method *Show the collection unit information*

---

**Description**

Show the collection unit information

**Usage**

```
## S4 method for signature 'collunits'
show(object)
```

**Arguments**

object            collunits object

**Value**

null used for side effects. Printing a data.frame

---

show,contact-method    *Show contact object*

---

**Description**

Show contact object  
Show a contact object

**Usage**

```
## S4 method for signature 'contact'  
show(object)  
  
## S4 method for signature 'contact'  
show(object)
```

**Arguments**

object            a contact object

**Value**

null - side effect for printing contact object  
Null - prints a data.frame

---

show,contacts-method    *Show a contacts object.*

---

**Description**

Show a contacts object.

**Usage**

```
## S4 method for signature 'contacts'  
show(object)
```

**Arguments**

object            A contacts object.

**Value**

null - side effect for printing contacts object

show,dataset-method    *Show Dataset Method*

---

**Description**

Show Dataset Method

**Usage**

```
## S4 method for signature 'dataset'  
show(object)
```

**Arguments**

object                  dataset object

**Value**

null - side effect, prints a data.frame with dataset metadata

---

show,datasets-method    *Show Datasets object as a dataframe*

---

**Description**

Show Datasets object as a dataframe

**Usage**

```
## S4 method for signature 'datasets'  
show(object)
```

**Arguments**

object                  datasets object

**Value**

null - side effect, prints a data.frame with datasets metadata

---

show,publication-method  
*Print publications to screen.*

---

**Description**

Print publications to screen.

**Usage**

```
## S4 method for signature 'publication'  
show(object)
```

**Arguments**

object            A publication object.

**Value**

NULL - side effect function of printing a data.frame

---

show,publications-method  
*Show the contents of a publication object.*

---

**Description**

Show the contents of a publication object.

**Usage**

```
## S4 method for signature 'publications'  
show(object)
```

**Arguments**

object            A publications object

**Value**

NULL - side effect function of printing a data.frame

---

show,site-method	<i>Show a site object as a dataframe</i>
------------------	--

---

**Description**

Convert a Neotoma package site object into a `data.frame()` returning the `siteid`, `sitename`, `latitude`, `longitude` and `altitude` of the site.

**Usage**

```
## S4 method for signature 'site'  
show(object)
```

**Arguments**

object            site object

**Value**

NULL - side effect for printing a `data.frame` object

---

show,sites-method	<i>Show sites objects as a dataframe</i>
-------------------	--

---

**Description**

Return a set of site objects as a single `data.frame()`.

**Usage**

```
## S4 method for signature 'sites'  
show(object)
```

**Arguments**

object            sites object

**Value**

NULL - side effect for printing a `data.frame` object

---

show,specimen-method *Show Specimen Method*

---

### **Description**

Show Specimen Method

### **Usage**

```
## S4 method for signature 'specimen'  
show(object)
```

### **Arguments**

object           specimen object

### **Value**

NULL - side effect for printing a data.frame object

---

show,specimens-method *Show Specimens object as a dataframe*

---

### **Description**

Show Specimens object as a dataframe

### **Usage**

```
## S4 method for signature 'specimens'  
show(object)
```

### **Arguments**

object           specimens object

### **Value**

NULL - side effect for printing a data.frame object

---

showMatch	<i>Show matches for objects.</i>
-----------	----------------------------------

---

**Description**

Show matches for objects.

**Usage**

```
showMatch(x)
```

**Arguments**

x                    object to show matches for

**Value**

data.frame that marks if a site exists in another sites object

---

showMatch,publication-method	<i>Show matched publication objects.</i>
------------------------------	--

---

**Description**

Show matched publication objects.

**Usage**

```
## S4 method for signature 'publication'  
showMatch(x)
```

**Arguments**

x                    A publication object.

**Value**

NULL printed matches with other publications



---

site-class	<i>An S4 class for site information</i>
------------	---

---

**Description**

The standard object class for sites from the Neotoma Paleoecology Database.

**Value**

object of class site

---

sites-class	<i>An S4 class for multi-site information</i>
-------------	---

---

**Description**

The standard object class for multi-sites from the Neotoma Paleoecology Database. from @ returns object of class sites

---

specimen-class	<i>S4 class for specimen information</i>
----------------	--

---

**Description**

The standard object class for specimen from the Neotoma Paleoecology Database.

**Value**

object of class specimen

specimens

*Obtain specimens from a record or multiple records.*

---

**Description**

Obtain specimens from a record or multiple records.

**Usage**

```
specimens(x)
```

**Arguments**

x                    sites object

**Value**

data.frame with record information regarding specimens

---

specimens,collunit-method

*specimens*

---

**Description**

Obtain specimen elements from a collunit

**Usage**

```
## S4 method for signature 'collunit'  
specimens(x)
```

**Arguments**

x                    collunit object

**Value**

data.frame with specimens summary table

---

*specimens,collunits-method*  
*specimens*

---

**Description**

Obtain specimen elements from collunits

**Usage**

```
## S4 method for signature 'collunits'  
specimens(x)
```

**Arguments**

x collunits object

**Value**

data.frame with specimens summary table

---

*specimens,site-method* *specimens*

---

**Description**

Obtain elements on the specimens level

**Usage**

```
## S4 method for signature 'site'  
specimens(x)
```

**Arguments**

x site object

**Value**

data.frame with specimens summary table

specimens, sites-method

*specimens*

---

**Description**

Information table for Specimens

**Usage**

```
## S4 method for signature 'sites'  
specimens(x)
```

**Arguments**

x                    Use a sites object that has specimens added.

**Value**

data.frame with specimens summary table

**Author(s)**

Socorro Dominguez <s.dominguez@ht-data.com>

**Examples**

```
# To return a specimens table do:  
my_specimens <- get_specimens(7)  
my_tbl <- specimens(my_specimens)
```

---

specimens-class

*S4 class for specimens information*

---

**Description**

The grouped class for specimens from the Neotoma Paleoecology Database.

**Value**

object of class specimens

---

summary,sites-method *Summary of objects within a sites object.*

---

**Description**

This function summarizes a sites object, from site level and returns a data.frame that contains the site ID, sitename, collectionunit ID, count of chronologies, count of datasets and types of datasets within the site.

**Usage**

```
## S4 method for signature 'sites'  
summary(object, ...)
```

**Arguments**

object	sites object
...	additional properties passed to summary

**Value**

data.frame object with site summary information

---

taxa	<i>taxa</i>
------	-------------

---

**Description**

Show the samples table

**Usage**

```
taxa(object)
```

**Arguments**

object	Sites object to extract taxa table from
--------	---

**Value**

data.frame with taxa records

---

taxa,collunit-method *Extract taxonomic data from a set of sites.*

---

**Description**

Extract taxonomic data from a set of sites.

**Usage**

```
## S4 method for signature 'collunit'  
taxa(object)
```

**Arguments**

object            A collunit object.

**Value**

A data.frame reporting the taxa/data objects, units, elements and other features within a set of records.

---

taxa,collunits-method *Extract taxonomic data from a set of sites.*

---

**Description**

Extract taxonomic data from a set of sites.

**Usage**

```
## S4 method for signature 'collunits'  
taxa(object)
```

**Arguments**

object            A collunits object.

**Value**

A data.frame reporting the taxa/data objects, units, elements and other features within a set of records.

---

taxa,site-method      *Extract taxonomic data from a single site.*

---

**Description**

Extract taxonomic data from a single site.

**Usage**

```
## S4 method for signature 'site'  
taxa(object)
```

**Arguments**

object      A site object.

**Value**

A data.frame reporting the taxa/data objects, units, elements and other features within a set of records.

**Examples**

```
somesites <- get_sites(datasestype = "pollen", limit = 3) %>%  
  get_downloads()  
diatomtaxa <- taxa(somesites[[1]])
```

---

taxa,sites-method      *Extract taxonomic data from a set of sites.*

---

**Description**

From a sites object,

**Usage**

```
## S4 method for signature 'sites'  
taxa(object)
```

**Arguments**

object      A sites object.

**Value**

A data.frame reporting the taxa/data objects, units, elements and other features within a set of records.

**Examples**

```
somesites <- get_sites(datasettype = "diatom", limit = 3) %>%  
  get_downloads()  
diatomtaxa <- taxa(somesites)  
common_taxa <- diatomtaxa %>%  
  dplyr::filter(sites == 3)
```

---

toJSON

*toJSON*

---

**Description**

Export toJSON

**Usage**

```
toJSON(x)
```

**Arguments**

x                    Sites object to extract taxa table from

**Value**

JSON translation of sites object to JSON

**Author(s)**

Socorro Dominguez <sedv8808@gmail.com>



---

toJSON,sites-method    *toJSON*

---

### Description

Convert a Neotoma2 sites object into a standardized JSON file for API management.

### Usage

```
## S4 method for signature 'sites'  
toJSON(x = NA)
```

### Arguments

x                    sites R object to be converted

### Value

The function returns a character string in JSON format

### Author(s)

Socorro Dominguez <s.dominguez@ht-data.com>

### Examples

```
{  
# To find all sites that contain the string "Alexander%"  
alex.sites <- get_sites(sitename="Alexander%")  
# Convert the object to json  
toJSON(alex.sites)  
}
```

---

toWide                    *toWide*

---

### Description

Obtain a wide table with information regarding of samples grouped by variable name and depth/age.

**Usage**

```
toWide(
  x,
  variablenames = c(),
  ecologicalgroups = c(),
  elementtypes = c(),
  unit = c(),
  groupby = "age",
  operation = "prop"
)
```

**Arguments**

x	dataframe object with samples
variablenames	Optional vector to filter by specific variable names.
ecologicalgroups	Vector stating the ecological groups to be filtered by, e.g "DIAT", "TRSH"
elementtypes	Label of element type to filter by, e.g. "pollen", "valve"
unit	Label stating which units to filter by, e.g. "NISP"
groupby	Group by 'age' or 'depth'
operation	label or vector of operations to be chosen from: 'prop', 'sum', 'presence'.

**Value**

wide data.frame obtained from long samples data.frame

**Examples**

```
fourcorners <- '{"type": "Polygon",
"coordinates": [[
[-109.36060497194846, 37.69552879956651],
[-107.813845732192, 37.69552879956651],
[-107.813845732192, 36.80303716260222],
[-109.36060497194846, 36.80303716260222],
[-109.36060497194846, 37.69552879956651]
]]}'

# Download all vertebrate localities within a bounding box.
fc_sites <- neotoma2::get_sites(loc = fourcorners[1])
fc_ds <- neotoma2::get_datasets(fc_sites) %>%
neotoma2::filter(datasettype=="vertebrate fauna")

fc_dl <- neotoma2::get_downloads(fc_ds)
fc_dl1 <- fc_dl[[1]]

fc_smp <- samples(fc_dl1)
toWide(fc_smp, ecologicalgroups=c('AVES', 'RODE'),
elementtypes='bone/tooth', unit='present/absent')
```

---

write.csv,chronologies-method  
*write CSV*

---

**Description**

write CSV

**Usage**

```
## S4 method for signature 'chronologies'  
write.csv(x, ...)
```

**Arguments**

x                    chronologies object  
...                  Additional parameters associated with the call.

**Value**

null, called for side effects

---

write.csv,collunits-method  
*write CSV*

---

**Description**

write CSV

**Usage**

```
## S4 method for signature 'collunits'  
write.csv(x, ...)
```

**Arguments**

x                    collunits object  
...                  Additional parameters associated with the call.

**Value**

null side effect for saving a CSV file.

---

```
write.csv,datasets-method  
write CSV
```

---

**Description**

write CSV

**Usage**

```
## S4 method for signature 'datasets'  
write.csv(x, ...)
```

**Arguments**

x	datasets object
...	Additional parameters associated with the call.

**Value**

null -side effect for printing a CSV file

---

```
write.csv,sites-method  
write CSV
```

---

**Description**

write CSV

**Usage**

```
## S4 method for signature 'sites'  
write.csv(x, ...)
```

**Arguments**

x	A sites object
...	Other options to pass to write.csv().

**Value**

NULL side effect from saving a csv file

---

```
write.csv,specimens-method  
write CSV
```

---

**Description**

write CSV

**Usage**

```
## S4 method for signature 'specimens'  
write.csv(x, ...)
```

**Arguments**

x	specimens object
...	Additional parameters associated with the call.

**Value**

NULL - side effect of saving CSV file

---

```
[,collunits,numeric,ANY-method  
Get or remove sites by numeric index
```

---

**Description**

Retrieve sites by numeric index

**Usage**

```
## S4 method for signature 'collunits,numeric,ANY'  
x[i]
```

**Arguments**

x	The collunits object
i	The numeric index

**Value**

null used for side effects. Printing a data.frame

---

```
[,collunits,numeric-method
```

*Get or remove sites by numeric index*

---

**Description**

Retrieve sites by numeric index

**Usage**

```
## S4 method for signature 'collunits,numeric'  
x[i]
```

**Arguments**

x	The collunits object
i	The numeric index

**Value**

null used for side effects. Printing a data.frame

---

```
[,datasets,numeric,ANY-method
```

*Get or remove datasets by numeric index*

---

**Description**

Get or remove datasets by numeric index

**Usage**

```
## S4 method for signature 'datasets,numeric,ANY'  
x[i]
```

**Arguments**

x	The datasets object
i	The numeric index

**Value**

Get or remove datasets by numeric index

---

[,datasets,numeric-method  
*Get or remove datasets by numeric index*

---

### **Description**

Get or remove datasets by numeric index

### **Usage**

```
## S4 method for signature 'datasets,numeric'  
x[i]
```

### **Arguments**

x	The datasets object
i	The numeric index

### **Value**

Get or remove datasets by numeric index

---

[,site,character,ANY-method  
*Get site field by character index*

---

### **Description**

Get site field by character index

### **Usage**

```
## S4 method for signature 'site,character,ANY'  
x[i]
```

### **Arguments**

x	The site object
i	The column indicator

### **Value**

sliced site object

---

[,site,character-method

*Get site field by character index*

---

### Description

Get site field by character index

### Usage

```
## S4 method for signature 'site,character'  
x[i]
```

### Arguments

x	The site object
i	The column indicator

### Value

sliced site object

---

[,site,numeric,ANY-method

*Get site field by numeric index*

---

### Description

Get site field by numeric index

### Usage

```
## S4 method for signature 'site,numeric,ANY'  
x[i]
```

### Arguments

x	The site object
i	The column indicator

### Value

sliced site object



---

[,site,numeric-method *Get site field by numeric index*

---

### **Description**

Get site field by numeric index

### **Usage**

```
## S4 method for signature 'site,numeric'  
x[i]
```

### **Arguments**

x	The site object
i	The column indicator

### **Value**

sliced site object

---

[,sites,numeric,ANY-method  
*Get or remove sites by numeric index*

---

### **Description**

Get or remove sites by numeric index

### **Usage**

```
## S4 method for signature 'sites,numeric,ANY'  
x[i]
```

### **Arguments**

x	The sites object
i	The numeric index

### **Value**

sliced site object

---

```
[,sites,numeric-method
```

*Get or remove sites by numeric index*

---

### Description

Get or remove sites by numeric index

### Usage

```
## S4 method for signature 'sites,numeric'
x[i]
```

### Arguments

x	The sites object
i	The numeric index

### Value

sliced site object

---

```
[,specimens,numeric,ANY-method
```

*Get or remove specimens by numeric index*

---

### Description

Get or remove specimens by numeric index

### Usage

```
## S4 method for signature 'specimens,numeric,ANY'
x[i]
```

### Arguments

x	The specimens object
i	The numeric index

### Value

sliced specimens object

---

[,specimens,numeric-method  
*Get or remove specimens by numeric index*

---

**Description**

Get or remove specimens by numeric index

**Usage**

```
## S4 method for signature 'specimens,numeric'  
x[i]
```

**Arguments**

x	The specimens object
i	The numeric index

**Value**

sliced specimens object

---

[<- ,collunit,character,ANY,ANY-method  
*Assign collunit field by numeric index*

---

**Description**

Assign collunit field by numeric index

**Usage**

```
## S4 replacement method for signature 'collunit,character,ANY,ANY'  
x[i] <- value
```

**Arguments**

x	The collunit object.
i	The column indicator.
value	The value to be used.

**Value**

sliced element

---

```
[<-,collunit,numeric,ANY,ANY-method
```

*Assign collunit field by numeric index*

---

### Description

Assign collunit field by numeric index

### Usage

```
## S4 replacement method for signature 'collunit,numeric,ANY,ANY'  
x[i] <- value
```

### Arguments

x	The collunit object.
i	The column indicator.
value	The value to be used.

### Value

sliced value

---

```
[<-,dataset,character,ANY,ANY-method
```

*Assign dataset field by numeric index*

---

### Description

Assign dataset field by numeric index

### Usage

```
## S4 replacement method for signature 'dataset,character,ANY,ANY'  
x[i] <- value
```

### Arguments

x	The dataset object.
i	The column indicator.
value	The value to be used.

### Value

dataset slot with new assigned character value

---

[<- ,dataset,numeric,ANY,ANY-method  
*Assign dataset field by numeric index*

---

### Description

Assign dataset field by numeric index

### Usage

```
## S4 replacement method for signature 'dataset,numeric,ANY,ANY'  
x[i] <- value
```

### Arguments

x	The dataset object.
i	The column indicator.
value	The value to be used.

### Value

dataset slot with new assigned numeric value

---

[<- ,site,character,ANY,ANY-method  
*Assign site field by numeric index*

---

### Description

Assign site field by numeric index

### Usage

```
## S4 replacement method for signature 'site,character,ANY,ANY'  
x[i] <- value
```

### Arguments

x	The site object.
i	The column indicator.
value	The value to be used.

### Value

site object with reassigned character values

---

*[<- ,site,numeric,ANY,ANY-method*  
*Assign site field by numeric index*

---

**Description**

Assign site field by numeric index

**Usage**

```
## S4 replacement method for signature 'site,numeric,ANY,ANY'  
x[i] <- value
```

**Arguments**

x	The site object.
i	The column indicator.
value	The value to be used.

**Value**

sites object with reassigned numeric values

---

*[<- ,specimen,character,ANY,ANY-method*  
*Assign specimen field by numeric index*

---

**Description**

Assign specimen field by numeric index

**Usage**

```
## S4 replacement method for signature 'specimen,character,ANY,ANY'  
x[i] <- value
```

**Arguments**

x	The specimen object.
i	The column indicator.
value	The value to be used.

**Value**

specimen object with reassigned character values

---

[<- ,specimen,numeric,ANY,ANY-method  
*Assign specimen field by numeric index*

---

### Description

Assign specimen field by numeric index

### Usage

```
## S4 replacement method for signature 'specimen,numeric,ANY,ANY'  
x[i] <- value
```

### Arguments

x	The specimen object.
i	The column indicator.
value	The value to be used.

### Value

sites object with reassigned numeric values

---

[[,chronologies,numeric,ANY-method  
*Extract*

---

### Description

Obtain one of the elements within a chronologies list either by element order or by element name.

### Usage

```
## S4 method for signature 'chronologies,numeric,ANY'  
x[[i]]
```

### Arguments

x	chronologies object
i	iteration in chronologies list

### Value

selected chronology object

[[,chronologies,numeric-method  
*Extract*

---

**Description**

Obtain one of the elements within a chronologies list either by element order or by element name.

**Usage**

```
## S4 method for signature 'chronologies,numeric'  
x[[i]]
```

**Arguments**

x	chronologies object
i	iteration in chronologies list

**Value**

selected chronology object

---

[[,collunits,numeric,ANY-method  
*Slicer*

---

**Description**

Obtain one of the elements within a collunits list

**Usage**

```
## S4 method for signature 'collunits,numeric,ANY'  
x[[i]]
```

**Arguments**

x	collunits object
i	iteration in collunits list

**Value**

sliced collunits object



---

```
[[,collunits,numeric-method
    Slicer
```

---

**Description**

Obtain one of the elements within a collunits list

**Usage**

```
## S4 method for signature 'collunits,numeric'
x[[i]]
```

**Arguments**

x	collunits object
i	iteration in collunits list

**Value**

sliced collunits object

---

```
[[,contacts,numeric,ANY-method
    Extract or Replace Parts of an Object
```

---

**Description**

Extract or Replace Parts of an Object

**Usage**

```
## S4 method for signature 'contacts,numeric,ANY'
x[[i]]
```

**Arguments**

x	A contact object.
i	The numeric index of a contact slot.

**Value**

sliced contacts

---

[[,contacts,numeric-method

*Extract or Replace Parts of an Object*

---

### Description

Extract or Replace Parts of an Object

### Usage

```
## S4 method for signature 'contacts,numeric'
x[[i]]
```

### Arguments

x                    A contact object.  
i                    The numeric index of a contact slot.

### Value

sliced contacts

---

[[,datasets,numeric,ANY-method

*Slicer*

---

### Description

Obtain one of the elements within a datasets list

### Usage

```
## S4 method for signature 'datasets,numeric,ANY'
x[[i]]
```

### Arguments

x                    datasets object  
i                    iteration in datasets list

### Value

sliced dataset object

---

```
[[,datasets,numeric-method
      Slicer
```

---

**Description**

Obtain one of the elements within a datasets list

**Usage**

```
## S4 method for signature 'datasets,numeric'
x[[i]]
```

**Arguments**

x	datasets object
i	iteration in datasets list

**Value**

sliced dataset object

---

```
[[,publications,numeric,ANY-method
      Obtain one of the elements within a publication list.
```

---

**Description**

Obtain one of the elements within a publication list.

**Usage**

```
## S4 method for signature 'publications,numeric,ANY'
x[[i]]
```

**Arguments**

x	A publications object.
i	A numeric index for the requested publication

**Value**

selected publications object from index

---

```
[[,publications,numeric-method
```

*Obtain one of the elements within a publication list.*

---

### Description

Obtain one of the elements within a publication list.

### Usage

```
## S4 method for signature 'publications,numeric'
x[[i]]
```

### Arguments

x	A publications object.
i	A numeric index for the requested publication

### Value

selected publications object from index

---

```
[[,samples,numeric,ANY-method
```

*Slicer*

---

### Description

Obtain one of the elements within a samples list

### Usage

```
## S4 method for signature 'samples,numeric,ANY'
x[[i]]
```

### Arguments

x	samples object
i	iteration in samples list

### Value

samples sliced object

---

```
[[,samples,numeric-method  
      Slicer
```

---

**Description**

Obtain one of the elements within a samples list

**Usage**

```
## S4 method for signature 'samples,numeric'  
x[[i]]
```

**Arguments**

x	samples object
i	iteration in samples list

**Value**

samples sliced object

---

```
[[,sites,numeric,ANY-method  
      Slicer
```

---

**Description**

Obtain one of the elements within a sites list

**Usage**

```
## S4 method for signature 'sites,numeric,ANY'  
x[[i]]
```

**Arguments**

x	sites object
i	iteration in sites list

**Value**

sliced site object

**Examples**

```
{
  some_site <- get_sites(sitename = "Site%", limit=3)
  some_site[[2]]
}
```

---

```
[[,sites,numeric-method
      Slicer
```

---

**Description**

Obtain one of the elements within a sites list

**Usage**

```
## S4 method for signature 'sites,numeric'
x[[i]]
```

**Arguments**

x	sites object
i	iteration in sites list

**Value**

sliced site object

**Examples**

```
{
  some_site <- get_sites(sitename = "Site%", limit=3)
  some_site[[2]]
}
```

---

```
[[,specimens,numeric,ANY-method
      Slicer
```

---

**Description**

Obtain one of the elements within a specimens list

**Usage**

```
## S4 method for signature 'specimens,numeric,ANY'
x[[i]]
```

**Arguments**

x                   specimens object  
i                   iteration in specimens list

**Value**

sliced specimens object

---

[[,specimens,numeric-method  
*Slicer*

---

**Description**

Obtain one of the elements within a specimens list

**Usage**

```
## S4 method for signature 'specimens,numeric'  
x[[i]]
```

**Arguments**

x                   specimens object  
i                   iteration in specimens list

**Value**

sliced specimens object

---

[[<-,collunits,ANY,ANY,ANY-method  
*Insert collunit*

---

**Description**

Obtain one of the elements within a collunits list

**Usage**

```
## S4 replacement method for signature 'collunits,ANY,ANY,ANY'  
x[[i]] <- value
```

**Arguments**

x	collunits object
i	iteration in collunits list
value	The value to be used

**Value**

Modified collunits

---

[[<-,datasets,ANY,ANY,ANY-method  
*Insert dataset*

---

**Description**

Obtain one of the elements within a datasets list

**Usage**

```
## S4 replacement method for signature 'datasets,ANY,ANY,ANY'
x[[i]] <- value
```

**Arguments**

x	datasets object
i	iteration in datasets list
value	The value to be used

**Value**

One dataset slot's value

---

[[<-,publications,ANY,ANY,ANY-method  
*Assign value to an element in a publication list.*

---

**Description**

Assign value to an element in a publication list.

**Usage**

```
## S4 replacement method for signature 'publications,ANY,ANY,ANY'
x[[i]] <- value
```



**Arguments**

x	A publications object.
i	A numeric index for the requested publication
value	The value to be used

**Value**

publications with new assigned value.

---

```
[[<-,sites,ANY,ANY,ANY-method
      Insert site
```

---

**Description**

Obtain one of the elements within a sites list

**Usage**

```
## S4 replacement method for signature 'sites,ANY,ANY,ANY'
x[[i]] <- value
```

**Arguments**

x	sites object
i	iteration in sites list
value	The value to be used

**Value**

sites object with reassigned values

---

```
[[<-,specimens,ANY,ANY,ANY-method
      Insert specimen
```

---

**Description**

Obtain one of the elements within a specimens list

**Usage**

```
## S4 replacement method for signature 'specimens,ANY,ANY,ANY'
x[[i]] <- value
```

**Arguments**

x	specimens object
i	iteration in specimens list
value	The value to be used

**Value**

specimens object with reassigned values

---

\$.chronologies-method *Extract*

---

**Description**

Obtain chronology slots across all chronology elements within a chronologies object.

**Usage**

```
## S4 method for signature 'chronologies'
x$name
```

**Arguments**

x	chronologies object
name	name of the slot

**Value**

A multiple chronologies

---

\$.chronology-method *Extract*

---

**Description**

Extract chronology metadata by slot name.

**Usage**

```
## S4 method for signature 'chronology'
x$name
```

**Arguments**

x	chronology object
name	name of the slot

**Value**

value of the slot name

---

*\$,collunit-method*     *\$*

---

**Description**

Obtain slots of a collunit without using at-mark

**Usage**

```
## S4 method for signature 'collunit'  
x$name
```

**Arguments**

x	collunit object
name	name of the slot

**Value**

null prints element of a slot

---

*\$,collunits-method*     *\$ for collunits*

---

**Description**

Obtain slots of a collunit without using at-mark

**Usage**

```
## S4 method for signature 'collunits'  
x$name
```

**Arguments**

x	collunits object
name	name of the slot

**Value**

null prints element of a slot

\$.contact-method      *Extract or Replace Parts of an Object*

---

**Description**

Extract or Replace Parts of an Object

**Usage**

```
## S4 method for signature 'contact'  
x$name
```

**Arguments**

x	A contact object.
name	The name of a contact slot.

**Value**

value in the selected slot

---

\$.contacts-method      *Extract or Replace Parts of an Object*

---

**Description**

Extract or Replace Parts of an Object

**Usage**

```
## S4 method for signature 'contacts'  
x$name
```

**Arguments**

x	A contacts object.
name	The name of a contacts slot.

**Value**

object value from the slot

---

,\$dataset-method      \$

---

**Description**

Obtain slots of a dataset without using at-mark

**Usage**

```
## S4 method for signature 'dataset'  
x$name
```

**Arguments**

x	dataset object
name	name of the slot

**Value**

Obtain a dataset's slot value using \$

---

,\$datasets-method      \$ *for datasets*

---

**Description**

Obtain slots of a dataset without using at-mark

**Usage**

```
## S4 method for signature 'datasets'  
x$name
```

**Arguments**

x	datasets object
name	name of the slot.

**Value**

Obtain a datasets' slot value using \$

---

\$,publication-method    *Extract an element from a publication*

---

**Description**

Extract an element from a publication

**Usage**

```
## S4 method for signature 'publication'
x$name
```

**Arguments**

x	A publication object.
name	The slot to obtain (e.g., articletitle)

**Value**

value in the selected slot

---

\$,sample-method    \$

---

**Description**

Obtain slots of a sample without using at-mark

**Usage**

```
## S4 method for signature 'sample'
x$name
```

**Arguments**

x	sample object
name	name of the slot

**Value**

value at selected slot

---

`$.samples-method`      *\$.for samples*

---

**Description**

Obtain slots of a site without using at-mark

**Usage**

```
## S4 method for signature 'samples'  
x$name
```

**Arguments**

x	samples object
name	name of the slot

**Value**

value at selected slot

---

`$.site-method`      `$`

---

**Description**

Obtain slots of a site without using at-mark

**Usage**

```
## S4 method for signature 'site'  
x$name
```

**Arguments**

x	site object
name	name of the slot

**Value**

value at chosen slot in the site object

---

\$,sites-method	<i>\$ for sites</i>
-----------------	---------------------

---

**Description**

Obtain slots of a site without using at-mark

**Usage**

```
## S4 method for signature 'sites'
x$name
```

**Arguments**

x	sites object
name	name of the slot

**Value**

value at chosen slot in the site object

---

\$,specimen-method	\$
--------------------	----

---

**Description**

Obtain slots of a specimen without using at-mark

**Usage**

```
## S4 method for signature 'specimen'
x$name
```

**Arguments**

x	specimen object
name	name of the slot

**Value**

value at chosen slot in the specimen object



---

*\$.specimens-method*     *\$ for specimens*

---

**Description**

Obtain slots of a specimen without using at-mark

**Usage**

```
## S4 method for signature 'specimens'  
x$name
```

**Arguments**

x	specimens object
name	name of the slot.

**Value**

value at chosen slot in the site object

---

*\$<- ,chronology-method*     *Replace part of an object*

---

**Description**

Assign values to slots within a chronology object.

**Usage**

```
## S4 replacement method for signature 'chronology'  
x$name <- value
```

**Arguments**

x	A chronology object
name	The name of the chronology slot.
value	A value to be assigned to the chronology slot.

**Value**

reassigned chronology object

---

*\$<- ,collunit-method*     *Assign collunit field by numeric index*

---

**Description**

Assign collunit field by numeric index

**Usage**

```
## S4 replacement method for signature 'collunit'
x$name <- value
```

**Arguments**

x	The collunit object.
name	name of the slot.
value	The value to be used.

**Value**

assign a new value to a slot

---

*\$<- ,dataset-method*     *Assign dataset field by numeric index*

---

**Description**

Assign dataset field by numeric index

**Usage**

```
## S4 replacement method for signature 'dataset'
x$name <- value
```

**Arguments**

x	The dataset object.
name	name of the slot.
value	The value to be used.

**Value**

Assign new dataset by numeric index

---

`$<-`,site-method      *Assign site field by numeric index*

---

**Description**

Assign site field by numeric index

**Usage**

```
## S4 replacement method for signature 'site'  
x$name <- value
```

**Arguments**

x	The site object.
name	name of the slot
value	The value to be used.

**Value**

site object with reassigned values

---

`$<-`,specimen-method      *Assign specimen field by numeric index*

---

**Description**

Assign specimen field by numeric index

**Usage**

```
## S4 replacement method for signature 'specimen'  
x$name <- value
```

**Arguments**

x	The specimen object.
name	name of the slot.
value	The value to be used.

**Value**

specimen object with reassigned values

# Index

- [,collunits,numeric,ANY-method, [117](#)
- [,collunits,numeric-method, [118](#)
- [,datasets,numeric,ANY-method, [118](#)
- [,datasets,numeric-method, [119](#)
- [,site,character,ANY-method, [119](#)
- [,site,character-method, [120](#)
- [,site,numeric,ANY-method, [120](#)
- [,site,numeric-method, [121](#)
- [,sites,numeric,ANY-method, [121](#)
- [,sites,numeric-method, [122](#)
- [,specimens,numeric,ANY-method, [122](#)
- [,specimens,numeric-method, [123](#)
- [<-,collunit,character,ANY,ANY-method, [123](#)
- [<-,collunit,numeric,ANY,ANY-method, [124](#)
- [<-,dataset,character,ANY,ANY-method, [124](#)
- [<-,dataset,numeric,ANY,ANY-method, [125](#)
- [<-,site,character,ANY,ANY-method, [125](#)
- [<-,site,numeric,ANY,ANY-method, [126](#)
- [<-,specimen,character,ANY,ANY-method, [126](#)
- [<-,specimen,numeric,ANY,ANY-method, [127](#)
- [[,chronologies,numeric,ANY-method, [127](#)
- [[,chronologies,numeric-method, [128](#)
- [[,collunits,numeric,ANY-method, [128](#)
- [[,collunits,numeric-method, [129](#)
- [[,contacts,numeric,ANY-method, [129](#)
- [[,contacts,numeric-method, [130](#)
- [[,datasets,numeric,ANY-method, [130](#)
- [[,datasets,numeric-method, [131](#)
- [[,publications,numeric,ANY-method, [131](#)
- [[,publications,numeric-method, [132](#)
- [[,samples,numeric,ANY-method, [132](#)
- [[,samples,numeric-method, [133](#)
- [[,sites,numeric,ANY-method, [133](#)
- [[,sites,numeric-method, [134](#)
- [[,specimens,numeric,ANY-method, [134](#)
- [[,specimens,numeric-method, [135](#)
- [[<-,collunits,ANY,ANY,ANY-method, [135](#)
- [[<-,datasets,ANY,ANY,ANY-method, [136](#)
- [[<-,publications,ANY,ANY,ANY-method, [136](#)
- [[<-,sites,ANY,ANY,ANY-method, [137](#)
- [[<-,specimens,ANY,ANY,ANY-method, [137](#)
- \$.chronologies-method, [138](#)
- \$.chronology-method, [138](#)
- \$.collunit-method, [139](#)
- \$.collunits-method, [139](#)
- \$.contact-method, [140](#)
- \$.contacts-method, [140](#)
- \$.dataset-method, [141](#)
- \$.datasets-method, [141](#)
- \$.publication-method, [142](#)
- \$.sample-method, [142](#)
- \$.samples-method, [143](#)
- \$.site-method, [143](#)
- \$.sites-method, [144](#)
- \$.specimen-method, [144](#)
- \$.specimens-method, [145](#)
- \$<-,chronology-method, [145](#)
- \$<-,collunit-method, [146](#)
- \$<-,dataset-method, [146](#)
- \$<-,site-method, [147](#)
- \$<-,specimen-method, [147](#)
- add\_chronology, [7](#)
- add\_chronology,collunit,chronology,data.frame-method, [7](#)
- as.data.frame,authors-method, [8](#)
- as.data.frame,chronologies-method, [9](#)
- as.data.frame,chronology-method, [9](#)
- as.data.frame,collunit-method, [10](#)
- as.data.frame,collunits-method, [10](#)

- as.data.frame,contact-method, 11
- as.data.frame,contacts-method, 11
- as.data.frame,dataset-method, 12
- as.data.frame,datasets-method, 12
- as.data.frame,publication-method, 13
- as.data.frame,publications-method, 13
- as.data.frame,site-method, 14
- as.data.frame,sites-method, 14
- as.data.frame,specimen-method, 15
- as.data.frame,specimens-method, 15
- as.list,sites-method, 16
- author-class, 16
- authors-class, 17
  
- build\_chron, 17
- build\_collunits, 18
- build\_dataset, 18
- build\_sample, 19
- build\_sites, 19
- build\_specimen, 20
  
- c,chronologies-method, 20
- c,collunits-method, 21
- c,contact-method, 21
- c,contacts-method, 22
- c,datasets-method, 22
- c,missingOrNULL-method, 23
- c,publications-method, 23
- c,sites-method, 24
- c,specimens-method, 24
- check\_args, 25
- check\_contacts, 25
- check\_contacts.contacts, 26
- chroncontrols, 26
- chroncontrols,site-method, 27
- chroncontrols,sites-method, 27
- chronologies, 28
- chronologies,collunit-method, 28
- chronologies,collunits-method, 29
- chronologies,site-method, 29
- chronologies,sites-method, 30
- chronologies-class, 30
- chronology-class, 30
- cite\_data, 31
- cite\_data,site-method, 31
- cite\_data,sites-method, 32
- collunit-class, 32
- collunits, 33
- collunits,site-method, 33
  
- collunits,sites-method, 34
- collunits-class, 34
- contact-class, 34
- contacts-class, 35
- coordinates, 35
- coordinates,sites-method, 36
  
- dataset-class, 36
- datasets, 37
- datasets,collunit-method, 37
- datasets,collunits-method, 38
- datasets,site-method, 38
- datasets,sites-method, 39
- datasets-class, 39
- doi, 39
- doi,publication-method, 40
- doi,site-method, 40
- doi,sites-method, 41
  
- filter, 41
- filter.sites, 43
  
- get\_contacts, 47
- get\_contacts.default, 47
- get\_contacts.numeric, 48
- get\_datasets, 25, 49
- get\_datasets.default, 50
- get\_datasets.numeric, 51
- get\_datasets.site, 52
- get\_datasets.sites, 52
- get\_downloads, 25, 53
- get\_downloads.character, 55
- get\_downloads.numeric, 55
- get\_downloads.sites, 56
- get\_manual, 56
- get\_publications, 57
- get\_publications.default, 58
- get\_publications.numeric, 58
- get\_publications.publication, 59
- get\_publications.publications, 60
- get\_sites, 25, 61
- get\_sites.default, 63
- get\_sites.numeric, 64
- get\_sites.sites, 64
- get\_specimens, 65
- get\_specimens.default, 66
- get\_specimens.numeric, 66
- get\_specimens.sites, 67
- get\_stats, 68

- get\_table, 69
- getids, 44
- getids.collunit, 45
- getids.collunits, 45
- getids.site, 46
- getids.sites, 46
- length, chronologies-method, 69
- length, collunits-method, 70
- length, datasets-method, 70
- length, publications-method, 71
- length, samples-method, 71
- length, sites-method, 72
- length, specimens-method, 72
- missingOrNULL-class, 73
- names, collunit-method, 73
- names, contact-method, 73
- names, dataset-method, 74
- names, publication-method, 74
- names, publications-method, 75
- names, site-method, 75
- names, specimen-method, 76
- newURL, 76
- parse\_site, 77
- parseURL, 77
- pingNeotoma, 78
- plot, sites, ANY-method, 78
- plot, sites-method, 79
- plotLeaflet, 79
- plotLeaflet, site-method, 80
- plotLeaflet, sites-method, 80
- publication-class, 81
- publications-class, 81
- repositories-class, 82
- repository-class, 82
- sample-class, 82
- samples, 83
- samples, collunit-method, 83
- samples, collunits-method, 84
- samples, site-method, 84
- samples, sites-method, 85
- samples-class, 85
- selectMatch, 86
- selectMatch, publication, logical-method, 86
- selectMatch, publication, numeric-method, 87
- set\_chronology, 87
- set\_collunit, 88
- set\_contact, 90
- set\_dataset, 91
- set\_default, 92
- set\_default, chronologies-method, 93
- set\_publications, 93
- set\_sample, 95
- set\_server, 96
- set\_site, 97
- show, collunit-method, 98
- show, collunits-method, 98
- show, contact-method, 99
- show, contacts-method, 99
- show, dataset-method, 100
- show, datasets-method, 100
- show, publication-method, 101
- show, publications-method, 101
- show, site-method, 102
- show, sites-method, 102
- show, specimen-method, 103
- show, specimens-method, 103
- showMatch, 104
- showMatch, publication-method, 104
- site-class, 105
- sites-class, 105
- specimen-class, 105
- specimens, 106
- specimens, collunit-method, 106
- specimens, collunits-method, 107
- specimens, site-method, 107
- specimens, sites-method, 108
- specimens-class, 108
- summary, sites-method, 109
- taxa, 109
- taxa, collunit-method, 110
- taxa, collunits-method, 110
- taxa, site-method, 111
- taxa, sites-method, 111
- toJSON, 112
- toJSON, sites-method, 113
- toWide, 113
- write.csv, chronologies-method, 115
- write.csv, collunits-method, 115
- write.csv, datasets-method, 116

`write.csv,sites-method`, [116](#)  
`write.csv,specimens-method`, [117](#)