

Package ‘CSW’

November 19, 2018

Title Read access to Catalog Service for the Web information

Version 0.1

Description Functions for read access to Catalog Service for the Web information.

License MIT + file LICENSE

URL <https://github.com/Han0ostdijk/CSW>

Encoding UTF-8

LazyData true

Imports purrr (>= 0.2.5),
glue (>= 1.3.0),
xml2 (>= 1.2.0.9000),
tibble (>= 1.4.2),
magrittr (>= 1.5),
dplyr (>= 0.7.7),
tidyr (>= 0.8.2),
fs (>= 1.2.6)

Remotes r-lib/xml2

Roxygen list(markdown = TRUE)

RoxygenNote 6.1.1

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

CSW-package	2
CSW_DescribeRecord	2
CSW_display_node	4
CSW_GetCapabilities	4
CSW_GetData	5
CSW_GetDomain	6
CSW_GetDomainParameterNames	8
CSW_GetHits	9
CSW_GetQueryables	10
CSW_GetRecordById	10
CSW_GetRecords	12

CSW_GetRecordsAll	14
CSW_get_url	15
CSW_get_version	15
CSW_set_url	16
CSW_set_version	16
Index	18

CSW-package	<i>CSW: A package for reading catalog information with the Catalog Service for the Web (CSW) interface.</i>
-------------	---

Description

The [Catalogue Service](#) page of the [Open Geospatial Consortium \(OGC\)](#) describes: Catalogue services support the ability to publish and search collections of descriptive information (metadata) for data, services, and related information objects. Metadata in catalogues represent resource characteristics that can be queried and presented for evaluation and further processing by both humans and software. Catalogue services are required to support the discovery and binding to registered information resources within an information community.

One of the ways to get information about and from a CSW catalog is the GET method of the HTTP protocol. By specifying a properly formed URL in a internet browser the requested information is shown in the browser window.

This package provides functions that use the same GET method to place the result in an R object such as e.g. a data.frame, xml_document, list or integer variable.

The package has functions for the **read-only** operations of the CSW interface: e.g. the CSW_GetRecords function corresponds wiht the GetRecords operation. Also included are some utility functions.

Until now this package is only tested on the [Nationaal GeoRegister\(NGR\)](#) catalog with CSW 2.0.2. Let me know if you encounter problems with other catalogs: maybe these can be solved.

See the vignette for examples.

CSW_DescribeRecord	<i>Get Description of CSW Records</i>
--------------------	---------------------------------------

Description

This function gives a description of the catalog records in the form of an xml document. With typeName `csw:Record` the description follows a table format and that is why I like this format more than the one you get with typeName `gmd:MD_Metadata`. It is possible to get both formats in one xml document as shown in example exp12.

Usage

```
CSW_DescribeRecord(typeName = c("csw:Record", "gmd:MD_Metadata",
  "gmi:MI_Metadata"), typeNames = c("csw:Record", "gmd:MD_Metadata",
  "gmi:MI_Metadata"),
  namespace = c("csw:http://www.opengis.net/cat/csw/2.0.2",
  "gmd:http://www.isotc211.org/2005/gmd",
  "gmi:http://www.isotc211.org/2005/gmi"),
  outputFormat = "application/xml", version = CSW_get_version(),
  baseurl = CSW_get_url(), verbose = c("N", "F", "Y"))
```

Arguments

typeName	Character vector with one or both of <code>csw:Record</code> and <code>gmd:MD_Metadata</code> . But see remarks.
typeNames	alternative for <code>typeName</code> .
namespace	Character vector with one or both of <code>csw:http://www.opengis.net/cat/csw/2.0.2</code> and <code>gmd:http://www.isotc211.org/2005/gmd</code> . See Remarks for the relation between <code>typeName</code> and <code>namespace</code> .
outputFormat	Character vector. Only <code>application/xml</code> is apparently (?) supported by CSW
version	Character vector with CSW version. Default is <code>2.0.2</code> but this can be changed for the remainder of the session with <code>CSW_set_version</code> .
baseurl	Character vector with base url of the CSW server. Default is <code>http://nationaalgeoregister.nl/geonetwork/srv/dut/csw?</code> but this can be changed for the remainder of the session with <code>CSW_set_url</code> .
verbose	Character N, F or Y. When F the full generated and encoded url will be displayed, when Y the variable part of the url (without baseurl, service indication CSW and version) will be displayed in decoded form and when N (default) nothing of the generated url will be displayed.

Value

An xml document with the record description in the indicated layout.

Remarks

`typeName` `gfc:FC_FeatureCatalogue` and `dcat` were also mentioned as possible values but can not be combined with the two `namespace` options that are given here. Also found was the `gmi` analog for `gmd` but for the default catalog no results were delivered. Moreover the `typeName` and `namespace` must match: the first `typeName` with the first `namespace` and the same for the second one.

Examples

```
## Not run:
exp1 = CSW_DescribeRecord()
exp2 = CSW_DescribeRecord(typeName = 'gmd:MD_Metadata',
  namespace = 'gmd:http://www.isotc211.org/2005/gmd')
exp12 = CSW_DescribeRecord(
  typeName =
    c('csw:Record', 'gmd:MD_Metadata') ,
  namespace = c(
```

```
'csw:http://www.opengis.net/cat/csw/2.0.2',
'gmd:http://www.isotc211.org/2005/gmd') )

## End(Not run)
```

CSW_display_node	<i>Display an xml document or node</i>
------------------	--

Description

This function displays the contents of an xml document or node in the console log.

Usage

```
CSW_display_node(n1, file = NULL)
```

Arguments

n1	An object of type <code>xml_document</code> or <code>xml_node</code>
file	Path to file or connection to write to. Default NULL i.e. write to console log

Examples

```
## Not run:
CSW_display_node(xml1)

## End(Not run)
```

CSW_GetCapabilities	<i>Get Capabilities of the CSW server</i>
---------------------	---

Description

The `CSW_GetCapabilities` function gives information about the capabilities that are available to query the catalog. By calling this function without arguments we receive an `xml` document with this information. We can view the contents of this document (e.g. with the utility function `CSW_display_node`) and study its structure. We see among other things the operations that are available and the parameters with which these operations can be called. Apart of viewing the document we can also select parts of it with functions of the **xml2** package by using XPATH expressions. The vignette shows how one can see which operations are provided by the CSW interface and which parameters are available for the `GetRecords` operation (i.e. the `CSW_GetRecords` function).

Usage

```
CSW_GetCapabilities(sections = c("OperationsMetadata",
  "ServiceIdentification", "ServiceProvider", "Filter_Capabilities"),
  typeNames = c("csw:Record", "gmd:MD_Metadata",
  "gfc:FC_FeatureCatalogue", "dcat"), version = CSW_get_version(),
  baseUrl = CSW_get_url(), verbose = c("N", "F", "Y"))
```

Arguments

<code>sections</code>	Character vector to restrict the output to one of the <code>OperationsMetadata</code> , <code>ServiceIdentification</code> , <code>ServiceProvider</code> and <code>Filter_Capabilities</code> sections. When the argument is not specified all information is provided. NB whatever section is requested, the <code>Filter_Capabilities</code> section is always included.
<code>typeNames</code>	Character vector with one of <code>csw:Record</code> , <code>gmd:MD_Metadata</code> , <code>gfc:FC_FeatureCatalogue</code> and <code>dcat</code> . Default <code>csw:Record</code> .
<code>version</code>	Character vector with CSW version. Default is 2.0.2 but this can be changed for the remainder of the session with <code>CSW_set_version</code> .
<code>baseurl</code>	Character vector with base url of the CSW server. Default is http://nationaalgeoregister.nl/geonetwork/srv/dut/csw? but this can be changed for the remainder of the session with <code>CSW_set_url</code> .
<code>verbose</code>	Character N, F or Y. When F the full generated and encoded url will be displayed, when Y the variable part of the url (without baseurl, service indication CSW and version) will be displayed in decoded form and when N (default) nothing of the generated url will be displayed.

Value

An xml document with the indicated server capabilities.

Examples

```
## Not run:
exp1 = CSW_GetCapabilities(sections = 'OperationsMetadata',
                          typeNames = 'csw:Record')
# CSW_display_node(exp1)
exp2 = CSW_GetCapabilities(sections = 'OperationsMetadata',
                          typeNames = 'gmd:MD_Metadata')

## End(Not run)
```

CSW_GetData

Retrieve data from CSW server

Description

This function is the only function of the CSW package that actually retrieves data from the CSW Server. Most often this function will not be used directly by the user. For the various CSW operations dedicated functions are provided that generate a string with parameters that is passed to `CSW_GetData`. With the parameter `verbose` one can indicate that this generated string is printed in the console log. These parameters can then specified for this function (e.g. when an option would be needed that is not provided for by the dedicated functions). See example.

Usage

```
CSW_GetData(request = c("GetCapabilities", "GetRecords",
                        "DescribeRecord", "GetDomain", "GetRecordById", "Transaction",
                        "Harvest"), version = CSW_get_version(), baseurl = CSW_get_url(),
            verbose = c("N", "F", "Y"), ...)
```

Arguments

request	Character vector indicating the CSW operation that has to be performed. One of <code>GetCapabilities</code> , <code>GetRecords</code> , <code>DescribeRecord</code> , <code>GetDomain</code> , <code>GetRecordById</code> , <code>Transaction</code> , <code>Harvest</code> . Default <code>GetCapabilities</code> .
version	Character vector with CSW version. Default is 2.0.2 but this can be changed for the remainder of the session with <code>CSW_set_version</code> .
baseurl	Character vector with base url of the CSW server. Default is <code>http://nationaalgeoregister.nl/geonetwork/srv/dut/csw?</code> but this can be changed for the remainder of the session with <code>CSW_set_url</code> .
verbose	Character N, F or Y. When F the full generated and encoded url will be displayed, when Y the variable part of the url (without baseurl, service indication CSW and version) will be displayed in decoded form and when N (default) nothing of the generated url will be displayed.
...	Optionally one or more name=value arguments. See example exp2.

Value

An xml document with the requested information.

Remarks

CSW_GetData always returns an xml document where other functions sometimes try to re-format the information when requested. Using this function can help with debugging when there is a problem (e.g. with this formatting).

Examples

```
## Not run:
# exp1 and exp2 request the same information from the server:
exp1 = CSW_DescribeRecord(typeName = 'gmd:MD_Metadata',
  namespace = 'gmd:http://www.isotc211.org/2005/gmd', verbose='N')
exp2 = CSW_GetData(request='DescribeRecord',
  namespace = 'gmd:http://www.isotc211.org/2005/gmd',
  typeName = 'gmd:MD_Metadata', verbose='N')

## End(Not run)
```

CSW_GetDomain

Get Domain values of the CSW server

Description

With the `GetDomain` operation the CSW server can return information about `ParameterNames` or `PropertyNames` (fields).

We can inquire after the parameters of an operation by specifying the `ParameterName` argument. This argument consists of one or more `operation.parameter` pairs as in example exp1. The output is a list (or alternatively an `xml_document`) with the values that can be used for the parameters. In each pair the part before the point should be an operation; the part after the point a parameter. `CSW_GetDomainParameterNames` gives the same information for all possible combinations (at the cost of some extra run-time).

We can also use the `CSW_GetDomain` function to inquire which values a certain field can take in a catalog. We do this by specifying the fieldname in the `PropertyName` argument. This argument consists of one or more fieldnames as e.g. in example `exp2`. The output is (just as in the `ParameterName` case) a list (or alternatively an `xml_document`) with the values that can be taken by the field. [CSW_GetQueryables](#) will show which fieldnames (`PropertyNames`) are recognized.

Usage

```
CSW_GetDomain(ParameterName = "", PropertyName = "",
  output = c("list", "xml"), version = CSW_get_version(),
  baseurl = CSW_get_url(), verbose = c("N", "F", "Y"))
```

Arguments

<code>ParameterName</code>	Character vector with one or more CSW parameters. See CSW_GetDomainParameterNames for possible values. When <code>ParameterName</code> and <code>PropertyName</code> are both used in the same call <code>ParameterName</code> will be ignored.
<code>PropertyName</code>	Character vector with one or more CSW parameters. See CSW_GetQueryables for possible values. When <code>ParameterName</code> and <code>PropertyName</code> are both used in the same call <code>ParameterName</code> will be ignored.
<code>output</code>	Character vector with one of <code>list</code> and <code>xml</code> . With the default <code>list</code> the <code>xml</code> output is converted to a list.
<code>version</code>	Character vector with CSW version. Default is <code>2.0.2</code> but this can be changed for the remainder of the session with CSW_set_version .
<code>baseurl</code>	Character vector with base url of the CSW server. Default is http://nationaalgeoregister.nl/geonetwork/srv/dut/csw? but this can be changed for the remainder of the session with CSW_set_url .
<code>verbose</code>	Character <code>N</code> , <code>F</code> or <code>Y</code> . When <code>F</code> the full generated and encoded url will be displayed, when <code>Y</code> the variable part of the url (without <code>baseurl</code> , service indication <code>CSW</code> and version) will be displayed in decoded form and when <code>N</code> (default) nothing of the generated url will be displayed.

Value

An `xml_document` or list with the values permitted (for `ParameterName`) or to be used (for `PropertyName`)

See Also

[CSW_GetDomainParameterNames](#), [CSW_GetQueryables](#)

Examples

```
## Not run:
exp1 = CSW_GetDomain(
  ParameterName = "GetRecords.outputFormat,GetRecords.ElementSetName",
  verbose = "Y")
exp2 = CSW_GetDomain(
  PropertyName = "Language,GeographicDescriptionCode",
  verbose = "Y")

## End(Not run)
```

CSW_GetDomainParameterNames

Get elements that can be used in GetDomain as ParameterName

Description

CSW_GetDomainParameterNames calls CSW_GetDomain for all (im)possible combinations of Operation and Parameter names assembles the result in a `data.frame`. Not all parameters are found in this way: e.g. `DescribeRecord.outputSchema` can be found in the output of the `GetCapabilities` output but not in the output of `CSW_GetDomainParameterNames` !?

Usage

```
CSW_GetDomainParameterNames(version = CSW_get_version(),
  baseurl = CSW_get_url(), verbose = c("N", "F", "Y"))
```

Arguments

version	Character vector with CSW version. Default is 2.0.2 but this can be changed for the remainder of the session with CSW_set_version .
baseurl	Character vector with base url of the CSW server. Default is http://nationaalgeoregister.nl/geonetwork/srv/dut/csw? but this can be changed for the remainder of the session with CSW_set_url .
verbose	Character N, F or Y. When F the full generated and encoded url will be displayed, when Y the variable part of the url (without baseurl, service indication CSW and version) will be displayed in decoded form and when N (default) nothing of the generated url will be displayed.

Value

A `data.frame` with columns `n` (element names) and `v` (possible parameter values). An element name has the format `CSW_operation.CSW_parameter`.

Remarks

I have seen examples where `CSW_GetCapabilities` directly shows which combinations are supported by `CSW_GetDomain` (see https://docs.oracle.com/cd/E11882_01/appdev.112/e11830/sdo_csw.htm#SPATL963) but for the default baseurl this is apparently not available.

Examples

```
## Not run:
exp1 = CSW_GetDomainParameterNames()

## End(Not run)
```

CSW_GetHits

Get the number of records that match a constraint

Description

CSW_GetHits returns the number of records that satisfy a constraint. The default constraint language and its version are CQL_TEXT and 1.1.0 but can be overwritten by arguments of this function.

Usage

```
CSW_GetHits(constraint = "", constraintLanguage = "CQL_TEXT",
  constraint_language_version = "1.1.0", version = CSW_get_version(),
  baseurl = CSW_get_url(), verbose = c("N", "F", "Y"))
```

Arguments

constraint	Character vector with a constraint written with “ and <code>constraint_language_version==1.1.0</code> .
constraintLanguage	Character vector indicating in which language the constraint is written. Default CQL_TEXT.
constraint_language_version	Character vector indicating the version of the constraint language. Default 1.1.0.
version	Character vector with CSW version. Default is 2.0.2 but this can be changed for the remainder of the session with CSW_set_version .
baseurl	Character vector with base url of the CSW server. Default is http://nationaalgeoregister.nl/geonetwork/srv/dut/csw? but this can be changed for the remainder of the session with CSW_set_url .
verbose	Character N, F or Y. When F the full generated and encoded url will be displayed, when Y the variable part of the url (without baseurl, service indication CSW and version) will be displayed in decoded form and when N (default) nothing of the generated url will be displayed.

Value

An integer with the number of records that satisfy the constraint or an xml_ document with the exception in case of an error

Examples

```
## Not run:
# total number of records in collection :
exp1 = CSW_GetHits()
# total number of records related to 'duin*' (dune*)
exp2 = CSW_GetHits(constraint="AnyText LIKE 'duin%'" )

## End(Not run)
```

CSW_GetQueryables	<i>Get Queryable values of the CSW server</i>
-------------------	---

Description

The function `CSW_GetQueryables` retrieves the names of properties (fields) that can be used in the `CSW_GetDomain` function or in a query. The output is a list with two sublists: one with the 'SupportedISOQueryables' and one with the 'AdditionalQueryables'.

Usage

```
CSW_GetQueryables(version = CSW_get_version(), baseurl = CSW_get_url(),
  verbose = c("N", "F", "Y"))
```

Arguments

<code>version</code>	Character vector with CSW version. Default is 2.0.2 but this can be changed for the remainder of the session with CSW_set_version .
<code>baseurl</code>	Character vector with base url of the CSW server. Default is http://nationaalgeoregister.nl/geonetwork/srv/dut/csw? but this can be changed for the remainder of the session with CSW_set_url .
<code>verbose</code>	Character N, F or Y. When F the full generated and encoded url will be displayed, when Y the variable part of the url (without baseurl, service indication CSW and version) will be displayed in decoded form and when N (default) nothing of the generated url will be displayed.

Value

A list with two sublists: one with the 'SupportedISOQueryables' and one with the 'AdditionalQueryables'

Examples

```
## Not run:
exp1 = CSW_GetQueryables()

## End(Not run)
```

CSW_GetRecordById	<i>Get catalog info for entry with given id</i>
-------------------	---

Description

The only difference with [CSW_GetRecords](#) is that the `constraint` and a 'resultsarguments' of [CSW_GetRecords](#) are argument of [CSW_GetRecordById](#). Therefore see [CSW_GetRecords](#) for details.

Usage

```
CSW_GetRecordById(id = "", typeNames = c("csw:Record",
    "gmd:MD_Metadata", "gfc:FC_FeatureCatalogue", "dcat"),
    outputSchema = c("http://www.opengis.net/cat/csw/2.0.2",
    "http://www.isotc211.org/2005/gmd", "http://www.isotc211.org/2005/gfc",
    "http://www.w3.org/ns/dcat#"), ElementSetName = c("summary", "full",
    "brief"), namespace = c("csw:http://www.opengis.net/cat/csw/2.0.2",
    "gmd:http://www.isotc211.org/2005/gmd"), output = c("table", "list",
    "xml"), version = CSW_get_version(), baseurl = CSW_get_url(),
    verbose = c("N", "F", "Y"))
```

Arguments

id	Character vector with the Identifier of the catalog entry that has to be retrieved.
typeNames	Character vector with one (or both?) of <code>csw:Record</code> and <code>gmd:MD_Metadata</code> . Also <code>gfc:FC_FeatureCatalogue</code> and <code>dcat</code> can be used.
outputSchema	Character vector with one (or more?) of <code>http://www.opengis.net/cat/csw/2.0.2</code> , <code>http://www.isotc211.org/2005/gmd</code> , <code>http://www.isotc211.org/2005/gfc</code> and <code>http://www.w3.org/ns/dcat#</code> . The first two work and the first of them give a more table like presentation.
ElementSetName	Character vector with one of <code>summary</code> , <code>full</code> and <code>brief</code> indicating how much data is presented. Default <code>summary</code> .
namespace	Character vector with one or both of <code>csw:http://www.opengis.net/cat/csw/2.0.2</code> and <code>gmd:http://www.isotc211.org/2005/gmd</code> . See Remarks for the relation between <code>typeName</code> and <code>namespace</code> .
output	Character vector with one of <code>table</code> , <code>list</code> and <code>xml</code> . With <code>table</code> (the default) the results will be put in a (one row) data.frame. With <code>list</code> the output is conforming to that of CSW_GetRecords when <code>list</code> is requested (the number of retrieved (here 1) and total records (here 1) are gathered in a list). With <code>xml</code> the retrieved <code>xml</code> document is passed through without conversion.
version	Character vector with CSW version. Default is <code>2.0.2</code> but this can be changed for the remainder of the session with CSW_set_version .
baseurl	Character vector with base url of the CSW server. Default is http://nationaalgeoregister.nl/geonetwork/srv/dut/csw? but this can be changed for the remainder of the session with CSW_set_url .
verbose	Character N, F or Y. When F the full generated and encoded url will be displayed, when Y the variable part of the url (without baseurl, service indication CSW and version) will be displayed in decoded form and when N (default) nothing of the generated url will be displayed.

Value

An `xml` document with the record description in the indicated layout.

Remarks

We have tried various combinations of `typeNames` (first two only), `outputSchema` (same) and `namespace`. These have only two possible outcomes and depend only on `outputSchema`: the `csw` output is more table like and the `gmd` output is more verbose (?)

See Also[CSW_GetRecords](#)**Examples**

```
## Not run:
exp1 = CSW_GetRecordById(
  id = "ecb96a41-19da-4a57-b570-05a1ee5743ca",
  ElementSetName = "brief",
  output = "xml")

## End(Not run)
```

CSW_GetRecords

*Get catalog info for records satisfying constraint***Description**

The `CSW_GetRecords` function retrieves catalog entries from the CSW server (unless `resultType='validate'` is specified). The default output format is a `data.frame` with as alternatives an `xml_document` or `list` (see `output`). The fields of the `data.frame` are determined by the `ElementSetName` argument that results in few (`brief`), more (`summary`) and most (`full`) fields. The relation between `typeNames`, `outputSchema` and `namespace` is not clear to me, but it seems that the argument values must correspond: taking the first option for `typeNames` one should also take the first one for `outputSchema` and `namespace`. Also see remarks.

The default constraint language and its version are `CQL_TEXT` and `1.1.0` but can be overwritten by arguments of this function. See the vignette for examples and references for the query language.

Usage

```
CSW_GetRecords(constraint = "", constraintLanguage = "CQL_TEXT",
  constraint_language_version = "1.1.0", resultType = c("results",
    "validate"), typeNames = c("csw:Record", "gmd:MD_Metadata",
    "gfc:FC_FeatureCatalogue", "dcat"),
  outputSchema = c("http://www.opengis.net/cat/csw/2.0.2",
    "http://www.isotc211.org/2005/gmd", "http://www.isotc211.org/2005/gfc",
    "http://www.w3.org/ns/dcat#"), ElementSetName = c("summary", "full",
    "brief"), namespace = c("csw:http://www.opengis.net/cat/csw/2.0.2",
    "gmd:http://www.isotc211.org/2005/gmd"), maxRecords = "",
  startPosition = "", output = c("table", "list", "xml"),
  version = CSW_get_version(), baseurl = CSW_get_url(),
  verbose = c("N", "F", "Y"))
```

Arguments

<code>constraint</code>	Character vector with a constraint written with <code>constraintLanguage=CQL_TEXT</code> and <code>constraint_language_version==1.1.0</code> .
<code>constraintLanguage</code>	Character vector indicating in which language the constraint is written. Default <code>CQL_TEXT</code> .

constraint_language_version	Character vector indicating the version of the constraint language. Default 1.1.0.
resultType	Character vector with one of results and validate . With results actual search results will be returned. With validate a Boolean will be returned indicating if the constraint is valid.
typeNames	Character vector with one (or both?) of csw:Record and gmd:MD_Metadata . Also gfc:FC_FeatureCatalogue and dcat can be used.
outputSchema	Character vector with one (or more?) of http://www.opengis.net/cat/csw/2.0.2 , http://www.isotc211.org/2005/gmd , http://www.isotc211.org/2005/gfc and http://www.w3.org/ns/dcat# . The first two work and the first of them give a more table like presentation.
ElementSetName	Character vector with one of summary , full and brief indicating how much data is presented. Default summary .
namespace	Character vector with one or both of csw: http://www.opengis.net/cat/csw/2.0.2 and gmd: http://www.isotc211.org/2005/gmd . See Remarks for the relation between typeName and namespace .
maxRecords	Integer indicating the number of records that will be read from the server. Default is determined by the CSW server (for the default baseurl the default maxRecords is 10 but values in the order of 100 are accepted)
startPosition	Integer indicating the number of the first record that will be returned. The default is 1. To read e.g. the records of a constraint with 150 hits one can issue two requests: the first one with maxRecords = 100 and startPosition = 1 (returning 100 records) and the second with maxRecords = 100 and startPosition = 101 (that will return the last 50 records)
output	Character vector with one of table , list and xml . With table (the default) the results will be put in a data.frame. With list this table is placed in a list with the number of retrieved and total number of records satisfying the constraint. With xml the retrieved xml_document is passed through without conversion.
version	Character vector with CSW version. Default is 2.0.2 but this can be changed for the remainder of the session with CSW_set_version .
baseurl	Character vector with base url of the CSW server. Default is http://nationaalgeoregister.nl/geonetwork/srv/dut/csw? but this can be changed for the remainder of the session with CSW_set_url .
verbose	Character N, F or Y. When F the full generated and encoded url will be displayed, when Y the variable part of the url (without baseurl, service indication CSW and version) will be displayed in decoded form and when N (default) nothing of the generated url will be displayed.

Value

An xml document with the record description in the indicated layout.

Remarks

We have tried various combinations of **typeNames** (first two only), **outputSchema** (same) and **namespace**. These have only two possible outcomes and depend only on **outputSchema**: the csw output is more table like and the gmd output is more verbose (?)

Examples

```
## Not run:
exp1 = CSW_GetRecords(constraint="AnyText LIKE 'duin%'" )

## End(Not run)
```

CSW_GetRecordsAll	<i>Get catalog info for all records satisfying constraint</i>
-------------------	---

Description

CSW_GetRecordsAll retrieves all records that satisfy a constraint by looping over CSW_GetRecords until all records are handled

Usage

```
CSW_GetRecordsAll(constraint, typeNames = c("csw:Record",
      "gfc:FC_FeatureCatalogue", "dcat", "gmd:MD_Metadata"),
  ElementSetName = c("summary", "full", "brief"), output = c("table",
    "list"), batchsize = 100, version = "2.0.2", verbose = F,
  baseurl = CSW_get_url())
```

Arguments

constraint	Character vector with a constraint written with <code>constraintLanguage=CQL_TEXT</code> and <code>constraint_language_version==1.1.0</code> .
typeNames	Character vector with one (or both?) of <code>csw:Record</code> and <code>gmd:MD_Metadata</code> . Also <code>gfc:FC_FeatureCatalogue</code> and <code>dcat</code> can be used.
ElementSetName	Character vector with one of <code>summary</code> , <code>full</code> and <code>brief</code> indicating how much data is presented. Default <code>summary</code> .
output	Character vector with one of <code>table</code> and <code>list</code> . With <code>table</code> (the default) the results will be put in a data.frame. With <code>list</code> this table is placed in a list with the number of retrieved and total number of records satisfying the constraint.
batchsize	Integer with the number of records that will be retrieved in each iteration of the retrieval loop. Default 100.
version	Character vector with CSW version. Default is 2.0.2 but this can be changed for the remainder of the session with CSW_set_version .
verbose	Character N, F or Y. When F the full generated and encoded url will be displayed, when Y the variable part of the url (without baseurl, service indication CSW and version) will be displayed in decoded form and when N (default) nothing of the generated url will be displayed.
baseurl	Character vector with base url of the CSW server. Default is http://nationaalgeoregister.nl/geonetwork/srv/dut/csw? but this can be changed for the remainder of the session with CSW_set_url .

Value

An xml document with the record description in the indicated layout.

See Also[CSW_GetRecords](#)**Examples**

```
## Not run:
exp1 = CSW_GetRecordsAll(constraint="AnyText LIKE 'duin%'" )

## End(Not run)
```

CSW_get_url	<i>Gets the url for the catalog structure</i>
-------------	---

Description

Determines from which structure catalog information will be extracted. If the function CSW_set_url is not called yet this will be done first with the default <http://nationaalgeoregister.nl/geonetwork/srv/dut/csw?> (the url for the PDOK/NGR structures).#'

Usage

```
CSW_get_url()
```

Value

Character vector with the active url.

See Also[CSW_set_url](#)**Examples**

```
CSW_get_url()
```

CSW_get_version	<i>Gets the version for the catalog structure</i>
-----------------	---

Description

Determines the version of the structure catalog information that will be extracted. If the function CSW_set_version is not called yet this will be done first with the default 2.0.2 (the version for the PDOK/NGR structures).

Usage

```
CSW_get_version()
```

Value

Character vector with the active version.

See Also[CSW_set_version](#)**Examples**

```
CSW_get_version()
```

CSW_set_url	<i>Sets the url for CSW catalog structure</i>
-------------	---

Description

Determines from which structure catalog information will be extracted. If this function is not called the default <http://nationaalgeoregister.nl/geonetwork/srv/dut/csw?> (the url for the PDOK/NGR structures) will be used

Usage

```
CSW_set_url(url = NULL)
```

Arguments

url NULL for the default or the url of the structure otherwise

Value

Character vector with (invisible) the url.

See Also[CSW_get_url](#)**Examples**

```
CSW_set_url("http://nationaalgeoregister.nl/geonetwork/srv/dut/inspire?")
```

CSW_set_version	<i>Sets the version for CSW catalog structure</i>
-----------------	---

Description

Determines the version of the structure catalog information that will be extracted. If this function is not called the default 2.0.2 (the version for the PDOK/NGR structures) will be used

Usage

```
CSW_set_version(version = NULL)
```

Arguments

version NULL for the default or the version of the structure otherwise

Value

Character vector with (invisible) the active version.

See Also

[CSW_get_version](#)

Examples

```
CSW_set_version()  
CSW_set_version('2.0.2')
```

Index

CSW-package, [2](#)
CSW_DescribeRecord, [2](#)
CSW_display_node, [4](#)
CSW_get_url, [15](#), [16](#)
CSW_get_version, [15](#), [17](#)
CSW_GetCapabilities, [4](#)
CSW_GetData, [5](#)
CSW_GetDomain, [6](#)
CSW_GetDomainParameterNames, [6](#), [7](#), [8](#)
CSW_GetHits, [9](#)
CSW_GetQueryables, [7](#), [10](#)
CSW_GetRecordById, [10](#), [10](#)
CSW_GetRecords, [10–12](#), [12](#), [15](#)
CSW_GetRecordsAll, [14](#)
CSW_set_url, [3](#), [5–11](#), [13–15](#), [16](#)
CSW_set_version, [3](#), [5–11](#), [13](#), [14](#), [16](#), [16](#)