

Package: xmlrpc2 (via r-universe)

August 25, 2024

Type Package

Title Implementation of the Remote Procedure Call Protocol ('XML-RPC')

Version 1.1

Author Florian Schwendinger [aut, cre]

Maintainer Florian Schwendinger <FlorianSchwendinger@gmx.at>

Description The 'XML-RPC' is a remote procedure call protocol based on 'XML'. The 'xmlrpc2' package is inspired by the 'XMLRPC' package but uses the 'curl' and 'xml2' packages instead 'RCurl' and 'XML'.

License GPL-3

Imports curl, xml2, base64enc

RoxygenNote 6.1.0

NeedsCompilation no

Repository CRAN

Date/Publication 2018-08-23 13:30:02 UTC

Contents

from_xmlrpc	2
rpc_serialize	2
to_xmlrpc	3
xmlrpc	4

Index	5
--------------	----------

from_xmlrpc	<i>Convert from the XML-RPC Format into an R Object.</i>
-------------	--

Description

Convert an object of class "xml_code" or a character in the XML-RPC Format into an R Object.

Usage

```
from_xmlrpc(xml, raise_error = TRUE)
```

Arguments

xml	a character string containing XML in the remote procedure call protocol format.
raise_error	a logical controlling the behavior if the XML-RPC signals a fault. If TRUE an error is raised, if FALSE an object inheriting from "c("xmlrpc_error", "error")" is returned.

Value

an R object derived from the input.

Examples

```
params <- list(1L, 1:3, rnorm(3), LETTERS[1:3], charToRaw("A"))
xml <- to_xmlrpc("some_method", params)
from_xmlrpc(xml)
```

rpc_serialize	<i>Convert R Objects into the XML-RPC Format</i>
---------------	--

Description

Serialize R Objects so they can be passed to to_xmlrpc as parameters.

Usage

```
rpc_serialize(x, ...)
```

Arguments

x	an R object.
...	additional optional arguments (currently ignored).

Value

an object of class "xml_node".

Examples

```
rpc_serialize(1L)
rpc_serialize(1:2)
rpc_serialize(LETTERS[1:2])
```

to_xmlrpc

Create a XML-RPC Call

Description

abc

Usage

```
to_xmlrpc(method, params)
```

Arguments

method	a character string giving the name of the method to be invoked.
params	a list containing the parameters which are added to the XML file sent via the remote procedure call.

Value

an object of class "xml_node" containing a XML-RPC call.

Examples

```
params <- list(1L, 1:3, rnorm(3), LETTERS[1:3], charToRaw("A"))
cat(as.character(to_xmlrpc("some_method", params)))
```

xmlrpc

*Call the Remote Procedure***Description**

Call a remote procedure with the XML-RPC protocol.

Usage

```
xmlrpc(url, method, params = list(), handle = NULL, opts = list(),
       convert = TRUE, useragent = "xmlrpc", raise_error = TRUE)
```

Arguments

url	a character string giving the url to the server.
method	a character string giving the name of the method to be invoked.
params	a list containing the parameters which are added to the XML file sent via the remote procedure call.
handle	a object of class "curl_handle".
opts	a list of options passed to the function "handle_setopt".
convert	a logical, if convert is TRUE (default) the curl response is converted else it is left unchanged.
useragent	a character string giving the name of the "User-Agent".
raise_error	a logical controlling the behavior if the status code of curl_fetch_memory signals an error. If raise_error is TRUE an error is raised, if raise_error is FALSE no error is raised and an object inheriting from c("fetch_error", error)} is returned. This object is the return value from <code>curl_fetch_memory</code> where just the class <code>c("fetch_error", error)</code> is added.

Value

the response of curl or the response converted to R objects.

Examples

```
## Not run:
url <- "https://www.neos-server.org"
xmlrpc(url, "listAllSolvers")
xmlrpc(url, "listSolversInCategory", params = list(category = "socp"))

## End(Not run)
```

Index

`from_xmlrpc`, 2

`rpc_serialize`, 2

`to_xmlrpc`, 3

`xmlrpc`, 4