⇐ XML-RPC Api

Примеры клиентов, использующих API XML-RPC

Beside the simple clients listed below, there is also available:

- DokuWikiConnector .NET API

Sample curl client

This is a very simple example. It's meant to quickly test if everything is correctly setup on the DokuWiki server. This code snippet works from Windows 10 and unixlike systems (Linux, macos, bsd...). This is to be seen as some kind of DW XMLRPC ping.

To run this code :

- from windows, open the windows menu, type cmd, then open the cmd application. NB, do **NOT** use powershell.
- from macos, look for «terminal» in the macos search tool, then open the terminal application
- from linux, look for a terminal in the list of applications, then open it

Once you have a terminal app running, type the following, end the line with the return key. Take care to adapt the URL to your own xmlrpc.php.

```
curl -k -v -H "Content-Type: application/xml" --data-binary "<?xml
version="1.0"?><methodCall><methodName>dokuwiki.getVersion</methodName></met
hodCall>" https://your.wiki.org/lib/exe/xmlrpc.php
```

In return, you'll get the connection information and the DokuWiki version formatted as XML. In the example below, the answer is «Release 2018-04-22c «Greebo»».

```
*   Trying 192.168.1.2...
* TCP_NODELAY set
* Connected to your.wiki.org (192.168.1.2) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* successfully set certificate verify locations:
*   CAfile: /etc/ssl/cert.pem
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Change cipher spec (1):
```

```
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Change cipher spec (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES128-GCM-SHA256
* ALPN, server accepted to use http/1.1
* Server certificate:
*  subject: CN=your.wiki.org
*  start date: Jul  2 18:17:15 2021 GMT
*  expire date: Sep 30 18:17:14 2021 GMT
*  issuer: C=US; O=Let's Encrypt; CN=R3
*  SSL certificate verify ok.
> POST /lib/exe/xmlrpc.php HTTP/1.1
> Host: your.wiki.org
> User-Agent: curl/7.64.1
> Accept: */*
> Content-Type: application/xml
> Content-Length: 88
>
* upload completely sent off: 88 out of 88 bytes
< HTTP/1.1 200 OK
< Server: nginx
< Date: Sun, 12 Sep 2021 17:05:40 GMT
< Content-Type: text/xml; charset=utf-8
< Transfer-Encoding: chunked
< Connection: close
< X-Clacks-overhead: GNU Terry Pratchett
< Vary: Cookie
< Expires: Thu, 19 Nov 1981 08:52:00 GMT
< Cache-Control: no-store, no-cache, must-revalidate
< Pragma: no-cache
< Strict-Transport-Security: max-age=15768000
<
<?xml version="1.0"?>
<methodResponse>
  <params>
    <param>
      <value>
        <string>Release 2018-04-22c &amp;quot;Greebo&amp;quot;</string>
      </value>
    </param>
  </params>
</methodResponse>
* Closing connection 0
* TLSv1.2 (OUT), TLS alert, close notify (256):
```

Note that you could also get a «not authorized» response. This still indicates that DW XMLRPC works perfectly. it simply means that you must authenticate using the dokuwiki.login method before you can get this information. How to authenticate is not shown here.

```
<?xml version="1.0"?>
<methodResponse>
```

```
  <fault>
    <value>
      <struct>
        <member>
          <name>faultCode</name>
          <value><int>-32603</int></value>
        </member>
        <member>
          <name>faultString</name>
          <value><string>server error. not authorized to call method
dokuwiki.getVersion</string></value>
        </member>
      </struct>
    </value>
  </fault>
</methodResponse>
```

Sample TypeScript/Node/JavaScript Client

Use @glen/wiki-rpc-client package:

```
#!/usr/bin/env node -r ts-node/register

import { WikiRpcClient, WikiService, DokuwikiService } from "@glen/wiki-rpc-
client";

const main = async () => {
  const url = "https://www.dokuwiki.org/lib/exe/xmlrpc.php";
  const options = {
    basic_auth: {
      user: "glen",
      pass: "password",
    },
  };

  const client = WikiRpcClient.create<WikiService & DokuwikiService>(url,
options);

  const dwVersion = await client["dokuwiki.getVersion"]();
  const data = await client["wiki.getPage"]("start");

  console.log(dwVersion, data);
};

main().catch((e: Error) => console.error(e));
```

Sample Java Client

A java client for Dokuwiki is available on github.

Example which displays the title of the wiki and the list of its pages:

```
import dw.xmlrpc.DokuJClient;
import dw.xmlrpc.Page;

public class Main {
  public static void main(String[] args) throws Exception{
    String url = "http://mywiki/lib/exe/xmlrpc.php";
    String user = "myUser";
    String pwd = "myPassword";

    DokuJClient client = new DokuJClient(url, user, pwd);
    System.out.println("Pages in the wiki " + client.getTitle() + " are:");
    for(Page page : client.getAllPages()){
      System.out.println(page.id());
    }

  }
}
```

Sample PHP Client

This simple example will pull the version information from the given DokuWiki.

- You need to download and install the library XML-RPC for PHP for this example.
- The URL of the DW-Server must be given **without** the procotol (http://).

```
<?php
include('lib/xmlrpc.inc');  // or whereever you have downloaded and put the
XML-RPC for PHP libraries

// create a new client instance
$c = new xmlrpc_client('/lib/exe/xmlrpc.php', 'adminwiki.kreisbote.de', 80);

// enable debugging to see more infos :-) (well, not for production code)
$c->setDebug(1);

// create the XML message to send
$m = new xmlrpcmsg('dokuwiki.getVersion');

// send the message and wait for response
$r = $c->send($m);
if($r == false) die('error');
if(!$r->faultCode()){
```

```
    // seems good. Now do whatever you want with the data
    $v = php_xmlrpc_decode($r->value());
    echo "$v";
}
```

```
<?php
include('/lib/xmlrpc.inc');

$c = new xmlrpc_client('/lib/exe/xmlrpc.php', 'localhost', 80);
$c->setDebug = 1;

$m = new xmlrpcmsg('wiki.putAttachment');
$m->addParam(new xmlrpcval("file.png", "string"));
$m->addParam(new xmlrpcval("iVBORw0KGgoAAAANSUhEUgAAADAAAAAwC.....<base64
string here>...", "base64"));
$m->addParam(new xmlrpcval(array("ow" => new xmlrpcval(true, "boolean")),
"struct"));

$r = $c->send($m);
if($r == false) die('foo!');

// Output the response object
// var_dump($r);
$r = $c->send($m);

if(!$r->faultCode()) {
    $v = php_xmlrpc_decode($r->value());
    echo $v;
} else {
    echo $r->faultString();
}

?>
```

DokuWiki has its own XML-RPC library which could also be used. A simple client using that library looks like this:

```
<?php
namespace dokuwiki\Remote\IXR;
require_once('inc/init.php');
/*
Previous code for setting debugging to accommodate use of Remote\IXR:
$client = new
IXR_Client('http://localhost/devel/dokuwiki/lib/exe/xmlrpc.php');
$client->debug = 1; // enable for debugging
*/
$client = new Client( $url);
$http = $client->getHttpClient();
$http ->debug = $debug; // enable for debugging
$client->query('dokuwiki.login','admin','pass');
$ok = $client->getResponse();
```

```
if($ok) {
    $client->query('wiki.getPage','wiki:syntax');
    echo $client->getResponse();
}
```

Sample Perl Client

Dokuwiki::RPC::XML::Client is simple dokuwiki client written on top of RPC::XML::Client, it comes with a CLI command you can use from shell.

you can also write RPC::XML::Client code directly following this simple example will pull the version information and the list of documents in the `wiki` namespace from the given Dokuwiki. It also lists all `.png` mediafiles from the same namespace and its child namespaces.

```
use strict;
use warnings;
use feature 'say';
use RPC::XML::Client;
use Data::Dumper;

my $client =
  RPC::XML::Client->new('https://www.example.com/wiki/lib/exe/xmlrpc.php');

my $res = $client->send_request('dokuwiki.getVersion');
say 'dokuwiki.getVersion = ' . $res->value;

my $namespace = 'wiki';
my $req =
  RPC::XML::request->new( 'dokuwiki.getPagelist',
    RPC::XML::string->new($namespace),
    RPC::XML::struct->new() );
$res = $client->send_request($req);
say "dokuwiki.getPagelist = \n" . Dumper $res->value;


# if given and >0 'depth' is relative to the searched namespace and is a
maximum.
# setting 'hash' to 1 will put extra load on server.
$res = $client->send_request(
    RPC::XML::request->new( 'wiki.getAttachments',
        RPC::XML::string->new('$namespace'),
        RPC::XML::struct->new( {depth => 2, skipacl => 1, hash => 0, pattern
=> '/\.png$/' } )
    )
);
say "wiki.getAttachments = \n" . Dumper $res->value;
```

Sample Ruby Client

This simple example will pull the version information from the given Dokuwiki. No plugin required (in the Windows default package).

```
require "xmlrpc/client"
server = XMLRPC::Client.new( "localhost","/dokuwiki/lib/exe/xmlrpc.php")
begin
    puts server.call("dokuwiki.getVersion")
rescue XMLRPC::FaultException => e
    puts "Error:"
    puts e.faultCode
    puts e.faultString
end
```

This example demonstrates how to upload attachments:

```
require "xmlrpc/client"
server = XMLRPC::Client.new( "localhost","/dokuwiki/lib/exe/xmlrpc.php")
begin
    puts server.call("wiki.putAttachment", "wiki/example.png",
XMLRPC::Base64.new(IO.read("/path/to/example.png")))
rescue XMLRPC::FaultException => e
    puts "Error:"
    puts e.faultCode
    puts e.faultString
end
```

Sample jQuery Client

```
var xml='<?xml
version="1.0"?><methodCall><methodName>wiki.getRPCVersionSupported</methodNa
me><params></params></methodCall>';
 $.ajax({
   url: "http://<your wiki>/wiki/lib/exe/xmlrpc.php",
   data: xml,
   contentType:"text/xml",
   type:"post",
   beforeSend: function (xhr) {
     xhr.setRequestHeader ("Authorization", "Basic " + btoa('username' + ":"
+ 'password'));
   },
 });
```

Python Module

A Python module for DokuWiki's XML-RPC API can be found at the Python package index and on GitHub.

Alternative: python-dokuwiki

Sample AutoIt3 Client

My AutoIt3 client is mainly a proof of concept:

```
#cs ----------------------------------------------------------------------------

 AutoIt Version: 3.2.12.1
 Author:         Hugo Bossard

 Script Function:
    XML-RPC sample client AutoIt3 script.

#ce ----------------------------------------------------------------------------

; this script uses the UDF WinHttp
#include "WinHTTP.au3"

; get the info from page 'wiki:syntax' using wiki.getPageInfo()
$Xml = '<?xml version="1.0"?>'
$Xml = $Xml  & @CRLF & '<methodCall>'
$Xml = $Xml  & @CRLF & '<methodName>wiki.getPageInfo</methodName>'
$Xml = $Xml  & @CRLF & '<params>'
$Xml = $Xml  & @CRLF & '<param>'
$Xml = $Xml  & @CRLF & '<value>'
$Xml = $Xml  & @CRLF & '<string>wiki:syntax</string>'
$Xml = $Xml  & @CRLF & '</value>'
$Xml = $Xml  & @CRLF & '</param>'
$Xml = $Xml  & @CRLF & '</params>'
$Xml = $Xml  & @CRLF & '</methodCall>'
$XmlSize = StringLen($Xml)

; initializes the use of WinHTTP functions and returns a WinHTTP-session
handle.
$hw_open = _WinHttpOpen()

; creates a HTTP request handle to a lokal DokuWikiStick installation on
port 8800
$hw_connect = _WinHttpConnect($hw_open, "localhost", 8800)
```

```
; creates a HTTP request handle
$h_openRequest = _WinHttpOpenRequest($hw_connect, "POST",
"/lib/exe/xmlrpc.php", "/RPC2 HTTP/1.0", "localhost", "text/xml")

; sends the specified request to the HTTP server
_WinHttpSendRequest($h_openRequest, "", $Xml, $XmlSize)

; waits to receive the response to a HTTP request initiated by
WinHttpSendRequest().
_WinHttpReceiveResponse($h_openRequest)

$Data = ""
Do
  ; reads data from a handle opened by the _WinHttpOpenRequest() function.
  $Data&=_WinHttpReadData($h_openRequest)
Until @error

; close connection and clean up
_WinHttpCloseHandle($h_openRequest)
_WinHttpCloseHandle($hw_connect)
_WinHttpCloseHandle($hw_open)

; show error number and resulting data
MsgBox(4096,"Error: " & @error, $Data)
```

Sample Emacs Lisp Client

```
(require 'xml-rpc)
(defvar *dokuwiki-xml-rpc-url* "https://dokuwiki.url/lib/exe/xmlrpc.php")
(xml-rpc-method-call
  *dokuwiki-xml-rpc-url*
  'dokuwiki.getTime)
```

PowerShell Module

A PowerShell client for Dokuwiki is available on the powershell gallery and on github.

From:
https://wwoss.ru/ - **worldwide open-source software**

Permanent link:
https://wwoss.ru/doku.php?id=wiki:devel:xmlrpc:clients

Last update: **2025/01/15 10:26**