

EmitterInterface

Server + const DEPTH\_INFINITY + const NS\_SABREDAV

+ \$tree

+ \$sapi

+ \$xml

# \$baseUri # \$plugins

+ start() + exec()

+ setBaseUri(\$uri) + getBaseUri() + guessBaseUri()

\$plugin) + getPlugin(\$name) + getPlugins() + getLogger() et 24 de plus...

+ \$httpResponse + \$httpRequest

+ \$transactionType + \$protectedProperties + \$debugExceptions + \$resourceTypeMapping

+ static \$exposeVersion

+ static \$streamMultiStatus

+ \_\_construct(\$treeOrNode

+ addPlugin(ServerPlugin

## Client + const STATUS\_SUCCESS + const STATUS\_CURLERROR + const STATUS\_HTTPERROR # \$curlSettings # \$throwExceptions # \$maxRedirects # \$headerLinesMap + \_\_construct() + send(RequestInterface \$request) + sendAsync(RequestInterface \$request, callable \$success =null, callable \$error=null) + poll() + wait() + setThrowExceptions (bool \$throwExceptions) + addCurlSetting(int \$name, \$value) # receiveCurlHeader (\$curlHandle, \$headerLine) # doRequest(RequestInterface \$request) # createCurlSettingsArray (RequestInterface \$request) # parseCurlResponse (array \$headerLines, string \$body, \$curlHandle)

# parseCurlResult(string \$response, \$curlHandle)

# curlExec(\$curlHandle) # curlStuff(\$curlHandle)

+ \$xml

+ \$propertyMap

# \$baseUri # \$encoding

+ options()

\$body)

+ request(\$method, \$url=", \$body=null, array \$headers=[]) + getAbsoluteUrl(\$url) + parseMultiStatus(

\$settings) + propFind(\$url, array \$properties, \$depth=0) + propPatch(\$url, array \$properties)

+ const AUTH\_BASIC + const AUTH\_DIGEST + const AUTH\_NTLM

\_construct(array

(RequestInterface \$request, callable \$success, callable \$error, int \$retryCount=0)

Client

+ const ENCODING\_IDENTITY + const ENCODING\_DEFLATE + const ENCODING\_GZIP + const ENCODING\_ALL

# sendAsyncInternal