

Webget

Program Documentation

Webget is a command-line tool for retrieving files from webservers and FTP servers. It is ideal for automating file retrieval tasks, especially where one needs to query a large number of files. It has an array of special capabilities, including the ability to submit POST requests and to modify standard HTTP requests, which also make it useful for testing server capabilities.

Contents	Page
The Command-Line Interface	2
Command Line Parameters and Options	3
Technical Support and Further Information.....	5

The Command-Line Interface

Webget is a command-line program, so you must execute it from a command prompt or from the Windows Run command. As a command-line tool, it is designed for easy integration with other software or processes: it can be run automatically from a batch file, or by executing it as a separate process from within another program.

Running the program from a command prompt without any arguments gives the following usage summary:

```
v4.173.553 (c) 2006 Tennyson Maxwell Information Systems, Inc.

Retrieves a file from the Internet

Usage: webget [-shbrcauefndptwq] <urlstr>
  urlstr  url to get
  -s     save content filename (use -sconsole to output to console)
  -h     get only the HTTP response header
  -b     save even if return status is bad
  -r     re-request partial downloads N times until file is received
  -c     complete downloading using existing file (cannot use console output)
  -a     account:password (can also be given in url)
  -u     user-agent identity (default=none)
  -e     extra HTTP query lines (separate lines with double vbars ||)
  -f     file to post
  -n     allow up to N redirections (default=3)
  -d     retry 'unavailables' up to N times with M ms delay (default=1:500)
  -p     proxy to use (ip:port or 'auto' to use browser's proxy settings)
  -t     communications timeout in seconds (default=60)
  -w     use WinInet functions
  -q     quiet operation (no progress messages)
```

Note that you must be in the same directory as the Webget.exe program in order to run the program. Alternatively, you must specify the directory location of the program on your executable path; or you may give the full path to the program as part of the command line.

To change your executable path to include the path to Webget under Windows 9x/Me, you must edit the autoexec.bat file on your boot drive's root directory. At the end of the line beginning "PATH=", append a semicolon and then the complete path of the Webget program directory.

To change your executable path to include the path to Webget under the other Windows operating systems, you must edit the PATH variable in your System Environment. Using the Start button, choose Settings+Control Panel, choose the System icon, and select the Environment tab. Edit the PATH variable; at end of the path, append a semicolon and then the complete path of the Webget program directory.

Command Line Parameters and Options

Webget has one required parameter: the URL to the file you want to retrieve. This should be given in the same fashion that you would write it in your browser's address line; for example:

```
webget http://www.server.com/path/file.htm
```

The "http://" can be omitted. Webget assumes a URL is an HTTP request, unless you specify an alternate protocol, such as HTTPS or FTP, or use a local file address such as c:\file.htm or \\server\share\file.htm.

You can embed any required authentication in the URL using the form

```
webget http://username:password@www.server.com/path/file.htm
```

URLs that include spaces or question marks should be enclosed in quotation marks.

Webget also provides a number of options which you can specify on the command line. To specify an option, add its letter to the command line, prefixed by a dash ("-") or slash ("/") character.

When specifying additional information as part of the option (for example, to specify the filename to save), do not put any space between the option and the extra information.

The order of options and the URL parameter is irrelevant; you may specify them in any order.

Available options include—

- s save content filename (use -sconsole to output to console): Use this option to specify the filename that Webget will use to save the file it retrieves. If you omit the -s option, Webget will choose the name based on the file's URL. Using "console" as the file name will output the contents of the file to the console. (You should generally do this only with text files, or if you are redirecting the output to a file or pipe.)
- b save even if return status is bad: Use this option to force Webget to save the file even if the status code is "bad" — that is, a 400 or higher error code is returned by the server (or a 300 if redirections are not to be followed; see below). Normally, Webget will only save a file if the status code is "good" (200-299).
- r re-request partial downloads N times until file is received: You can use this option to facilitate large downloads, where it is likely that the download will be interrupted. When -r is specified, Webget will repeatedly request the file until the download is complete. The repeat requests perform "partial" downloads — that is, Webget requests only the missing portion of the file, so that the part that is already downloaded doesn't need to be retrieved again.
- c complete downloading using existing file (cannot use console output): Use this option in conjunction with -r, if you have already downloaded part of the file earlier. When -c is omitted, any existing file will be overwritten when retrieval begins.
- a account:password: Use this option to specify a username and password to be used when retrieving the file, if you don't want to embed the authentication in the URL directly. **Note** that in earlier versions of Webget, the account/password were given in the format "account/password." For reasons of compatibility with the prevailing format used on the Web, this has been changed to "account:password."
- u user-agent identity (default=none): You can specify the user-agent string to be passed with HTTP requests, using this option. A good user-agent string is

```
"-uMozilla/4.0 (compatible; MSIE 6.0; Windows NT)"
```

- e extra HTTP query lines (separate lines with double vbars ||): You can insert additional HTTP query parameters using this option. To specify a cookie, for example, you could use

```
"-eCookie: SHOPPER%5FID=236570453"
```

To specify multiple parameters, separate them with double vertical bars (||), like this:

```
"-eCookie: SHOPPER%5FID=236570453||Proxy-authentication: Basic =38483dk782df"
```

- f file to post: You can use this option to post a file using the POST protocol. The `-f` option should be followed with the path to the file to send with the request.
- n specifies the number of redirections Webget is allowed to follow. If `-n` or `-n0` is specified, Webget will not follow any redirections. By default, Webget will automatically request a chain of up to three redirected URLs, if they are sent by the server(s).
- d directs Webget to retry temporarily unavailable files up to N times with an M millisecond delay between requests (default=1:500).
- p proxy to use (ip:port or 'auto' to use browser's proxy settings): Use this option to specify a CERN HTTP proxy, if you wish to use one for HTTP or FTP retrieval. For example,

```
-plocalhost:8080
```

will force Webget to communicate through a proxy listening on port 8080 on the local machine.

- t communications timeout in seconds (default=60): Sets the timeout for all communications. If Webget has to wait for longer than the timeout period before getting data, it aborts with a timeout error.
- w use WinInet functions: You can use this option to route all communications through the Wininet system (essentially, Internet Explorer) on Windows computers. This option is useful when Internet Explorer on your machine is set up to use autodiscovered proxies or proxy configuration scripts. With this option enabled, Webget will be able to request files using those proxies. The Wininet system can be slightly slower, however, and it may impose additional constraints on the number of simultaneous requests that you can make (a concern only if you use many instances of Webget in parallel on the same machine).
- q quiet operation (no progress messages): When this option is enabled, Webget does not output progress messages. (Normally, it notifies you of each stage in the communication: for example, connecting, connected, requesting, requested, receiving... complete).

Technical Support and Further Information

Purchasing Webget entitles you to unlimited free technical support for the program, by email and by telephone. For all technical support questions, please contact the Support Team at our Los Angeles office:

Tennyson Maxwell Information Systems, Inc.

PO Box 2341

Los Angeles, CA 90078

+1 323-461-9512 voice

+1 323-461-9514 fax

email: support@tenmax.com