

Download from Git

Git is a distributed version control system. It allows you to download the very latest version of the source code, with all the branches and tagged releases at your disposal.

You should download from Git if you are a developer and want to submit patches.

If you do **not** want to develop but only install MediaWiki and extensions, then download stable tarball releases instead.

See [Git](#) for more details, particularly for making contributions. Below are some quick directions for a couple common tasks.

Prerequisites

You must have Git installed before you can use it. There are many ways to acquire Git, depending on your operating system. Follow [Gerrit/Tutorial#Set up Git](#), or use your favorite search engine.

It is recommended that you have [Composer](#) installed to download and install third party libraries, but not required.

Using Git to download MediaWiki

Download

You can download MediaWiki core using Git, as well as all extensions currently installed on the Wikimedia Foundation server cluster and many other extensions hosted on Gerrit.

The first step is to clone the MediaWiki core repository. This will take a while.

Download for development

The latest development version of MediaWiki is tracked in the 'master'

branch.

First make sure that you have created a developer account, so that you have an ssh username.

Then, in a terminal window, enter the following command to clone with your ssh *<USERNAME>* so that you can submit patches for review:

```
git clone  
ssh://<USERNAME>@gerrit.wikimedi  
a.org:29418/mediawiki/core.git  
mediawiki
```

This clones the entire MediaWiki core repository, synced to the master branch, into a sub-directory named `mediawiki`.

To install into a different directory, change that in the command line (for more info refer to these docs (<https://git-scm.com/docs/git-clone>)). Once the repository is cloned, you can switch to different branches or tags. The development branch, `master`, is the cutting-edge, developer version of MediaWiki; you **should not use master code for production** under any circumstances as it is not considered stable.

- [View latest source code](#)

Download a stable branch

If you do not want to develop software patches but want to anonymously clone the stable release branch 1.39, use this command instead:

```
git clone  
https://gerrit.wikimedia.org/r/media  
wiki/core.git --branch REL1_39  
mediawiki
```

If you have a slow internet connection and want to reduce the number of

revisions that are cloned, add `--`
`depth=1` to the `git clone` command.

MediaWiki tags (stable version)

Alternatively, specific stable versions of MediaWiki are tracked using 'tags'. These are analogous to the tarball releases.

Currently, these are 1.39.3 (stable), 1.39.3 (LTS) and 1.35.10 (legacy LTS).

You can see all available tags with:

```
git tag -l | sort -V
```

To use a specific tag, e.g. the latest stable release:

```
git checkout 1.39.3
```

Update the Git submodules

The branches and tags have a bunch of Git submodules (https://gerrit.wikimedia.org/r/plugins/gitiles/mediawiki/core/+refs/heads/REL1_39/.gitmodules) in them, for commonly-used extensions and skins and for the `vendor/` directory. The `master` branch does not have these. To update the submodules, run:

```
cd mediawiki
git submodule update --init --recursive
```


Fetch external libraries

MediaWiki uses Composer to manage external PHP libraries, all of which end up in the `vendor/` directory in your MediaWiki directory.

To install these needed libraries, you have a choice:

- Download and install the composer PHAR (<https://getcomposer.org/doc/00-intro.md#globally>) , optionally rename the composer.phar file as instructed for your OS, and then run `composer update --no-dev` from your MediaWiki directory. **This is the recommended approach.**

- If the default PHP CLI version does not match the web server's PHP version, specify it with e.g. `php7.4 composer.phar update --no-dev`.
- Or, if you don't want to use Composer or if you want to use the exact same set of vendor libraries as used on the WMF production cluster, you can instead create a `vendor/` directory inside the core folder of your MediaWiki installation:
 - With your developer account, use this command: `git clone ssh://<USERNAME>@gerrit.wikimedia.org:29418/mediawiki/vendor.git`

- For anonymous checkout, use this command: `git clone https://gerrit.wikimedia.org/r/mediawiki/vendor.git`
- Note that if any of your extensions have their own Composer requirements, then **you cannot use this option.**

Prior to MediaWiki 1.25, external libraries were kept in the core repository, and no package manager was required.

Keeping up to date

If you're using a particular branch or the development version ("master" branch) of

MediaWiki, picking up the latest changes is relatively easy. Change into your MediaWiki clone directory and issue this command:

```
git pull
```

All of the latest changes for the branch you are using will be applied.

The new version of core may require newer versions of extensions and skins, so you must go into each extension and skin directory and update it with a command like `git pull --recurse-submodules`.

You also need to update `vendor/` with any newer versions of required libraries. This often means running the following Composer command, but see [#Fetch external libraries](#) above for more details:

```
composer update --no-dev
```

After updating/upgrading the code and required libraries you should run the MediaWiki `update.php` command-line script to update database tables as needed:

```
php maintenance/update.php
```

If you use MediaWiki-Vagrant, it offers a single command, `vagrant git-update`, that performs all these steps.

Switching to a different version

Each of our versions are tracked as branches or tags. In order to switch to a different version (for example from the `master` branch to a different branch or tag), checkout (<https://git-scm.com/book/en/v2/Git-Branching-Basic-Branching-and-Merging>). the particular branch or tag you want from within your MediaWiki clone directory:

```
git checkout <branch_name>
```

or

```
git checkout <tag_name>
```

The changes will be applied automatically and you will be all set to go.

Using Git to download MediaWiki extensions

List of extensions in git (<https://gerrit.wikimedia.org/r/admin/repos/q/filter:mediawiki+extensions>)

Download an extension

In the next commands, replace <EXT> with the name of the extension you want to download, without spaces. For Extension:TitleKey, it would be TitleKey. (case sensitive!)

Download and clone an extension from Git:

With your developer account, use these commands to get the master branch:

```
cd /path/to/extensions  
git clone  
ssh://<USERNAME>@gerrit.wikimedi
```



```
a.org:29418/mediawiki/extensions/<EXT>
```

For anonymous checkout of the stable branch instead, use these commands:

```
cd /path/to/extensions  
git clone  
https://gerrit.wikimedia.org/r/mediawiki/extensions/<EXT> --branch  
REL1_39
```

You can view extension source code in Gerrit's gittiles (<https://gerrit.wikimedia.org/r/plugins/gittiles/>) application, and at the URL:

```
https://gerrit.wikimedia.org/g/mediawiki/extensions/<EXT>/+/refs/head  
s/master
```

Download all extensions

If you prefer to have *all* MediaWiki extensions that are at `gerrit.wikimedia.org` checked out to your machine, enter the following:

With your developer account, to get the master branch:

```
git clone  
ssh://<USERNAME>@gerrit.wikimedi
```

```
a.org:29418/mediawiki/extensions
```

For anonymous checkout of the stable branch instead, use this command:

```
git clone  
https://gerrit.wikimedia.org/r/mediawiki/extensions --branch REL1_39
```

After running the `git clone` command, continue with these commands:

```
cd /path/to/extensions  
git submodule update --init --recursive
```

At any time, to update all extensions to their latest versions of that branch, enter:

```
cd /path/to/extensions  
git pull  
git submodule update --init --  
recursive
```

To change to a different branch, such as after a new release:

```
git submodule foreach 'git checkout  
-b REL1_38 origin/REL1_38 || :'
```

Remember that you should only use versions of extensions from the same

release as that version of MediaWiki and each other.

To track the master branch:

```
git submodule foreach 'git checkout  
-b origin/master || :'
```

Please note that **you should not use master code for production** under any circumstances as it is not considered stable.

If you only need a read-only checkout (for instance to grep or analyse all MediaWiki code), you can use the shared MediaWiki

checkout on Labs, without downloading anything on your machines.

Remove an extension

1. Remove the " `require_once ...` " or " `wfLoadExtension(...)` " from `LocalSettings.php`
2. Remove any line referencing the extension in `composer.local.json` (usually in the "extra → merge-plugin → include" section)
3. Remove the extension's directory in `install-dir/extensions/`

Using Git to download MediaWiki skins

List of skins in git (<https://gerrit.wikimedia.org/r/admin/repos/q/filter:mediawiki%252Fskins>).

MediaWiki 1.24 and later don't include skins in the Git download.

Follow the exact same procedure as for extensions (described in the previous section), but using `skins` rather than `extensions` in all URLs and paths.

Detailed installation instructions are available on each skin's page here on MediaWiki.org, for example see

Skin:Vector#Installation. Instructions for all other skins are analogous.

See also

- Intranet/Intranet Installation, part of a fully worked series of articles that uses Git for MediaWiki deployment

Appendix

The Revision as of 14:26, 21 March 2019 changed the standard of linking to gerrit.wikimedia.org:

from:

gerrit.wikimedia.org/r/p/mediawiki

to:

gerrit.wikimedia.org/r/mediawiki

Retrieved from

"[https://www.mediawiki.org/w/index.php?title=Download from Git&oldid=5848842](https://www.mediawiki.org/w/index.php?title=Download_from_Git&oldid=5848842)"

MediaWiki

This page was last edited on 26 March 2023, at 10:35. •

Content is available under [CC BY-SA 4.0](#) unless otherwise noted.