

About large files on GitHub

In this article

About size limits on GitHub Enterprise Server

Removing files from a repository's history

Distributing large binaries

GitHub Enterprise Server limits the size of files you can track in regular Git repositories. Learn how to track or remove files that are beyond the limit.

[Mac](#) [Windows](#) [Linux](#)

About size limits on GitHub Enterprise Server

GitHub Enterprise Server tries to provide abundant storage for all Git repositories, although there are hard limits for file sizes and recommendations for repository sizes.

File size limits

GitHub Enterprise Server limits the size of files allowed in repositories. If you attempt to add or update a file that is larger than 50 MiB, you will receive a warning from Git. The changes will still successfully push to your repository, but you can consider removing the commit to minimize performance impact. For more information, see "[Removing files from a repository's history](#)."

Note: If you add a file to a repository via a browser, the file can be no larger than 25 MiB. For more information, see "[Adding a file to a repository](#)."

By default, GitHub Enterprise Server blocks files larger than 100 MiB. However, a site administrator can configure a different limit for your GitHub Enterprise Server instance. For more information, see "[Enforcing repository management policies in your enterprise](#)."

To track files beyond this limit, you must use Git Large File Storage (Git LFS). For more information, see "[About Git Large File Storage](#)."

If you need to distribute large files within your repository, you can create releases on your GitHub Enterprise Server instance instead of tracking the files. For more information, see "[Distributing large binaries](#)."

Git is not designed to handle large SQL files. To share large databases with other developers, we recommend using a file sharing service.

Repository size recommendations

We recommend repositories remain small, ideally less than 1 GB, and less than 5 GB is strongly recommended. Smaller repositories are faster to clone and easier to work with and maintain.

You can prevent your repository from impacting your infrastructure by effectively managing your repository's size and overall health. You can find advice and a tool for repository analysis in the [github/git-sizer](#) repository.

Removing files from a repository's history

Warning: These procedures will permanently remove files from the repository on your computer and your GitHub Enterprise Server instance. If the file is important, make a local backup copy in a directory outside of the repository.

Removing a file added in the most recent unpushed commit

If the file was added with your most recent commit, and you have not pushed to your GitHub Enterprise Server instance, you can delete the file and amend the commit:

- 1 Open TerminalTerminalGit Bash.
- 2 Change the current working directory to your local repository.
- 3 To remove the file, enter `git rm --cached :`

```
$ git rm --cached GIANT_FILE
# Stage our giant file for removal, but leave it on disk
```

- 4 Commit this change using `--amend -CHEAD`:

```
$ git commit --amend -CHEAD
# Amend the previous commit with your change
# Simply making a new commit won't work, as you need
# to remove the file from the unpushed history as well
```

- 5 Push your commits to your GitHub Enterprise Server instance:

```
$ git push
# Push our rewritten, smaller commit
```

Removing a file that was added in an earlier commit

If you added a file in an earlier commit, you need to remove it from the repository's history. To remove files from the repository's history, you can use the BFG Repo-Cleaner or the `git filter-repo` command. For more information see "[Removing sensitive data from a repository](#)."

Distributing large binaries

If you need to distribute large files within your repository, you can create releases on your GitHub Enterprise Server instance. Releases allow you to package software, release notes, and links to binary files, for other people to use. For more information, visit "[About releases](#)."

Legal

