Version control System

In [ren revision control systems, a](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)**[repository](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)**[[[1]](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Repository_(version_control)#cite_note-1)[is a data structure which stores](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[metadata](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Metadata)[for a set of files or directory structure. Depending on whether the version control system in use is distributed (for instance,](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[Git](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Git_(software))[or](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[Mercurial](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Mercurial)[) or centralized (](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[Subversion](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Apache_Subversion)[or](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[Perforce](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Perforce)[, for example), the whole set of information in the repository may be duplicated on every user's system or may be maintained on a single](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[server](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Server_(computing))[. Some of the metadata that a repository contains includes, among other things:](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)

* [A historical record of changes in the repository.](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)
* [A set of commit objects.](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)
* [A set of references to commit objects, called](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)*[heads](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)*[.](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)

[Storing changes[](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[edit](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/w/index.php?title=Repository_(version_control)&action=edit&section=1)[]](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)

[The main purpose of a repository is to store a set of files, as well as the history of changes made to those files.](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[[2]](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Repository_(version_control)#cite_note-2)[Exactly how each revision control system handles storing those changes, however, differs greatly: for instance, Subversion has in the past relied on a database instance and has since moved to storing its changes directly on the filesystem.](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[[3]](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Repository_(version_control)#cite_note-3)[These differences in methodology have generally led to diverse uses of revision control by different groups, depending on their needs.](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)[[[4]](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control)](https://en.wikipedia.org/wiki/Repository_(version_control)#cite_note-4)

[vision control systems](https://en.wikipedia.org/wiki/Revision_control" \o "Revision control), a repository[[1]](https://en.wikipedia.org/wiki/Repository_(version_control)#cite_note-1) is a data structure which stores [metadata](https://en.wikipedia.org/wiki/Metadata) for a set of files or directory structure. Depending on whether the version control system in use is distributed (for instance, [Git](https://en.wikipedia.org/wiki/Git_(software)" \o "Git (software)) or [Mercurial](https://en.wikipedia.org/wiki/Mercurial)) or centralized ([Subversion](https://en.wikipedia.org/wiki/Apache_Subversion) or [Perforce](https://en.wikipedia.org/wiki/Perforce), for example), the whole set of information in the repository may be duplicated on every user's system or may be maintained on a single [server](https://en.wikipedia.org/wiki/Server_(computing)). Some of the metadata that a repository contains includes, among other things:

* A historical record of changes in the repository.
* A set of commit objects.
* A set of references to commit objects, called *heads*.

Storing changes

The main purpose of a repository is to store a set of files, as well as the history of changes made to those files.[[2]](https://en.wikipedia.org/wiki/Repository_(version_control)#cite_note-2) Exactly how each revision control system handles storing those changes, however, differs greatly: for instance, Subversion has in the past relied on a database instance and has since moved to storing its changes directly on the filesystem.[[3]](https://en.wikipedia.org/wiki/Repository_(version_control)#cite_note-3) These differences in methodology have generally led to diverse uses of revision control by different groups, depending on their needs.[[4]](https://en.wikipedia.org/wiki/Repository_(version_control)#cite_note-4)