**SOURCE CODE MANAGEMENT**

**What is source code management?**

A source code management is a tool used by programmers to control different versions of the code.

It is primarily used to give versions to the code, time lapse of the code and the author of that changes in the code.

**Basics of SCM:**

When a company is developing a big, long time project. There are many things to be managed partly. There are many people included, they all develop their parts of the codes. To integrate or to manage new changes in the code we need some type of code management system along with the changes made. So, then this SCM has come in the way to clear all the problems of code management.

Suppose an employee got out of the project and it was assigned by another employee so to understand the last developed and changes made to that development this type of SCM will be used.

**Feature of SCM:**

* Authenticated access for commits
* Revision history on files
* Atomic commits of multiple files
* Versioning/Tagging

**Why do we need SCM?**

Backup and restore: suppose if the project is going on for long time we need to store the files and also retrieve them if required.

Synchronization: To have combined set of all codes in the project makes it easy to develop and understand.

Branching: giving different versions of product so that every developer can have they’re of development

Merging: combining all the developed codes to integrate as a single product and test.

**Different types of version control/source code:**

There are two types of version controls in the SCM

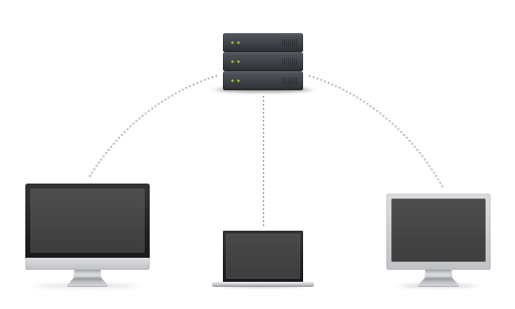
1. Centralized version control system
2. Distributed version control system

**Centralized version control system:**

It works as a client and server relationship system. The main repository will be in some place like cloud and will be accessed by multiple clients.

Its easy to understand and have more control over users and access.

But it depends on the access of the server so it might be slow to execute some commands. It only works on the internet.

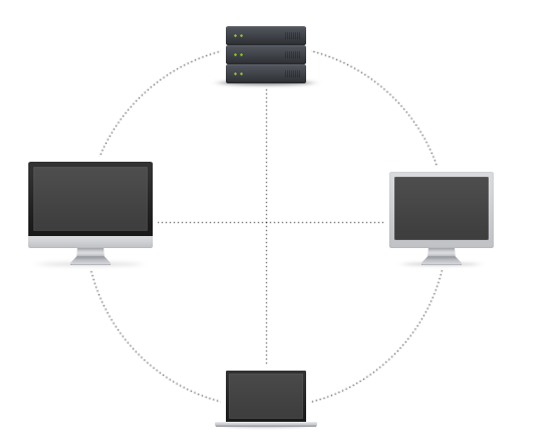


**Distributed version control system:**

In this system every user has their own local repository and files and history. So, this benefits more than the CVCS because it doesn’t need internet connection all times.

No need of centralized servers and branching, merging can be easily done very fast and easily.

It’s easier than the other type of VCS.



**SCM/VC tools:**

* GitHub
* GitLab
* BitBucket
* SourceForge
* Beanstalk
* Apache Allura
* AWS CodeCommit
* Launchpad
* Phabricator
* GitBucket