Version Control is a system that records changes to a file or set of files over time so that you can recall specific version later, a version control system is vital in software development because of this ability, it is a central location in which data is stored and managed which then can be tracked. Every change made to the source data is tracked so that people will know why the alteration was made, what the problem was in the first place and the enhancements introduced. It allows developers to keep a record of their changes becoming critical when collaborating on projects because other people in the group need to know what changes were made and why to gain a better understanding and sense of comprehension. A version control system acts as a growth chart for a project which allows participants to stay up to date with each other’s improvements. Without version control, it would be near to impossible to have lots of people work on a project together. If something went awry, there wouldn’t be a record that allowed the developers to easily locate the fault. In essence version control is important because it allows large groups of software developers to keep track of changes made to data and help others in the group understand why the changes were made. Version control keeps the developers in contact and ensures that the project will run smoothly by using code modification history (a complete history of every change made by a developer), concurrent file editing (allows multiple developers to edit at once), tagging (the ability to create a snapshot of codebase at any given moment), branching (branches let you easily maintain your “in-progress” work separately from your completed code) and merging (allowing many user to use the same file at once).

Git

Git is another type of a version control system. It stores and considers data differently than other systems, it is primarily used for source code management in software development, but can be used to keep track of changes in any sort of file set. There are many advantages to using Git over normal CVS and other version control systems. For instance when you want to switch branches there is no need to connect to a remote server or close your project and open a checkout in another directory. Another advantage of Git is that you can merge without connecting to a remote server, all the branches are local. In Git you could create a new branch to perform the merge in so that if you had a large amount of conflicts you could incrementally work on and commit without affecting the main branch, then merge or create a patch after everything has been fixed. Git also allows developers to connect to each other’s repositories. This allows teams to share code without affecting the central repository. One of the biggest advantages of Git would be its branching abilities. They are unlike any centralized version control systems because Git branches are cheap and easy to merge facilitating the feature branch workflow. Using feature branches is not only more reliable that directly editing production code, but it also provides more organizational benefits.

Many of Gits features are designed around improving or being able to deal with the development workflow. Tracking has been implemented in Git which is important because it tracks the content. This is important from whichever angle you look at it, however Git is capable of tracking things like a method being moved from one class to another class and being able to include the history prior to the move