**Differences Between Centralised and Decentralised Version Control**

Version control is a system by which file or series of files is kept coherent and up to date when there are multiple people working on a file or said file is subject to regular change. Using version control allows someone to keep track of the changes made to a file and allows them to also see when the changes were made. In terms of programming files (.py etc.) it can assist with reversion to a stable build should a change cause failure in the code.

There are two main types of version control: Centralised – where the code is held in one location and is accessed by people from that location via a snapshot of the files, or Decentralised – where the code is mirrored across multiple locations via a cloning of the repository itself.

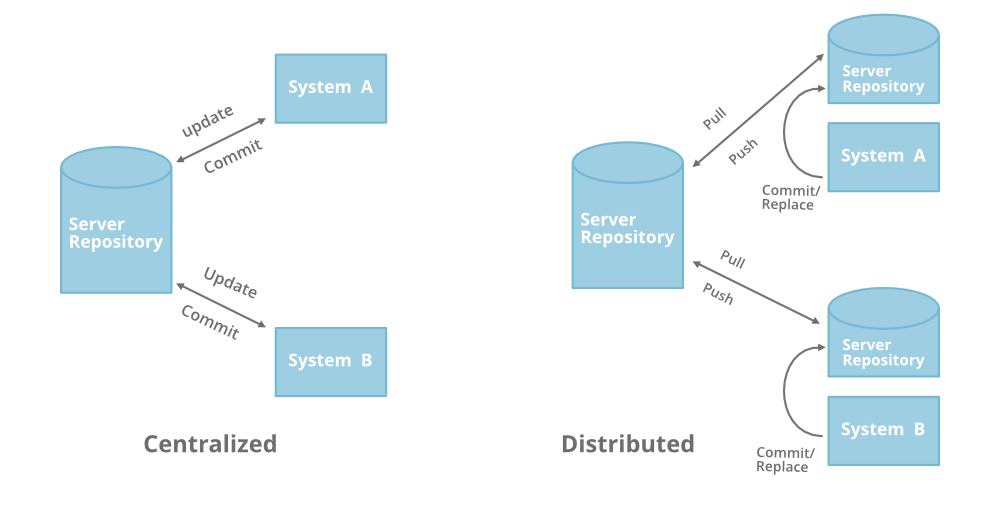


Fig 1. Visual representation of centralised and decentralised version control. (https://www.geeksforgeeks.org/centralized-vs-distributed-version-control-which-one-should-we-choose/)

There are many differences between the two types of version control. Primarily, with a centralised system there is a single point of failure. Namely if the repository hosting the files is offline, then no-one can update or request the files or submit updates/adjustments. With a decentralised method, should one person’s repository fail, then there will be other people with a copy of the files and folders who can share them with anyone who needs them.

In addition, decentralised systems require that a developer takes a copy (or clones) the entire repository and works on their own version before uploading (pushing) it to the designated ‘Central Repo’. This provides a lot of flexibility where required. (<https://scmquest.com/centralized-vs-distributed-version-control-systems/#:~:text=The%20concept%20of%20a%20centralized,repo%20on%20the%20server%20side>., 16/10/2020)

However, a centralised system is a lot easier to administrate and manage access control over users as the files are served from a single location. In addition, you do not have to merge different versions of the same files as everyone pushes to a single location.  
(<https://www.teamstudio.com/blog/distributed-vs-centralized-version-control-systems-for-lotus-notes>, 16/10/2020)

Decentralised version control systems are typically faster than centralised versions, especially when it comes to merging and creating branches etc. In addition, a decentralised system allows for a more tailored workflow between team members, again, due to the fact that each person has their own copy of the repository to work on interdependently of other team members.  
(<https://www.oshyn.com/blog/2012/06/version-control-systems-distributed-vs-centralized>, 16/10/2020)

Git and Mercurial are primary examples of decentralised VCS and Clearcase and Subversion are primary examples of centralised VCS.