What is GIT: GIT is a file change tracking tool.

Git is a distributed version control system that tracks changes in any set of computer files, usually used for coordinating work among programmers who are collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows

Programming languages used while writing GIT: Python, C, C++, Shell script, Perl, Tcl

Official link: https://git-scm.com/

Generations of version control system

Generation	Networking	Operations	Concurrency	Example Tool
First Generation	None	One file at a time	Locks	RCS, SCCS
Second Generation	Centralized	Multi-file	Merge before commit	CVS, Subversion
Third Generation	Distributed	Changesets	Commit before merge	Bazaar, Git

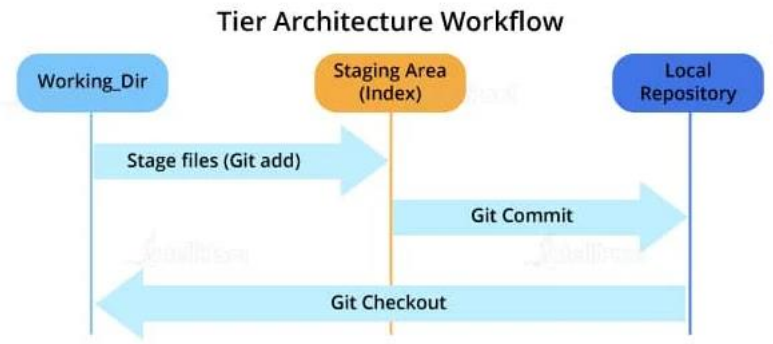
The Basic Terminology of Version Control System:

Working Directory: Where developers are required to create/modify files. Here version control is not applicable. Here we won't use the work like version-1, version-2 etc

Repository: Where we have to store files and metadata. Here version control is applicable. Here we can talk about versions like version-1, version-2 etc

Commit: The process of sending files from working directory to the repository.

Checkout: The process of sending files from repository to working directory.

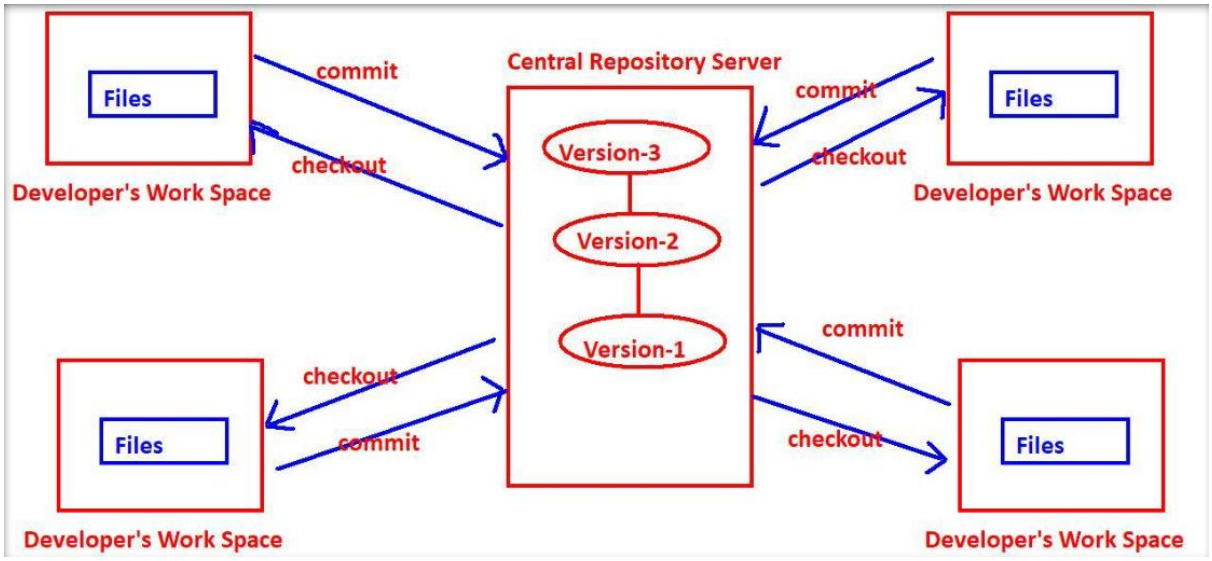


Role of Staging Area in GIT: Stagging area used for compression by which we can save 12 GB data [SVN] into 420MB

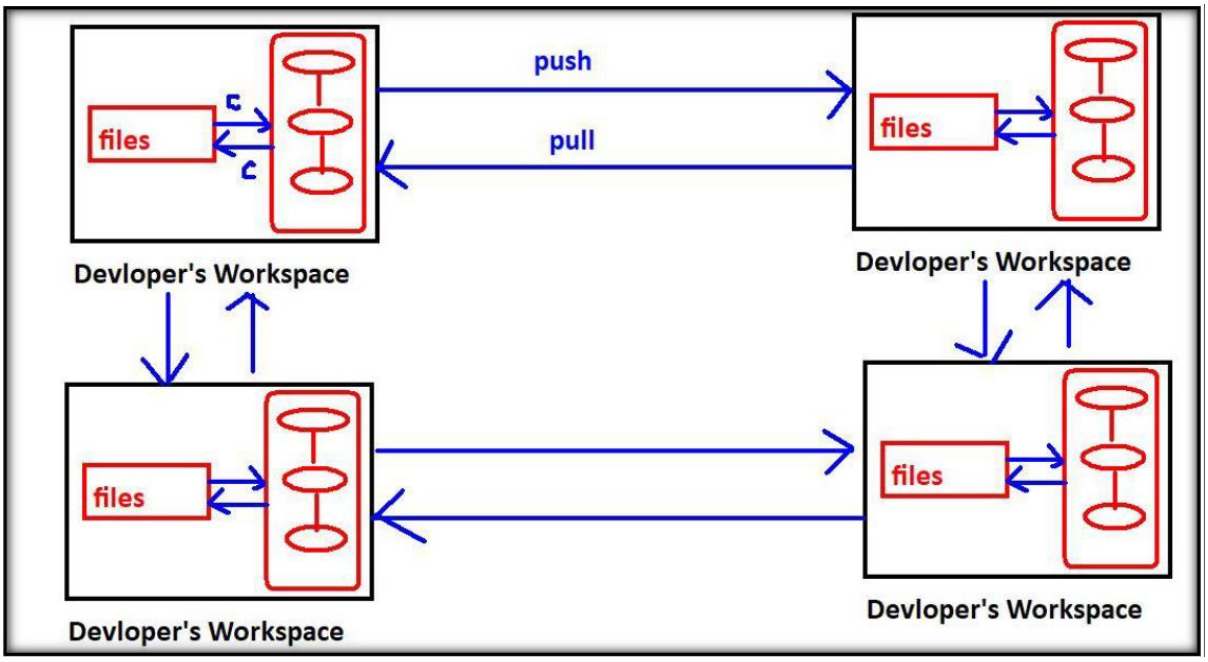
[Link]: https://www.devx.com/enterprise-zone/version-control-comparison-git-vs-dot-svn/

Centralized vs Distributed VCS

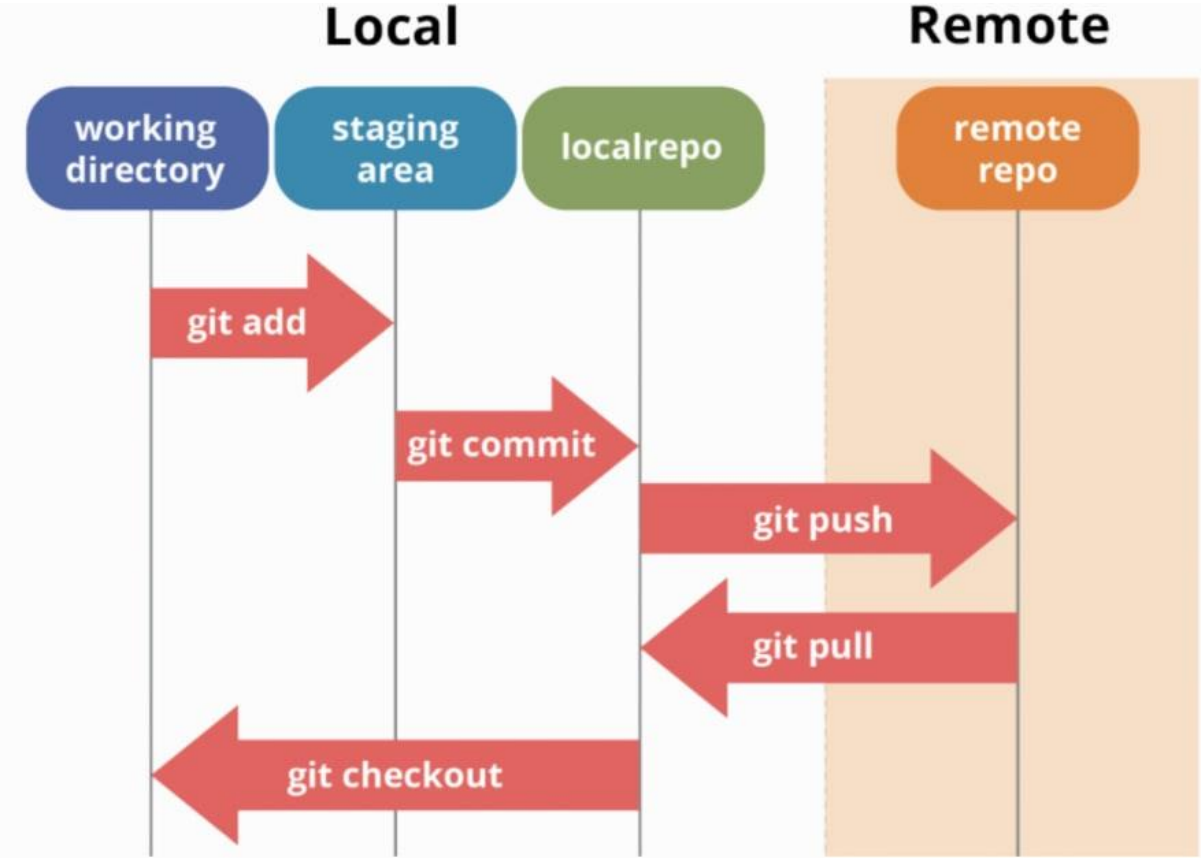
Centralized Version Control System: The name itself indicates that, this type contains only one central repository and every developer should be connected to that repository. The total project code will be stored in the central repository. If 4 developers are there, still we have only one repository.



Distributed Version Control Systems: The name itself indicates the repository is distributed and every developer's workspace contains a local copy of the repository. There is no question of central repository.



GIT workflow:



Support link:

Creating GitHub repository- https://docs.github.com/en/get-started/quickstart/create-a-repo

Creating GitHub account-https://docs.github.com/en/get-started/signing-up-for-github/signing-up-for-a-new-github-account

Types of repositories- https://www.simplilearn.com/tutorials/git-tutorial/what-is-a-git-repository

Username/email setup- https://git-scm.com/book/en/v2/Getting-Started-First-Time-Git-Setup

GIT to GitHub communication- git remote set-url origin https://<GITHUB_ACCESS_TOKEN>@github.com/<GITHUB_USERNAME>/<REPOSITORY_NAME>.git

Generating token for communication: https://docs.github.com/en/enterprise-server@3.6/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens