Version Control System

- Version Control System
- Distributed Version Control System
- Git Architecture
- Git Commands

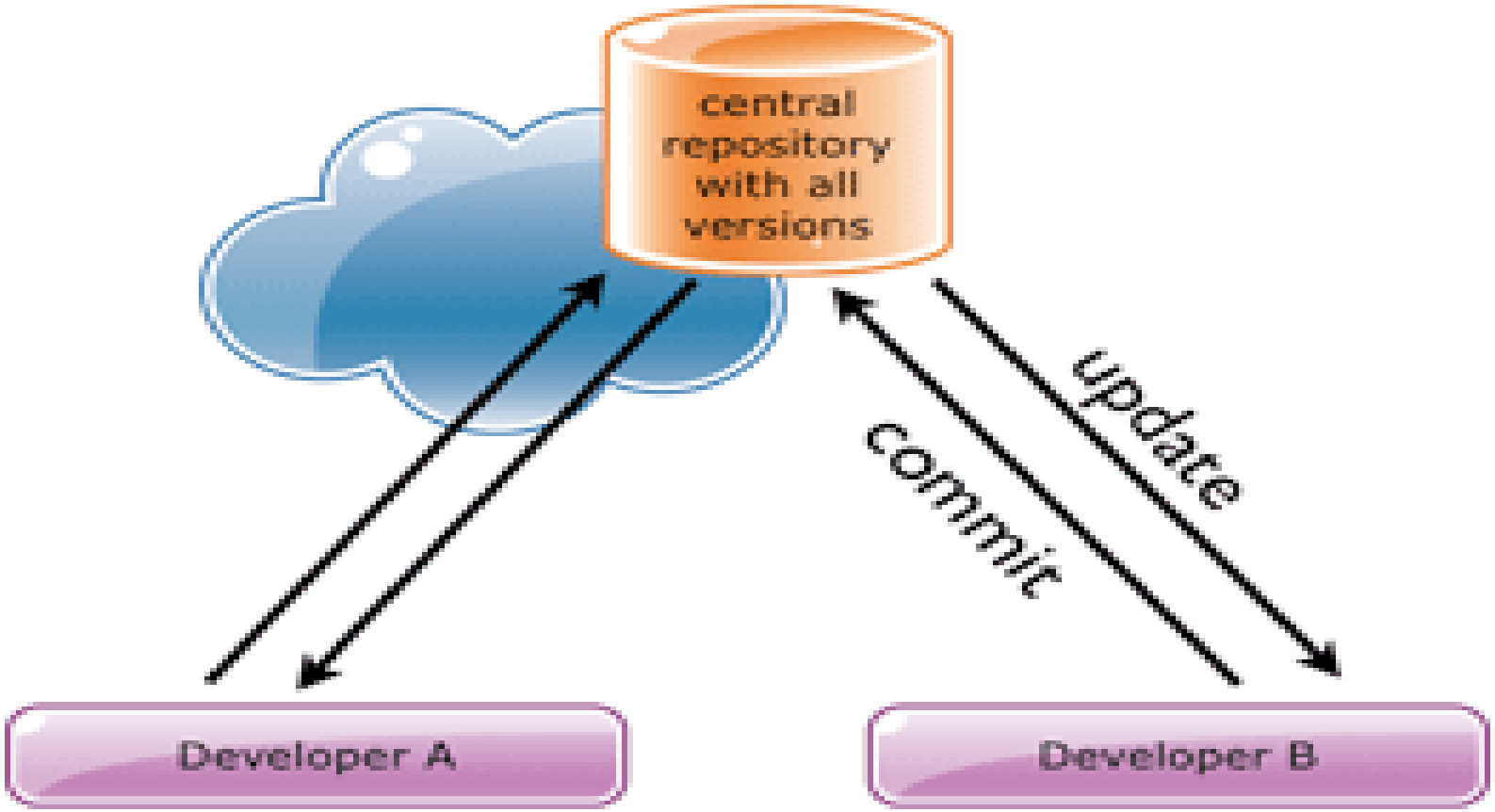
Problems While Developing & Managing Project

- Collaboration
- Storing Version
- Backup
- Analysis

What is the Solution?

Version Control System

Version Control System Architecture



Advantages of Version Controlling System

- It improves collaboration by storing the name of the developer who made the changes & reason for the change
- It stores every modification made into the software with different version number which helps in fetching the data from the version which is required
- After we perform modifications in the existing code we save it in server which helps in backing up the data. Even if the local system fails we can take the backup from server
- Version control system helps in analyse the project progress by defining the changes incorporated in the project like when, what, why & how much time is consumed to incorporate the changes.

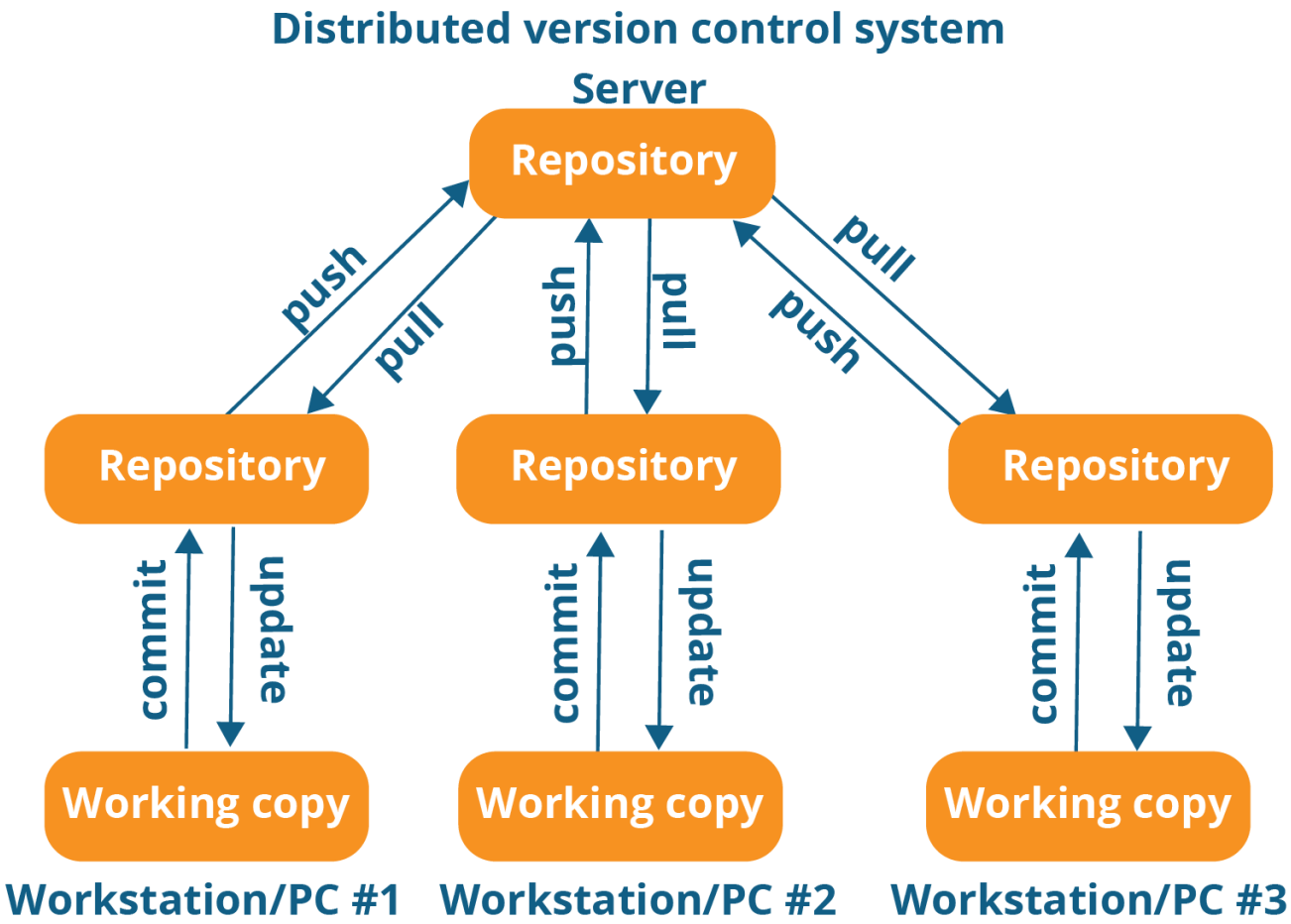
Drawback of Version Control System

- If the Central repository is crashed then the data saved in the central repository will also be get lost
- Whenever we perform any operations on the system & wants to save those changes in the repository then we need to have internet connectivity as it is located in remote server.
- While fetching or storing the data on centralized server the speed is low as it needs network connections.

What is the Solution?

Distributed Version Control System

Architecture Of Distributed Version Control System



Advantages of Distributed Version Controlling System

- It stores the data in both Local & Remote Repository which helps in recovering data in any of the repository fails or crashes
- We can save the files in local repository & when we get network connectivity we can push it to remote repository
- As we save the data in local repository hence fetching or storing will be faster

Git

- Git is a distributed version control system designed to ensure speed, integrity and support for non-linear workflows.
- It is used in both open source and commercial software development environments.
- Git is one of the most popular methods in the world for tracking changes.
- A key feature is that it enables developers to have independent branches, which turn enables, frictionless context switching, role-based codelines, feature-based workflows and disposable experimentation

Git

What is Git?

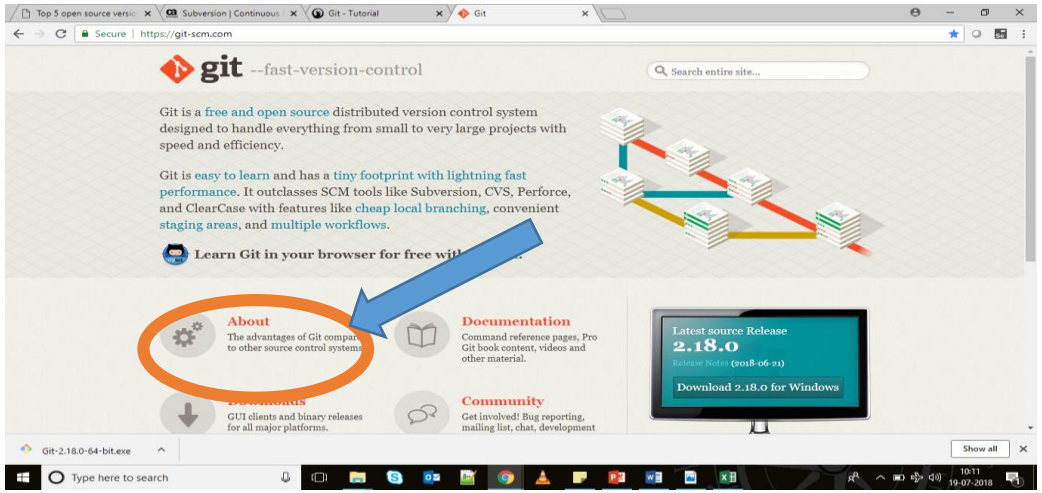
- Git is an Open Source Distributed Version Control System
- It is most Popular tool primary used for source code management in the software development
- It can also be used to keep track of changes in any set of files
- Git is created by Linus Torvald

Git Features

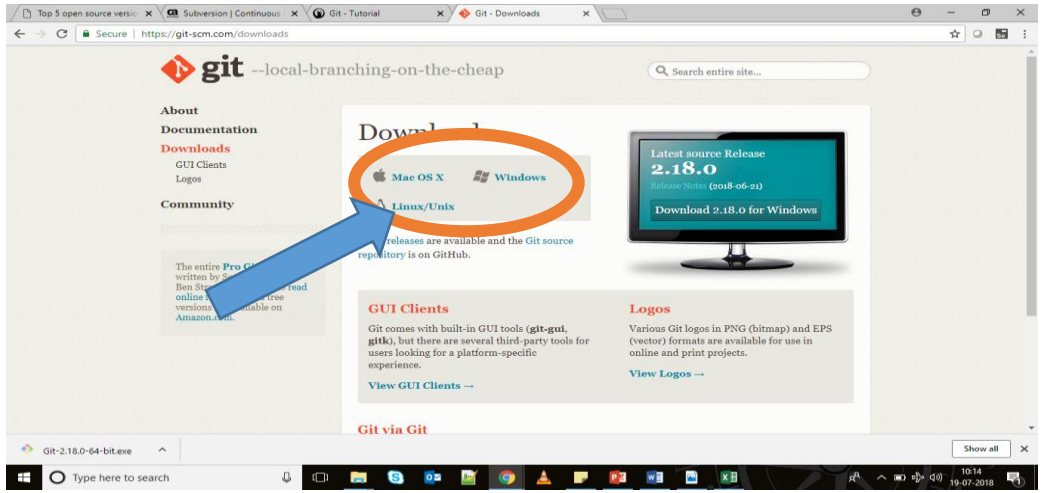
- Branching and Merging
- Small and Fast
- Distributed
- Data Assurance
- Staging Area
- Free and Open Source
- Trademark

Git Installation

- Step 1: Open the Git website & click Downloads link



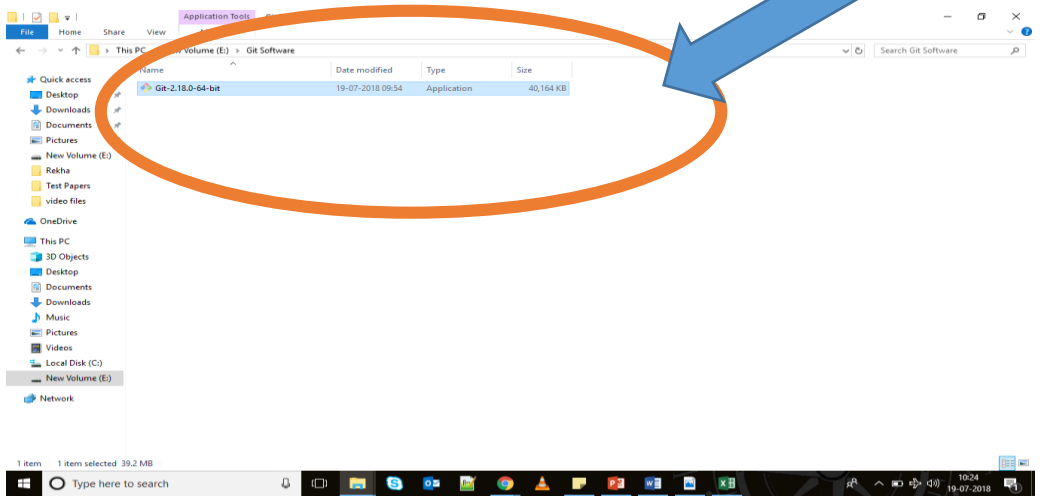
- Step 2: Click on appropriate link based on the system OS

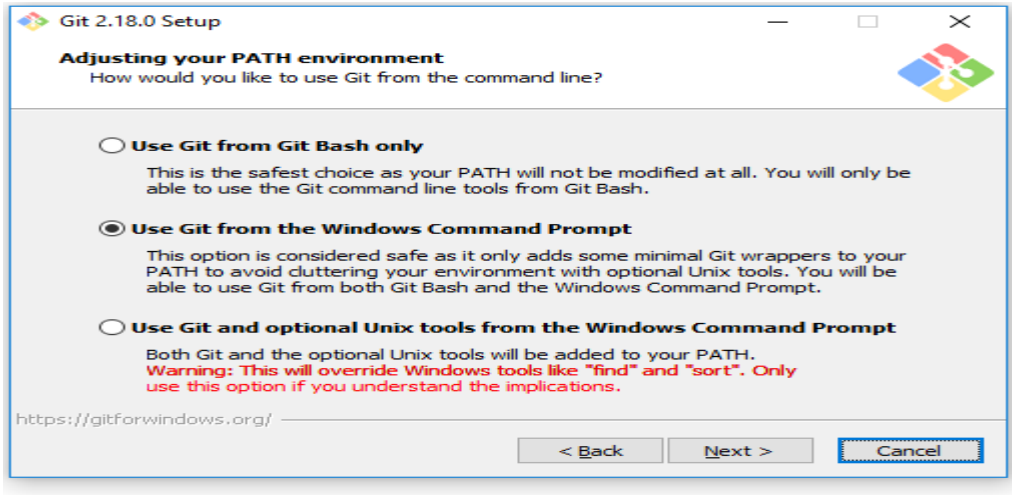


- Git .exe file will be get downloaded in the local system

Git Installation

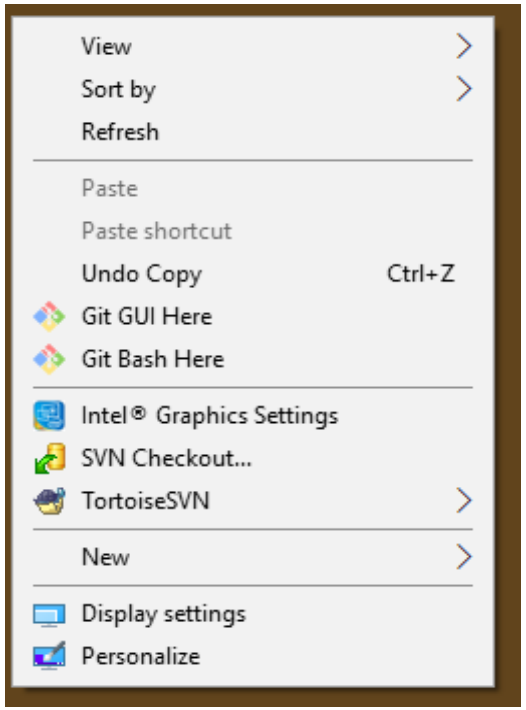
- Step 3: Run the .exe file for installation Step 4: Follow the simple step & select the appropriate option





Git Installation

- After installation you can see Git option upon right clicking



- It confirms the successful Git installation in the system

Git Repository Creation

Commands:

git –version : used to know the installed git version
mkdir Git_Repository : to create a directory named Git_Repository
git init Git_Repository : to create make Git_Repository as a repository

Note: git init creates local repository in the system which contains hidden .git folder

What is a Git Repository?

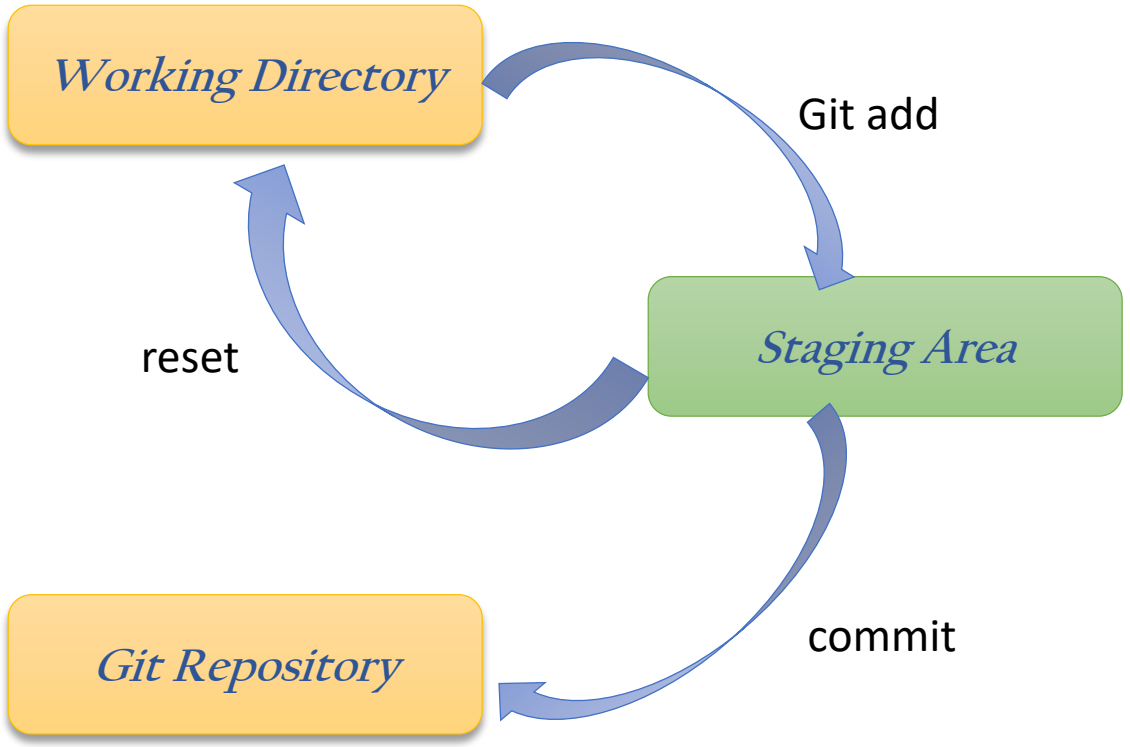
- A Git repository is a virtual storage of your project. It allows you to save versions of your code, which you can access when needed.

Types of Repository

- Local Repository: repository which is created in local system

- Remote Repository: repository which is present on remote location which is created with the help of github or gitlab etc.

Git Architecture



Git Commands

- git add <filename> : used to add the file into a staging area
- git status : to know the status of the files
- git commit : to add file into repository
- i : to insert the comment
- esc key: to come out of the comment line
- :wq : to save the comment message

Git Commands

- git add . : to add all files present in the directory to staging area
- git add *.txt : to add all files which are ending with txt present in the directory to staging area
- git log : to know the complete status of the repository
- Git reset : to remove the file from staging area to directory back

Note: reset can happen before commit only

- git diff
- git rm <filename> to remove the file from directory
- Git mv <file_from> <file_to> to rename a file

History

- To know the complete history of committed files in git use below command
- Git log

- To restrict to number of entries
- Git log –p -2

- To know statistics of committed files
- Git log –stat

- To display in one line
- Git log –pretty=oneline

Note: you can use short, full & fuller

Branching commands

- Creating branch
- git branch Manzoor

- Working on created branch
- git checkout Manzoor

Note: to switch between the branches use git checkout <branchname> which you want to switch

Merging Branches

- Know the destination branch where you want to merge the branch
- Point your git to the destination branch & perform below command

Commands

- git merge <branchname>

Note: while merging the branches if there is any difference between the files of merging branches then it will be get display as “fix conflicts and run git commit” message & difference in the file which has conflicts

Github

- GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere

- GitHub Flow is a lightweight, branch-based workflow that supports teams and projects where deployments are made regularly

Cloning Remote Repository

- Command:

- git clone <Remote_Repository_URL>

- To know which remote repository you cloned
- git remote

- To create repository in remote
- Git remote add <nameOfREpository> <URL_ofRemote>

Fetching and Pulling

- Git fetch <remote>
- Above command will fetch the remote file into local without merging the files

- Git pull <remote>
- Above command will fetch the remote file into local by merging the files

Adding an repository into git

- Add the repository in git to perform operations on remote repository
- Git remote add name <url of the repository>

Note: name created can be used to perform any operations on the remote repository

Pushing to Remote

- Git push <remote> <branch>
- Ex: git push origin master

- Git remote show origin

- To rename the repository
- Git remote rename <from> <to>

- To remove the repository
- Git remote remove <name>

Fork Repository

- To access the repository of some other person in github we use fork
- Search for a particular person → open the repository you want to copy → click on fork button
- Selected repository will be copied into your github account

- Introduction to Git
- Version Control System
- Distributed Version Control System
- Git Architecture
- Git Commands