By Nada Mostafa

What is version control?

- Version control, also known as source control, is the practice of tracking and managing changes to software code.
- Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake.

Version Control Terminologies

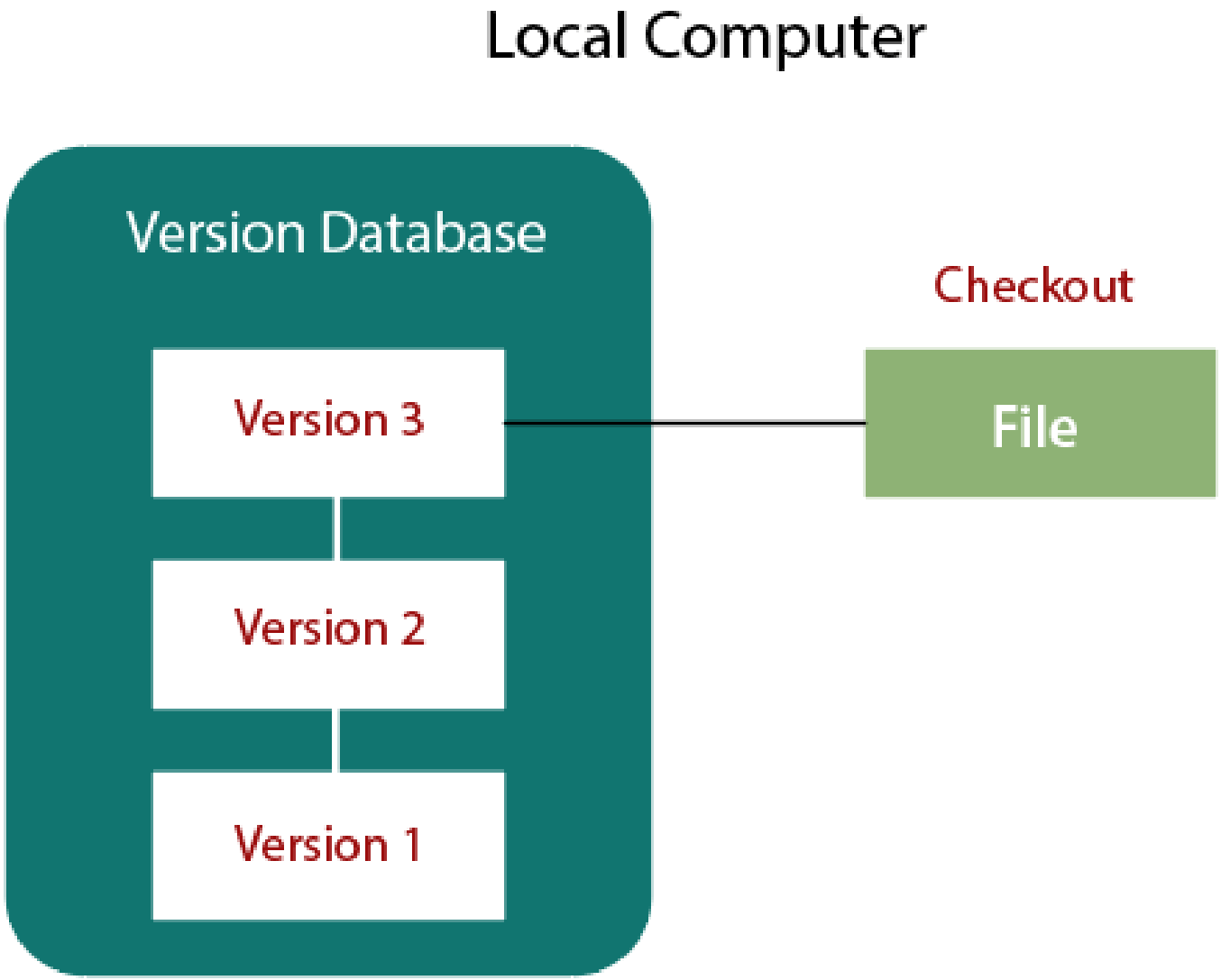
- Repository
- Branch
- Local Repo
- Remote Repo
- Commit (snapshot or save records(changes))
- Clone (from local or remote)
- Push (upload local changes to remote)
- Pull (you pull changes from remote repo to your local)
- Pull request (tell other about your changes to pull it from local to remote)

1- local version control

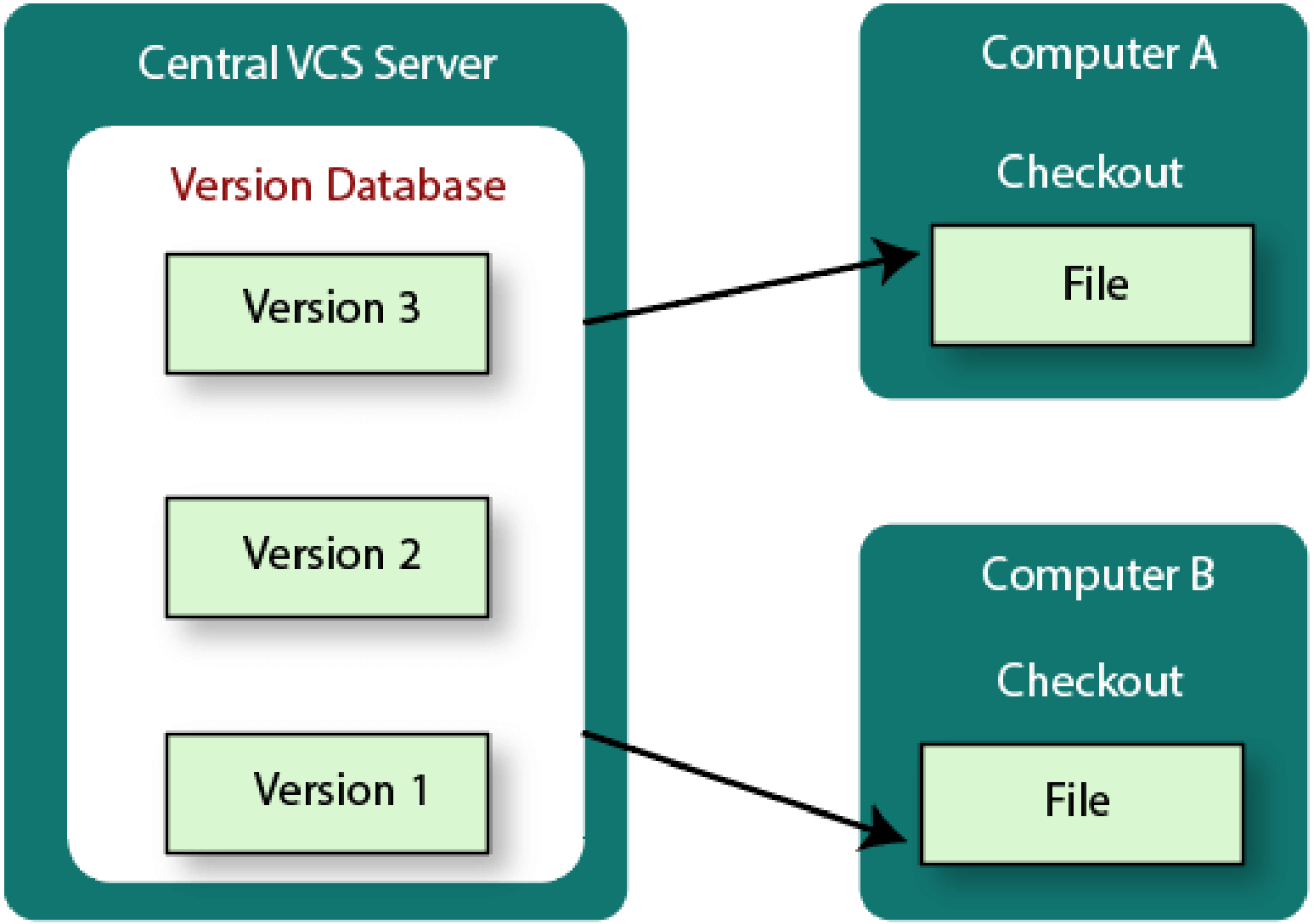
2- centralized version control

3- distributed version control

Local Version Control



Centralized version control systems are based on the idea that there is a single "central" copy of your project on a server, and programmers will "commit" their changes to this central copy.



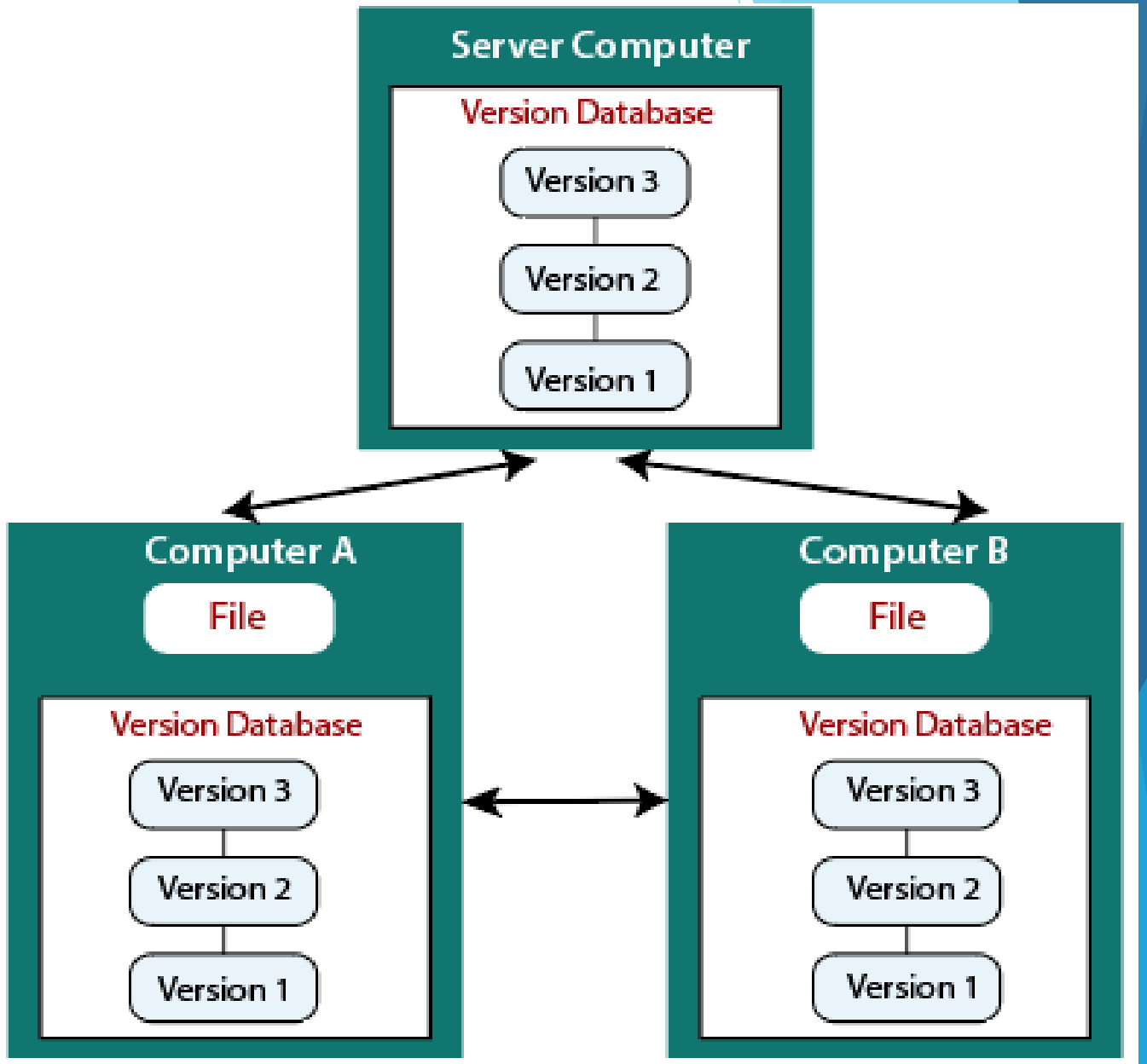
Disadvantages

- If the main server goes down, developers can’t save versioned changes.
- Need internet connection to commit the changes.

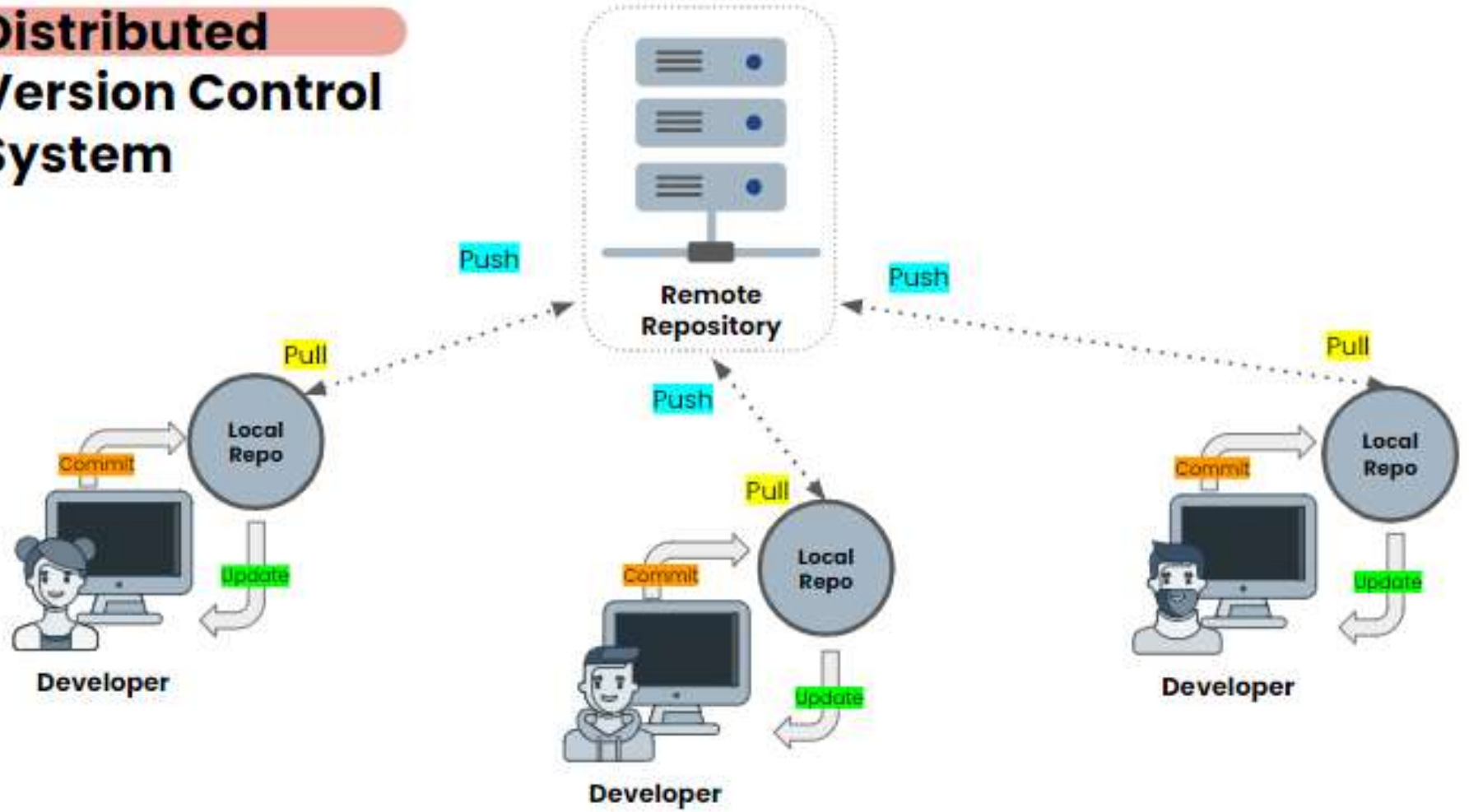
Distributed VCS

- These systems do not necessarily rely on a central server to store all the versions of a project's files. Instead, every developer "clones" a copy of a repository and has the full history of the project on their own hard drive.

- The act of getting new changes from a repository is usually called "pulling", and the act of moving your own changes to a repository is called "pushing.



Distributed Version Control System



Advantages

- Committing new changes can be done locally without anyone else seeing them. Once you have a group of changes ready, you can push all of them at once.

- Everything but pushing and pulling can be done without an internet connection. So you can work on a plane.
- Since each programmer has a full copy of the project repository, they can share changes with one or two other people at a time if they want to get some feedback before showing the changes to everyone.

Distributed VCS Examples

- Git

Git history

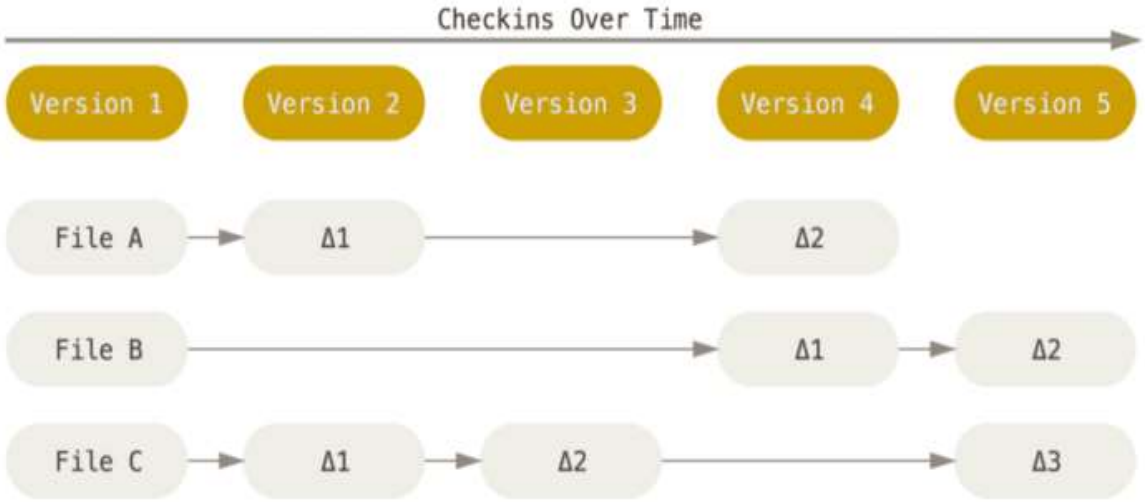
2002
Linux kernel project began using BitKeeper

2005
Linux kernel project stopped using BitKeeper

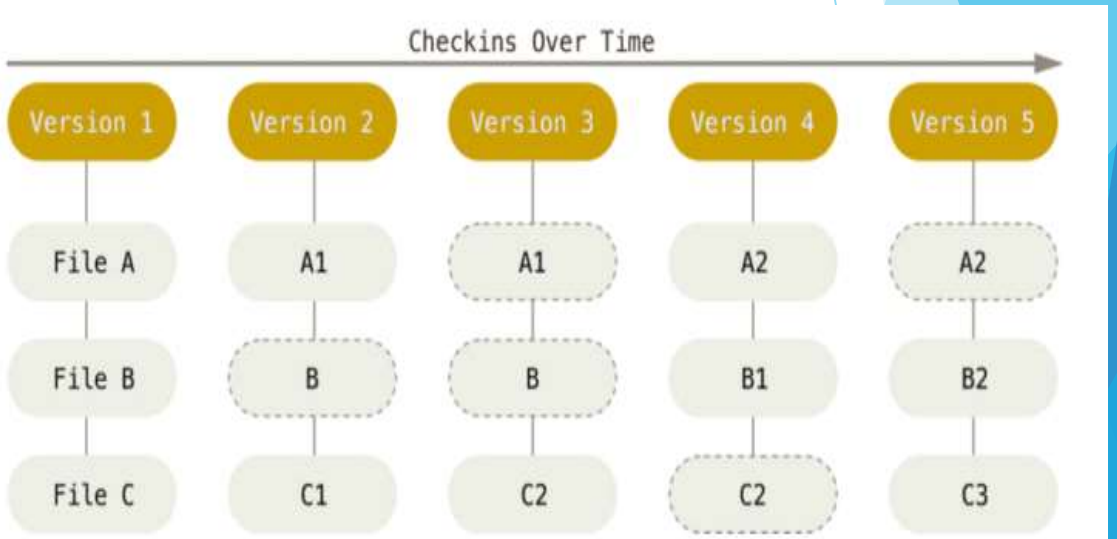
2005
Linus Torvalds started working on a new DVCS called Git

Advantages of git

others VC

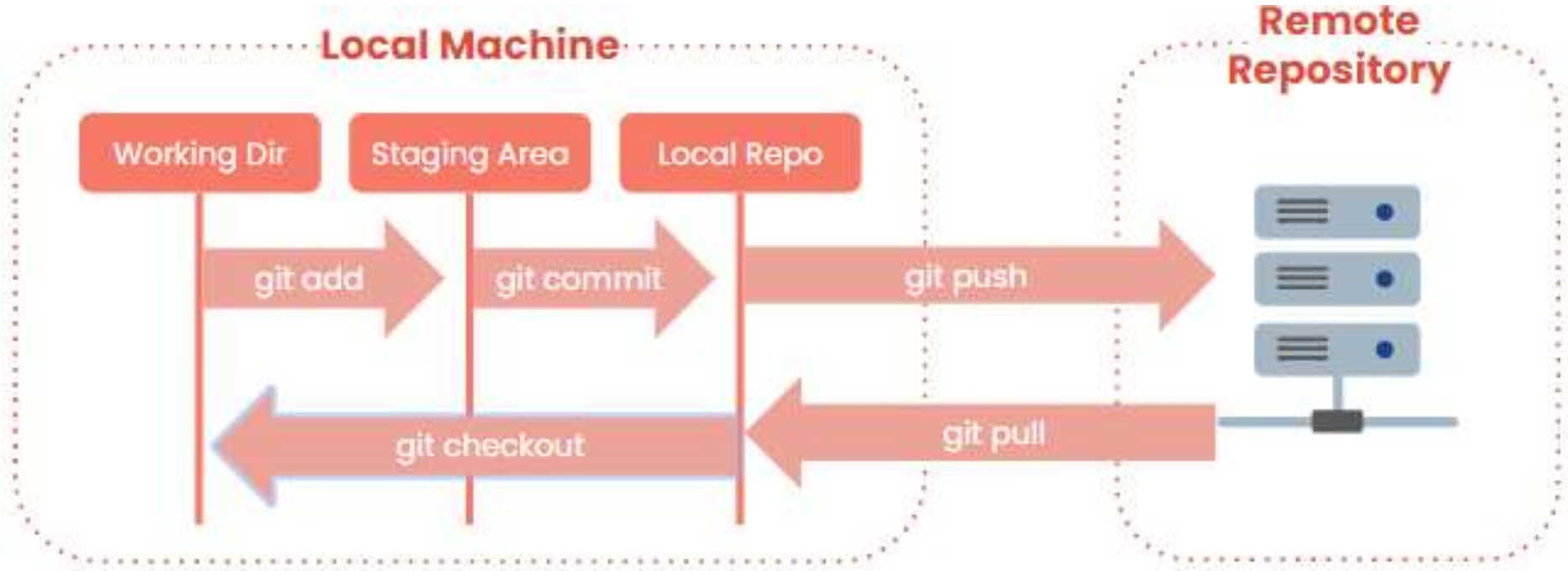


Git



- The Git Version Control System uses SHA-1 checksums on the contents of all change commits. In fact, the checksum is used as commit identifier and commonly referred to as "the SHA".
- Git's checksums include metadata about the commit including the author, date, and the previous commit's SHA.

- Git assures the integrity of the data being stored by using checksums as identifiers. If someone were to try to alter a commit or its meta data, it would change the SHA used to identify it. It would become a different commit.


Local Machine
Working Dir
Staging Area
Local Repo
git add
git commit
git push
git checkout
git pull
Remote
Repository

Git file states

- Un Tracked (U)

- Git has three main states that your files can reside in: Tracked
- **Modified** means that you have changed the file but have not committed it to your database yet.
- **Committed (unmodified)** means that the data is safely stored in your local database.
- **Staged** means that you have marked a modified file in its current version to go into your next commit snapshot.

First Time Git Setup

- The first thing you should do when you install Git is to set your user name and email address.
- This is important because every Git commit uses this information, and it's immutably baked into commits you start creating.
- **git config --global user.name "your name"**
- **git config --global user.email "your e-mail"**

Starting a repo

- mkdir firstdemo

  Make a directory
- cd firstdemo

  Change directory to the above directory
- git init

  Initialize an empty Git repository
- ls –a

  List all the files & dir and the hidden files & dir

Create a new file

- touch new.txt

  Create a new file called new.txt

- echo “hello git” << new.txt

  write in new.txt file

- git status

```
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        new.txt

nothing added to commit but untracked files present (use "git add" to track)
```

Add to Staging Area

- git add new.txt /git add * /.

  Staging the new.txt file

- git status

```
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   new.txt
```

Commit changes

- git commit -m “the first commit”

  Commit the changes with msg
- git status

```
$ git status
On branch master
nothing to commit, working tree clean
```

Add & Commit

- git commit -a -m “ commit”
- Add the changes & commit the changes in one line

- But , notice that this command doesn’t add new files
- It only works with the changes that made inside the files itself

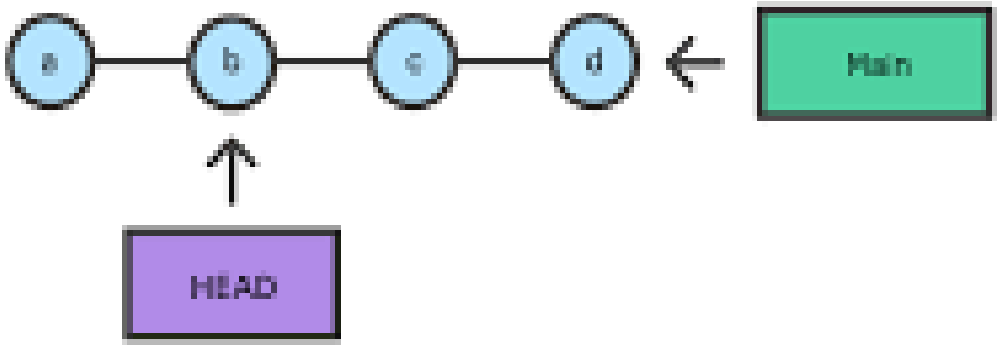
Git Logs

- git log

```
$ git log
commit bb2ffcbbc17d8eef90f64b9377c33e65536cda05 (HEAD -> master)
Author: nadamostafa <nadamostafa42127@gmail.com>
Date:   Wed Dec 7 01:55:54 2022 +0200

    inital commit
```

- git log --oneline


a
b
c
d
Main
HEAD

Git diff

- git diff

  Show the unstaged differences since the last commit (working tree & stage area)

- git diff --staged || --cached

  Show the staged differences since the last commit (stage area & repo)

-

Github

- is an Internet hosting service for software development and version control using Git
- Github is source for project and sources [gitlab, bitbucket]
- Github simplify using git

Clone a Remote Repo

- Https or ssh
- Git clone https://github.com/****/*****

To clone the entire repository to your local machine in a new directory

Git SSH Keys

- SSH keys come in pairs, public key that gets shared with services like GitHub and a private key that is stored only on your computer. If the keys match, you're granted access.

- Generate a new SSH key pairs
- ssh-keygen -t ed25519 -C "your e-mail"

- Copy the public key to your GitHub account
- cat ~/.ssh/id_ed25519.pub

Un doing things

- rm –rf .git
  - To remove local repo
- Git restore new.txt

  To restore from stage area (ignore changes in working tree)
- git commit --amend -m “your new msg”

  The above command will amend

  the added change to the last commit

Un staging changes

- git restore --staged new.txt

  To unstage the changes

- git reset HEAD script.py

  To unstage the changes

Why use version control?

- Developers contribute to the same project
- helps teams collaborate to fix issues & create new features
- You can control and backup your code (revert changes)
- Accelerates product delivery
- Keeping Track of All the Modifications Made to the Code
- Working on a new features without affecting the working code by branching

Lab

- Install Git.
- Create an account on GitHub.
- Create a new local repo and a remote repo on GitHub, then make a file contains your full name , then push it to the remote repo .
- Create a new remote repo on GitHub and clone it.

- Send me ➔ nm4378586@gmail.com