

# Git

# Objectives

- Version Control
- Git, Git repository
- Git vs Github
- Setup git environment
- Create a git repository ( adding file, adding folder)
- Checkout different version/ignore files...
- Git tools

# "FINAL".doc



FINAL.doc!



FINAL\_rev.2.doc



FINAL\_rev.6.COMMENTS.doc



FINAL\_rev.8.comments5.  
CORRECTIONS.doc



FINAL\_rev.18.comments7.  
corrections9.MORE.30.doc



FINAL\_rev.22.comments49.  
corrections.10.##\$%WHYDID  
ICOMETOGRADSCHOOL?????.doc



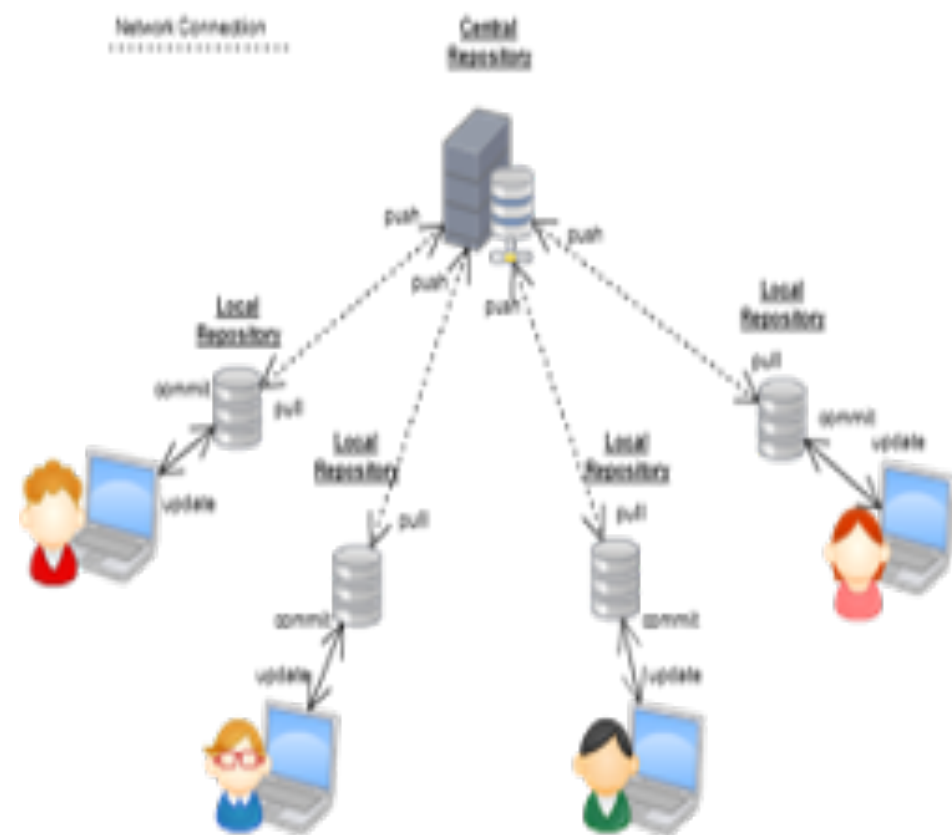
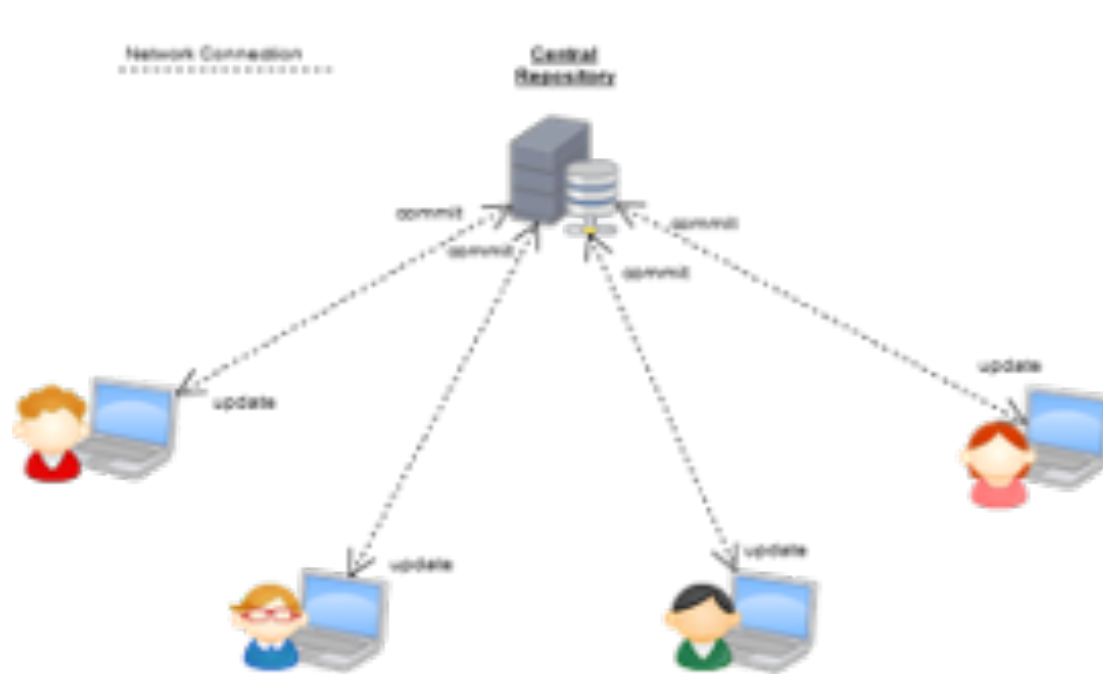
JORGES CHAM © 2012

# Why Version Control?

- Collaboration    Versioning    Rolling Back  
Understanding
- **Scenario** : Multiple students are doing a project together
- **Question**: Why not Google drive & One drive ????  
source code management vs file storage  
management

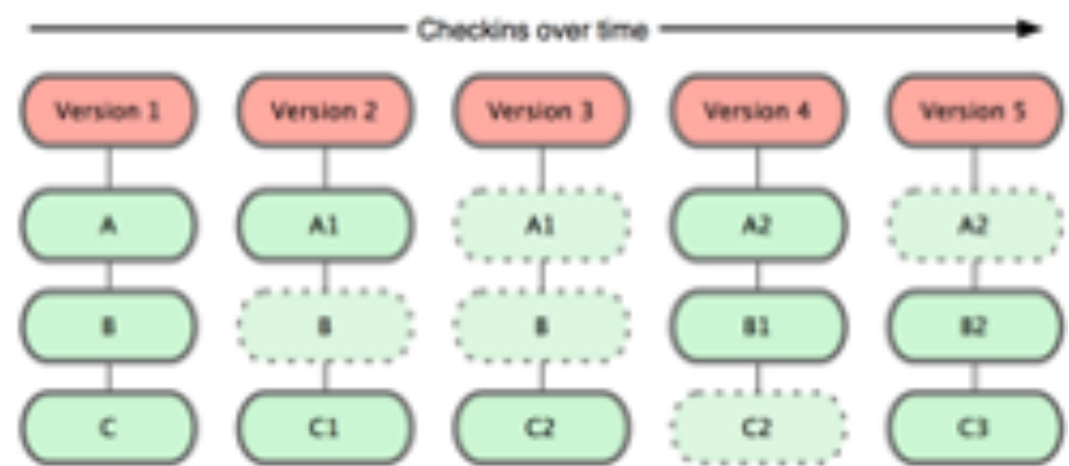
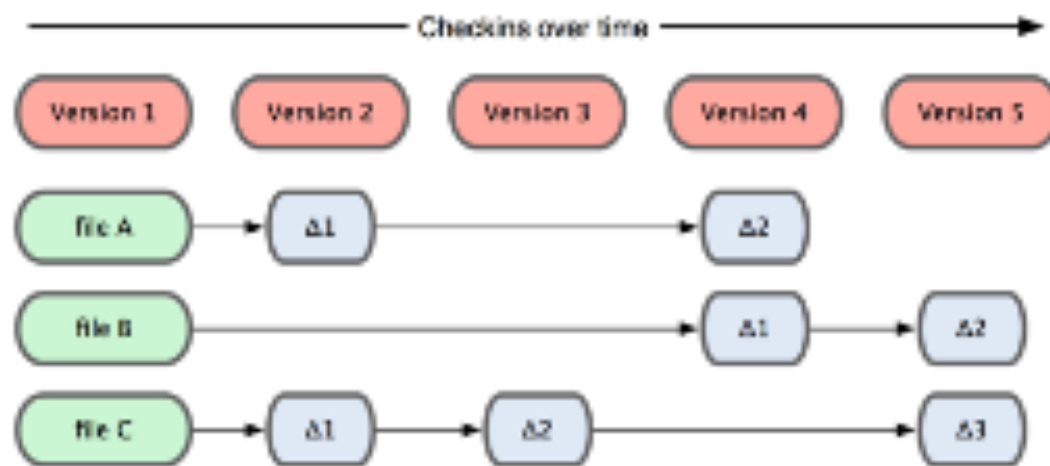
# Version Control Systems

- Some well-known version control systems are **CVS**, **Subversion**, **Mercurial**, and **Git**



# Why Git?

- Git :most commonly used software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development.
- Git has many advantages over earlier systems such as CVS and Subversion



# Git History

- Came out of Linux development community
- Linus Torvalds, **2005**
- Initial goals:
  - Speed
  - Support for non-linear development (thousands of parallel branches)
  - Fully distributed
  - Able to handle large projects like Linux efficiently
  - Global Information Tracker (acronym name)

# What is Github?

- GitHub was developed by Chris Wanstrath, P. J. Hyett, Tom Preston-Werner and Scott Chacon using Ruby on Rails, and started in February **2008**
- It is a subsidiary of **Microsoft**, which acquired the company in **2018** for \$7.5 billion
- GitHub.com is a site for online storage of Git repositories.
- You can get free space for open source projects or you can pay for private projects.



# Github

- **Question:** Do I have to use Github in order to use Git?
- Answer: No!
- you can use Git completely locally for your own purposes, or you or someone else could set up a server to share files, or you could share a repo with users on the same file system.

# Git

vs.

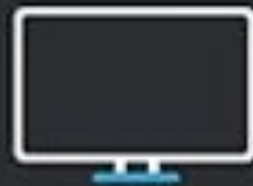
# GitHub



Git is installed and maintained on your local system (rather than in the cloud)



First developed in 2005



One thing that really sets Git apart is its branching model

Git is a high quality version control system

GitHub is designed as a Git repository hosting service



GitHub is exclusively cloud-based



You can share your code with others, giving them the power to make revisions or edits



GitHub is a cloud-based hosting service

# Download and install Git

- Here's the standard one: <http://git-scm.com/downloads>
- Git Repository is a place where Git can store versions of our files

```
$ cd ~/gitclass
```

```
$ mkdir demo
```

```
$ cd demo
```

```
$ git init
```

```
$ dir /ah or $ ls -a
```

(\* .git directory to store all the tracing info)

# Introduce yourself to Git

- Enter these lines (with appropriate changes):
  - `git config user.name "John Smith"`
  - `git config user.email jsmith@gmu.edu`
- `--global` param

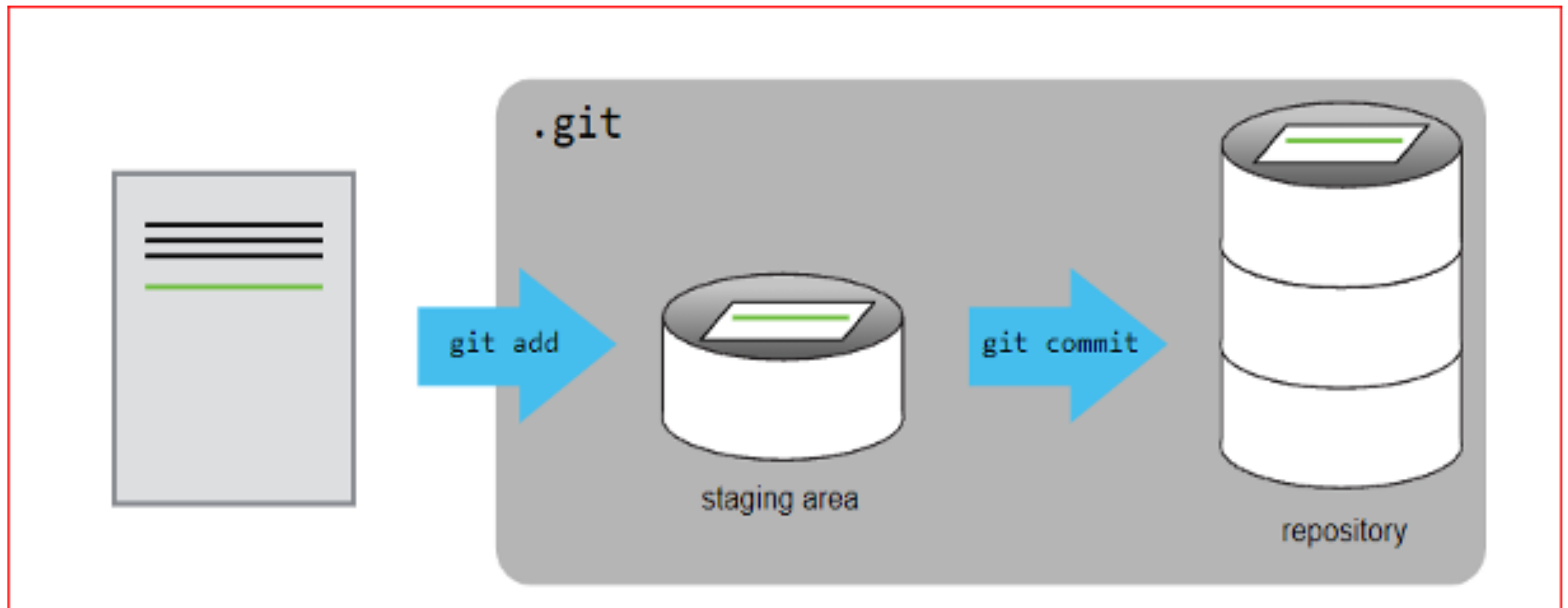
## command

## description

git clone <i>url [dir]</i>	copy a git repository so you can add to it
git add <i>files</i>	adds file contents to the staging area
git commit	records a snapshot of the staging area
git status	view the status of your files in the working directory and staging area
git diff	shows diff of what is staged and what is modified but unstaged
git help [ <i>command</i> ]	get help info about a particular command
git pull	fetch from a remote repo and try to merge into the current branch
git push	push your new branches and data to a remote repository

others: init, reset, branch, checkout, merge, log, tag

# Adding a file to git repository



# Adding a file to git repository

- Create a rose.txt file ,copy rose.txt to demo folder

```
$git checkout -b main
```

```
Git status $git status
```

```
$ git add rose.txt
```

```
$ git status
```

```
$ git commit -m "my first file checked in "
```

```
$ git log
```

# Continue...

- Change rose.txt file

```
$git status
```

```
$git diff
```

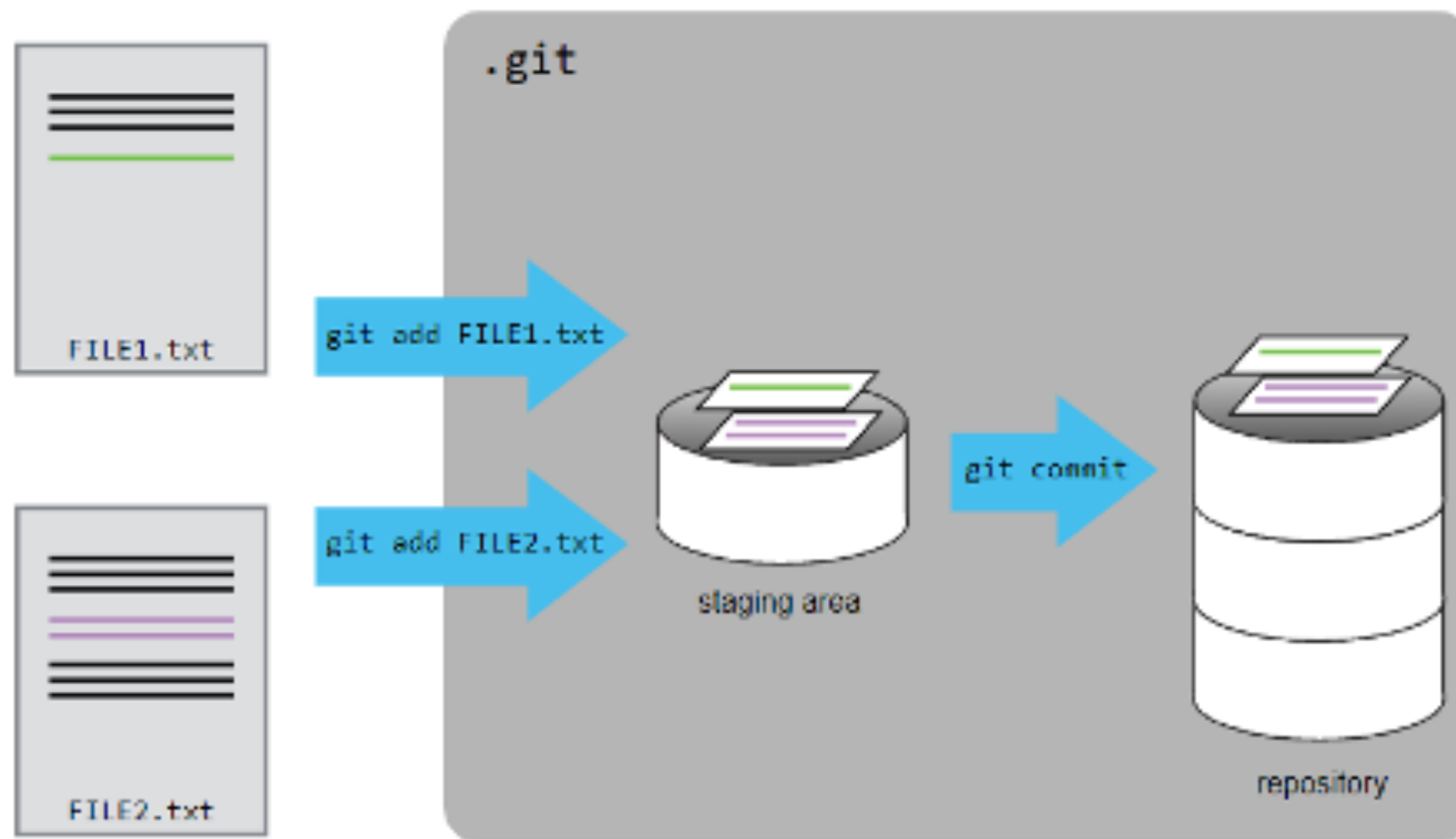
```
$git commit -a -m "new message...."
```

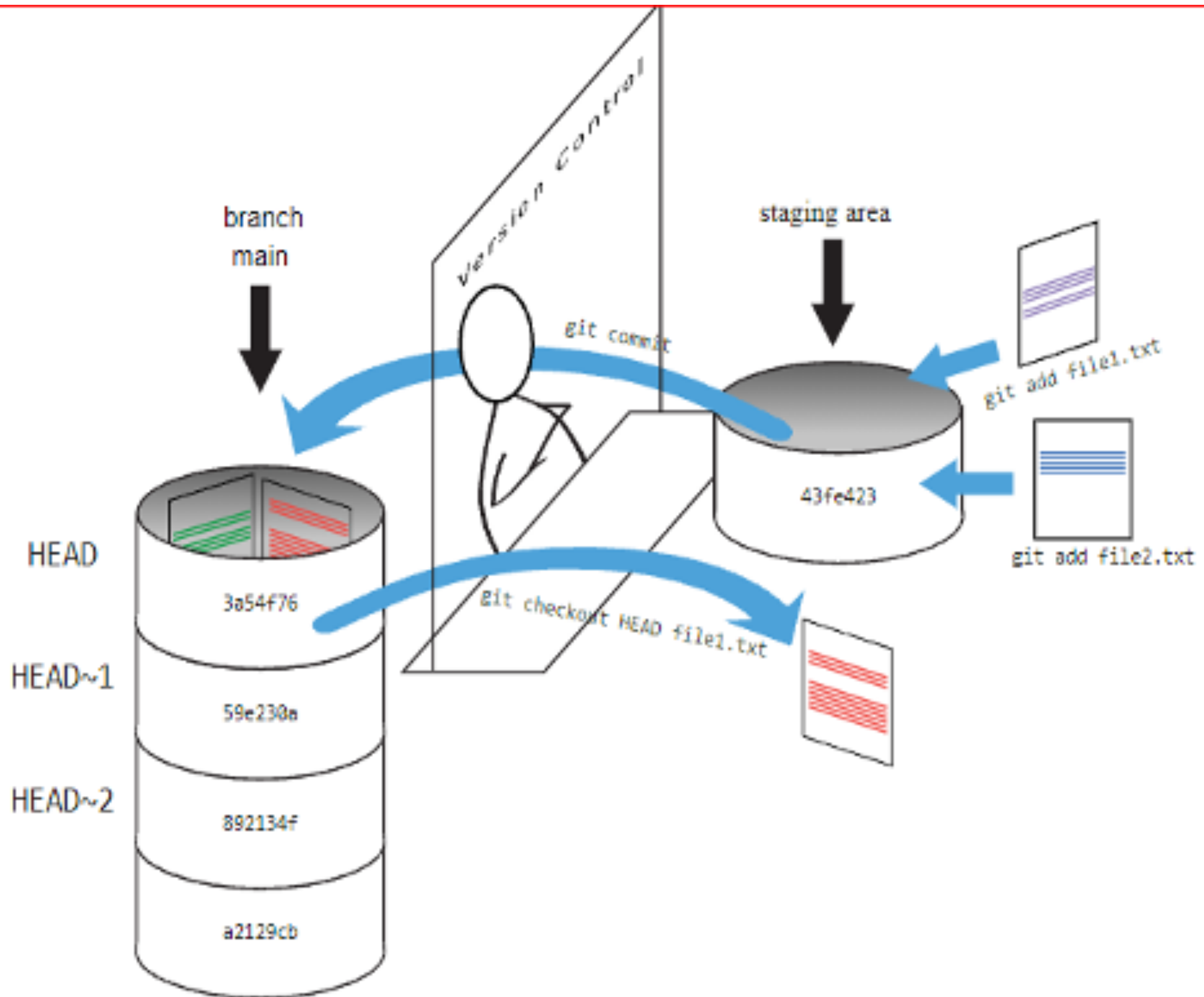


# Adding new folder

- `$mkdir poem`
- copy two text files to that **poem** folder
- `$git add poem`

# Adding Multiple files...





# Checking previous version

- `$ git diff HEAD~1 rose.txt`
- `$git checkout HEAD~1 rose.txt`
- `$git checkout HEAD rose.txt`

# Ignoring Things

- Configure Git to ignore specific files.
- Explain why ignoring files can be useful.
- Copy results folder with files
- Create .gitignore file
- Add

\*.dat

results/

# Git Tools

## Windows Git Clients:

- [SourceTree](#)
- [GitHub for Windows](#) .
- [Tortoise Git](#)

## Mac Git Clients:

- [GitUp](#)
- [GitBox](#)
- [Git-Xdev](#)