

# Git: Version Control System

# References:

- PPT: [https://github.com/shashwatx/git\\_101/blob/master/presentation.pptx](https://github.com/shashwatx/git_101/blob/master/presentation.pptx)
- Babel Git: <https://git.babel.es:4433/>
- SETUP: [https://github.com/shashwatx/git\\_101/blob/master/SETUP\\_GIT\\_DOC.MD](https://github.com/shashwatx/git_101/blob/master/SETUP_GIT_DOC.MD)
- WSL: <https://winaero.com/blog/enable-wsl-windows-10-fall-creators-update/>
- MobaXTerm: <https://mobaxterm.mobatek.net/download.html>
- git-create: [https://github.com/shashwatx/git\\_101/blob/master/scripts/git-create.sh](https://github.com/shashwatx/git_101/blob/master/scripts/git-create.sh)
- Gitlab Access Token:  
[https://docs.gitlab.com/ee/user/profile/personal\\_access\\_tokens.html](https://docs.gitlab.com/ee/user/profile/personal_access_tokens.html)

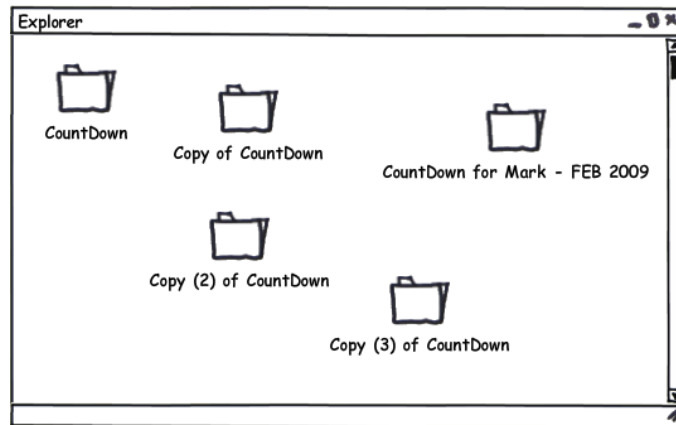
# 1.

## The Problem

How do you maintain  
and share code ?

“

- Copy? Email? Dropbox?
- Latest version?
- Roll back?
- Almost impossible to maintain if the number of people working in the project is large

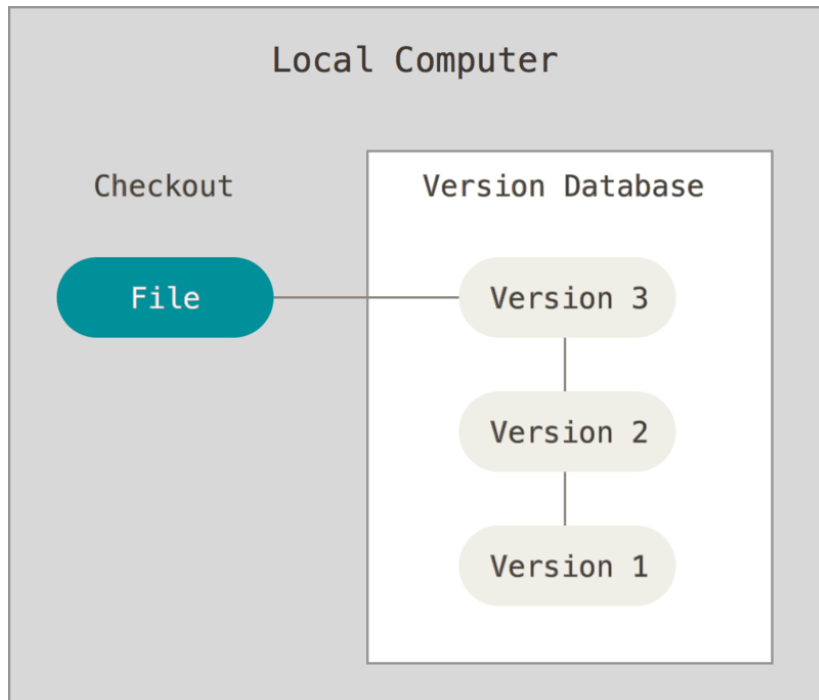


# 2.

## Version Control System

# Version Control: What is it?

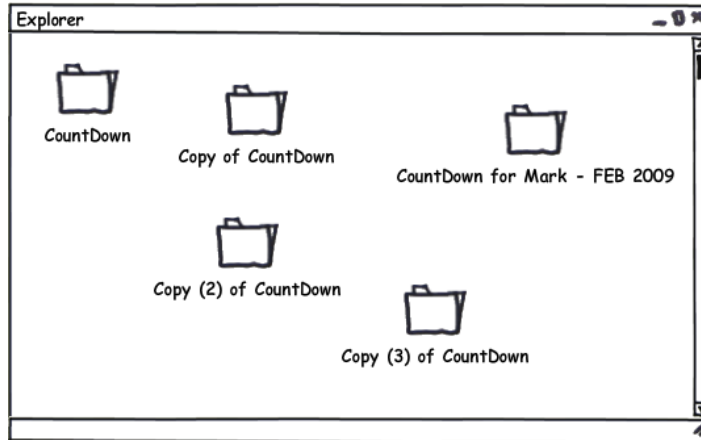
- ▶ A method for recalling versions of a codebase
- ▶ Keeping a record of changes
- ▶ Who did what and when in the system
- ▶ Save yourself when things inevitably go wrong



# Version Control: Why?

## Single Developer Workflow

- ▶ Back-ups of the project. **Fearlessly modify code**
- ▶ Create a “checkpoint” in the project at any stage:
- ▶ Tagging: Mark certain point in time
- ▶ Branching: Release versions and continue development



# Version Control: Why?

## Team Workflow

- Everything in “Individual”
- Allow multiple developer to work on the same codebase
- Merge changes across same files: handle conflicts
- Check who made which change: blame/praise



# Version Control: Types

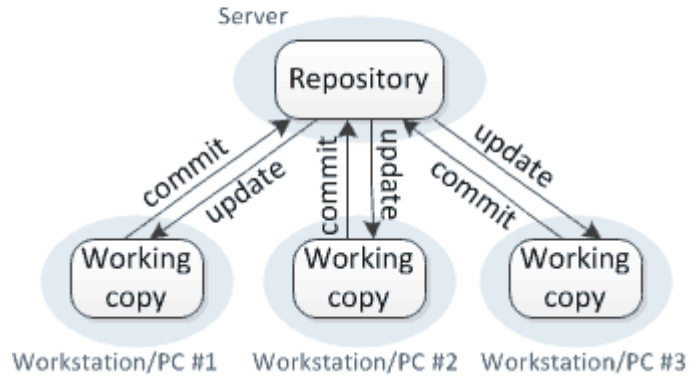
- Centralised VCS
- Distributed VCS

## Centralised VCS

- A **single** authoritative data source (repository)
- Check-outs and check-ins are done with reference to this central repository

# Centralised VCS

## Centralized version control



# Centralised VCS

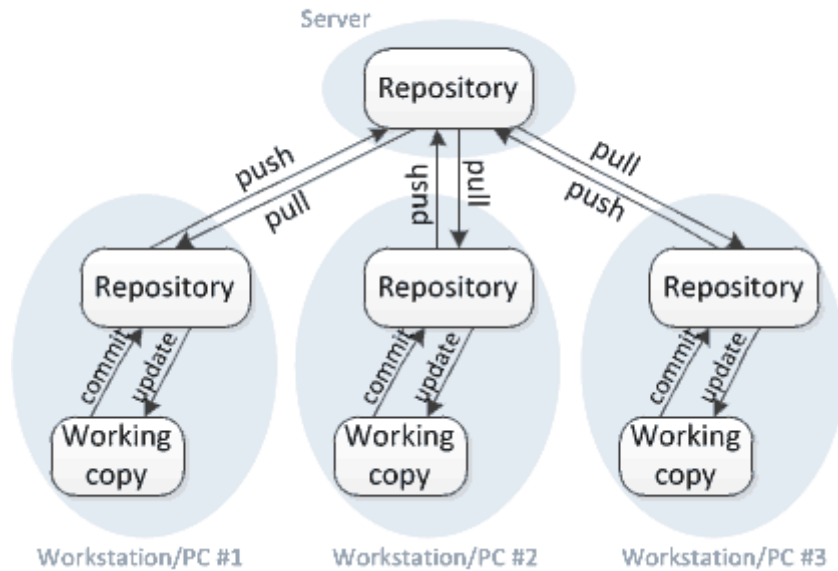
Examples:

- Concurrent Version System (CVS)
- Subversion (SVN)

- No single repository is authoritative
- Data can be checked in and out from any repository

Distributed  
VCS

## Distributed version control



Distributed  
VCS

## Examples

- Git
- Mercurial

Distributed  
VCS

# 3.

Git



--everything-is-local



- Free, open source
- Fully distributed
- Handle small files very effectively
- Tracks contents, not files
- Data = Snapshot
- Created by Linus Torvalds in less than 2 weeks



# git

# Git: Stages

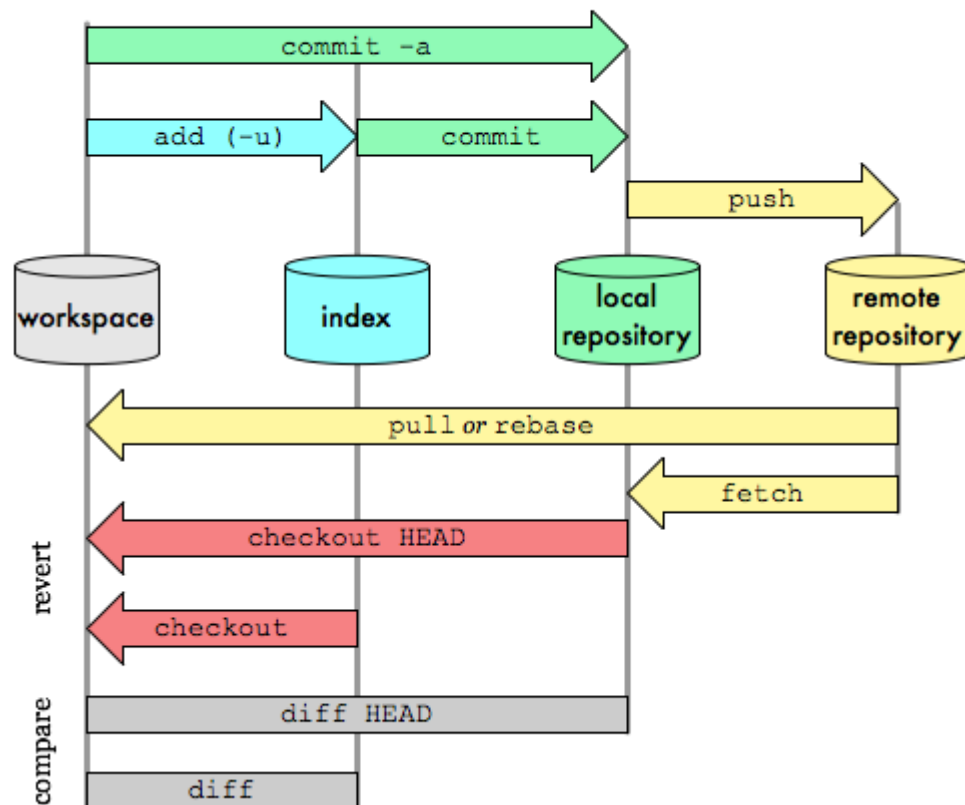
Four stages:

- Working directory
- Staging/Index directory
- Local repository
- Remote repository

# Git: Outline

## Git Data Transport Commands

<http://osteele.com>



# Git: Development

## Setup

- *git init*
- *git clone <remote-url>*

Save username and password.

*git config --global credential.helper store*

*git config --global push.default current*

# Git: Development

Show commit logs

- *git log*

# Git: Development

## Commit logs

```
git log
File Edit View Search Terminal Help
commit 115a041a1773a15968a059f8eef610a2b18d4cca
Author: mouril1 <rohit.das950@gmail.com>
Date: Sun May 28 15:19:21 2017 +0530

    Same as before

    Signed-off-by: mouril1 <rohit.das950@gmail.com>
commit 65e6907394f3b74ce854d283b29dcae4b28aedef
Author: mouril1 <rohit.das950@gmail.com>
Date: Sun May 28 15:14:12 2017 +0530

    Penultimate changes, I guess...

    Signed-off-by: mouril1 <rohit.das950@gmail.com>
commit b1c1027b8fac7d1c65e63da329838be172f496ce
Author: mouril1 <rohit.das950@gmail.com>
Date: Sun May 28 14:28:53 2017 +0530

    Still changing

    Signed-off-by: mouril1 <rohit.das950@gmail.com>
:|
```

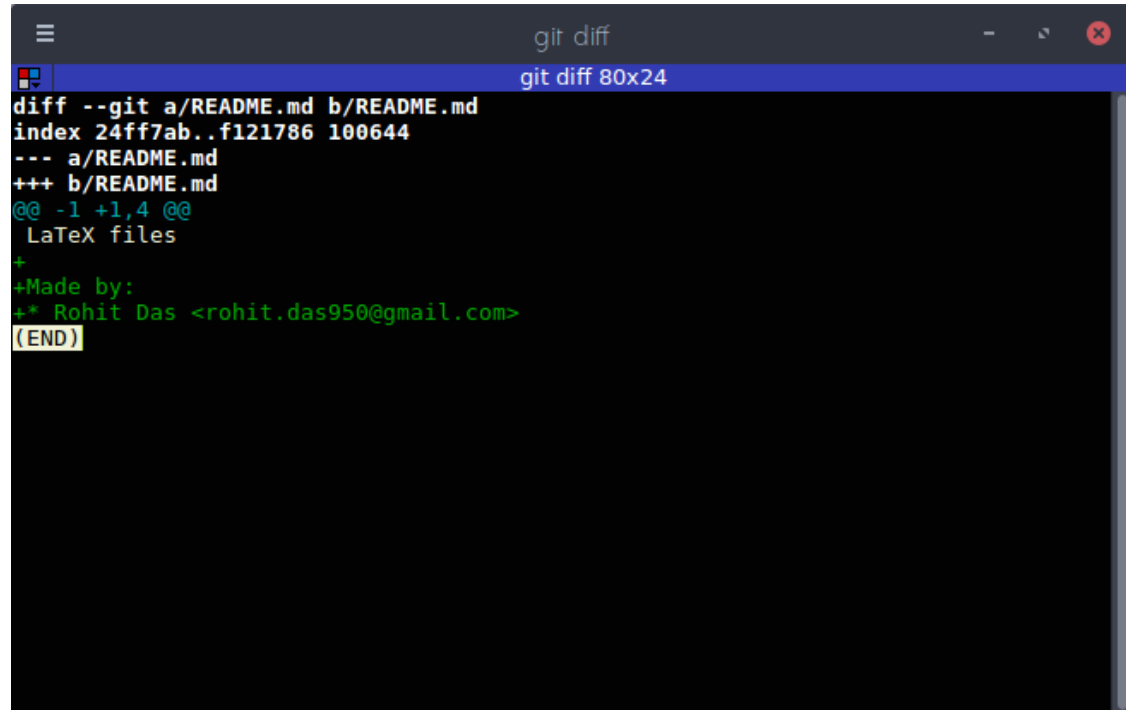
# Git: Development

View changes

- *git diff*

# Git: Development

## View changes



```
git diff
git diff 80x24
diff --git a/README.md b/README.md
index 24ff7ab..f121786 100644
--- a/README.md
+++ b/README.md
@@ -1,4 @@
 LaTeX files
+
+Made by:
+* Rohit Das <rohit.das950@gmail.com>
(END)
```



# Git: Development

Add changed files to the index

- *git add -u*

# Git: Development

Create “snapshots” of your codebase

- *git commit*

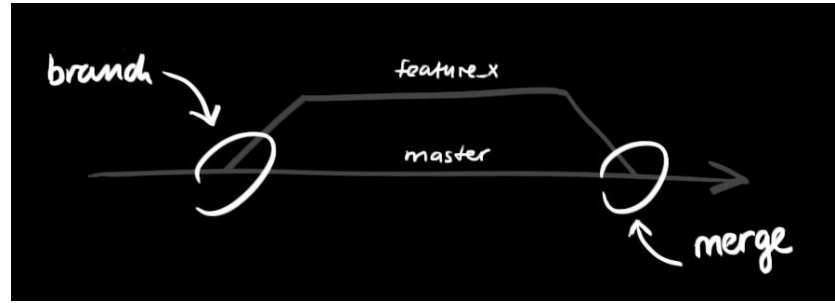
Push to remote

- *git push origin master*

# Git: Development

## Branches

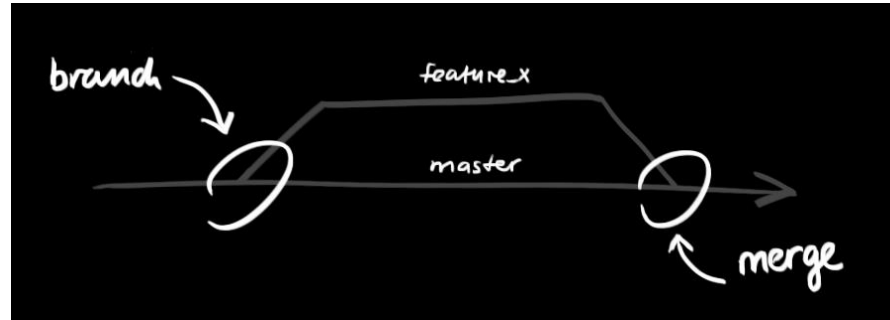
- `git checkout -b <branch-name>`



# Git: Development

Merge other branches

- `git merge`



## Result?

The development process of the Linux kernel is maintained using Git

The Linux kernel development process has:

- Over 2000 individual contributors per year
- Grows by nearly 300,000 lines per year

# THANK YOU!

Roberto Olmedo  
roberto.olmedo@babel.es

Shashwat Mishra  
shashwat.mishra@babel.es

