8 March 2020

VERSION CONTROL WITH GIT

Presented by Desmond Chin

ABOUT ME

DESMOND CHIN

Software Developer, Nuara Group

dsychin

WHAT IS VERSION CONTROL?

Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.

WHY USE VERSION CONTROL?

Storing Versions

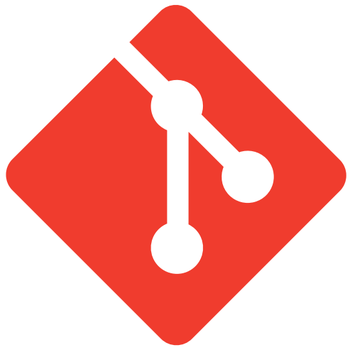
A complete history of changes is saved. You have access to who made what change when.

Restoring Versions

A version control system allows you to have 1 version you are working on with the ability to restore or compare any previous versions.

Collaboration

A version control system enables teams to work on a single project at the same time.



git

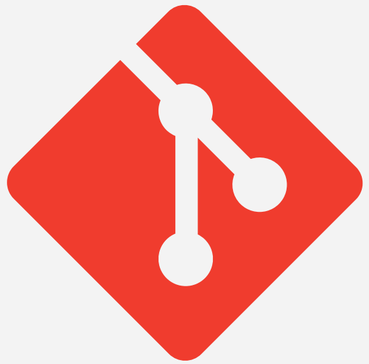
INTRODUCING GIT

WHY GIT?

1. Secure, flexible and performant
2. Distributed Version Control System
3. De-facto standard

GIT VS GITHUB

What's the difference?



git

Bitbucket

github

GitLab

Azure DevOps

Google Cloud

GIT TERMINOLOGY

REPOSITORIES

A directory of all content needed by your project

COMMIT

A single point in the Git history.
The action of storing a new snapshot of the project's state in the Git history.

CHECK OUT

The action of updating all or part of the working tree.

WORKING TREE

The tree of actual checked out files.

BRANCH

A branch represents an independent line of development.

HEAD

The tree of actual checked out files.

WAYS TO USE GIT

Command Line Interface

The original way to use Git.
Allows for more complex usage and automation.
Learn once and use everywhere.

Third-party GUI

There are third-party software that provides a graphical user interface for users who prefer it.
e.g. SourceTree, GItHub for Desktop, GItKraken, etc.

Code Editor Feature or Plugin

Most code editors come with built-in integration with Git.
e.g. Visual Studio, Visual Studio Code, Atom, Sublime, etc.

INSTALLING GIT

https://git-scm.com/downloads



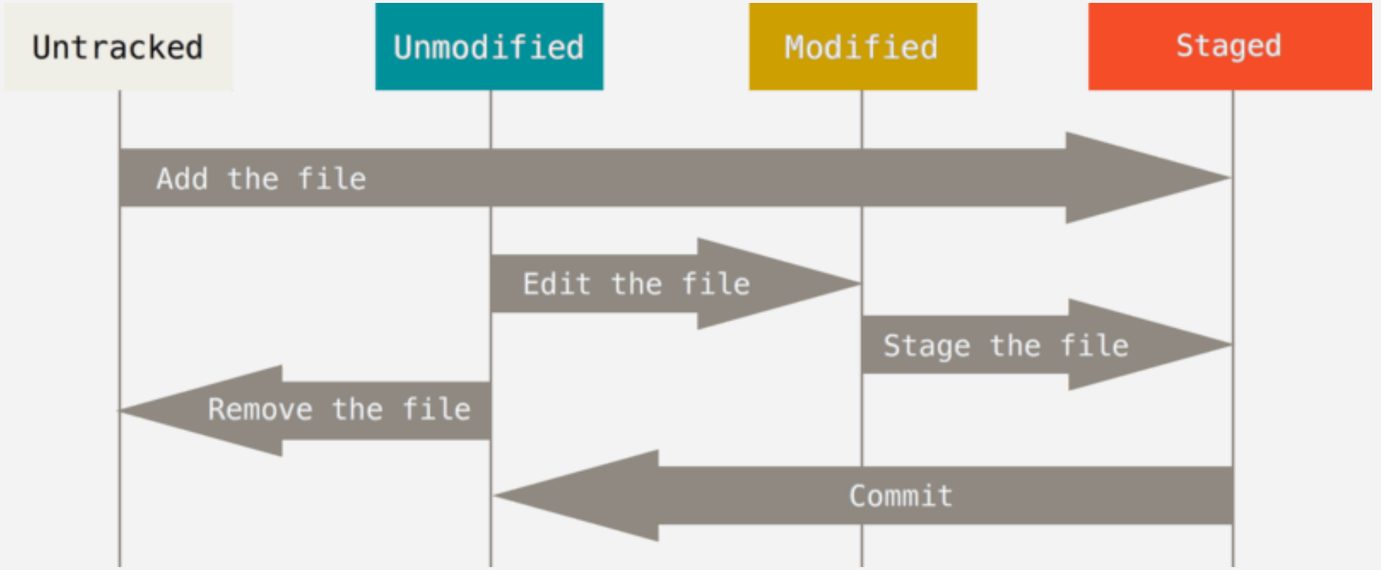
DEMONSTRATION

CREATING A GIT REPOSITORY

- Creating a new respository locally
- Cloning an existing repository

MAKING CHANGES

- Tracking new files
- Check status
- Staging modified files
- Ignoring files
- Comparing files
- Working on a different branch



Source: https://git-scm.com/book/en/v2/Git-Basics-Recording-Changes-to-the-Repository

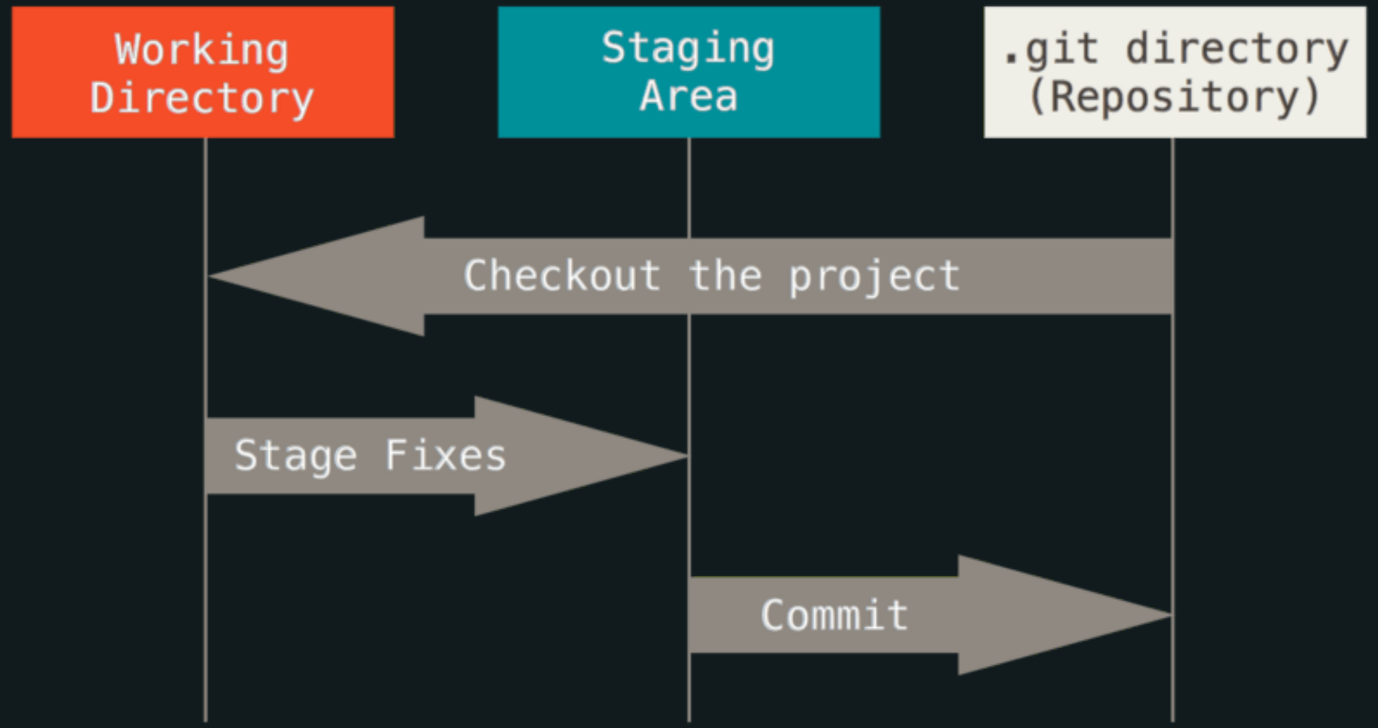
GOING BACK

- Who did this?
- Viewing previous snapshots
- Restarting from the last commit
- Revert to a previous commit

BACKING UP YOUR WORK

SYNC WITH GITHUB

HOW DOES GIT WORK?



Source: https://git-scm.com/book/en/v2/Getting-Started-What-is-Git%3F

GIT BEST PRACTICES

- Commit early and commit often
- Write meaningful commit messages
- Don't commit generated files
- Avoid altering published history

WHAT'S NEXT?

Resources

dsychin

- https://try.github.io
- https://www.atlassian.com/git
- https://git-scm.com/book/
- https://github.com/dictcp/awesome-git

This presentation slides are available at
https://github.com/dsychin/iwd2020-git-talk