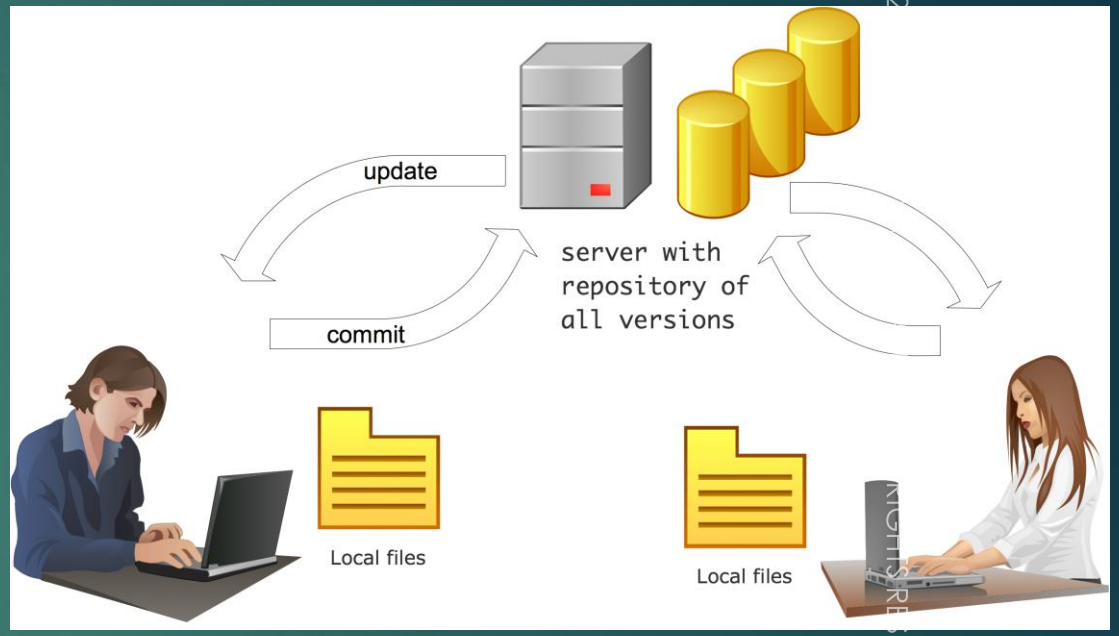
Software Development Tools

Agenda

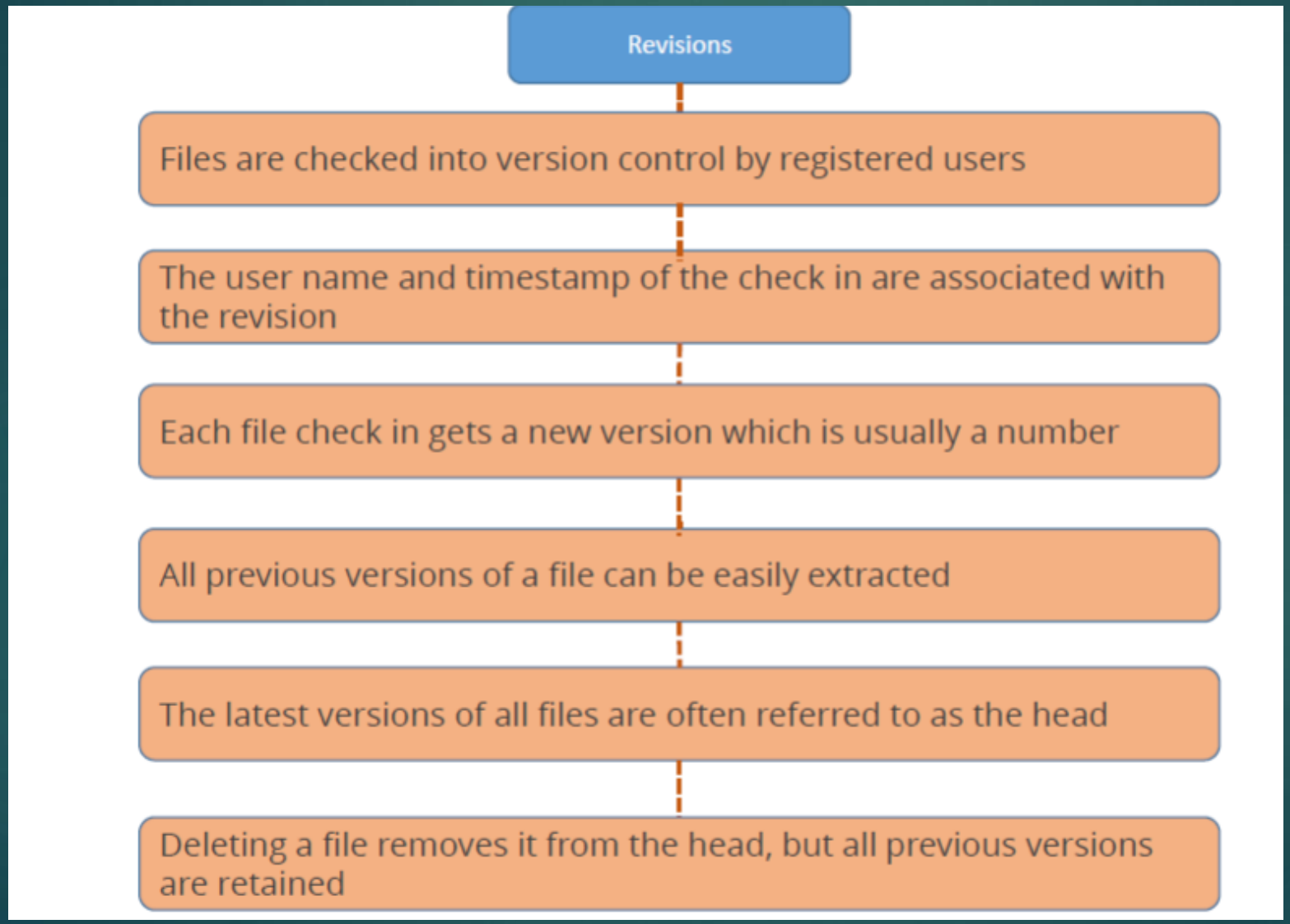
- Source Control Systems Introduction
- Types of Source Control Systems
- Overview of Subversion
- Using Git
- Working with GitHub
- Understanding & Using Build Tools

Source Control Systems

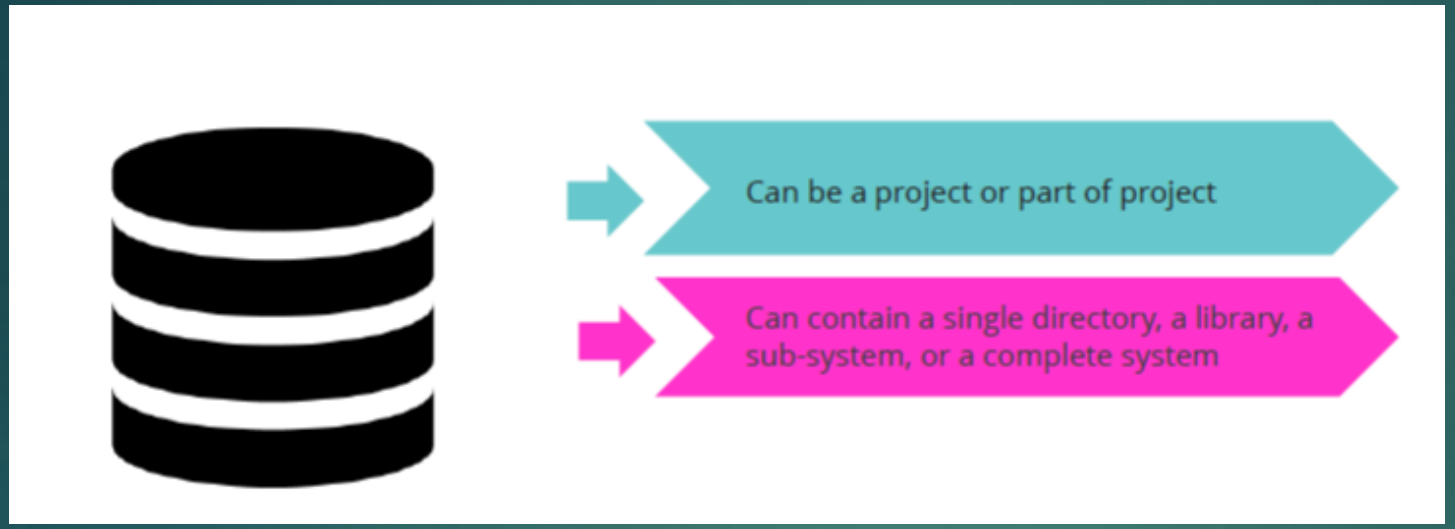
- Manages Changes to documents so that their state is consistent.
- Version Control Systems or Revision Control Systems
- Best suited for storing changing text documents and they are usually associated with Source Code



Revisions

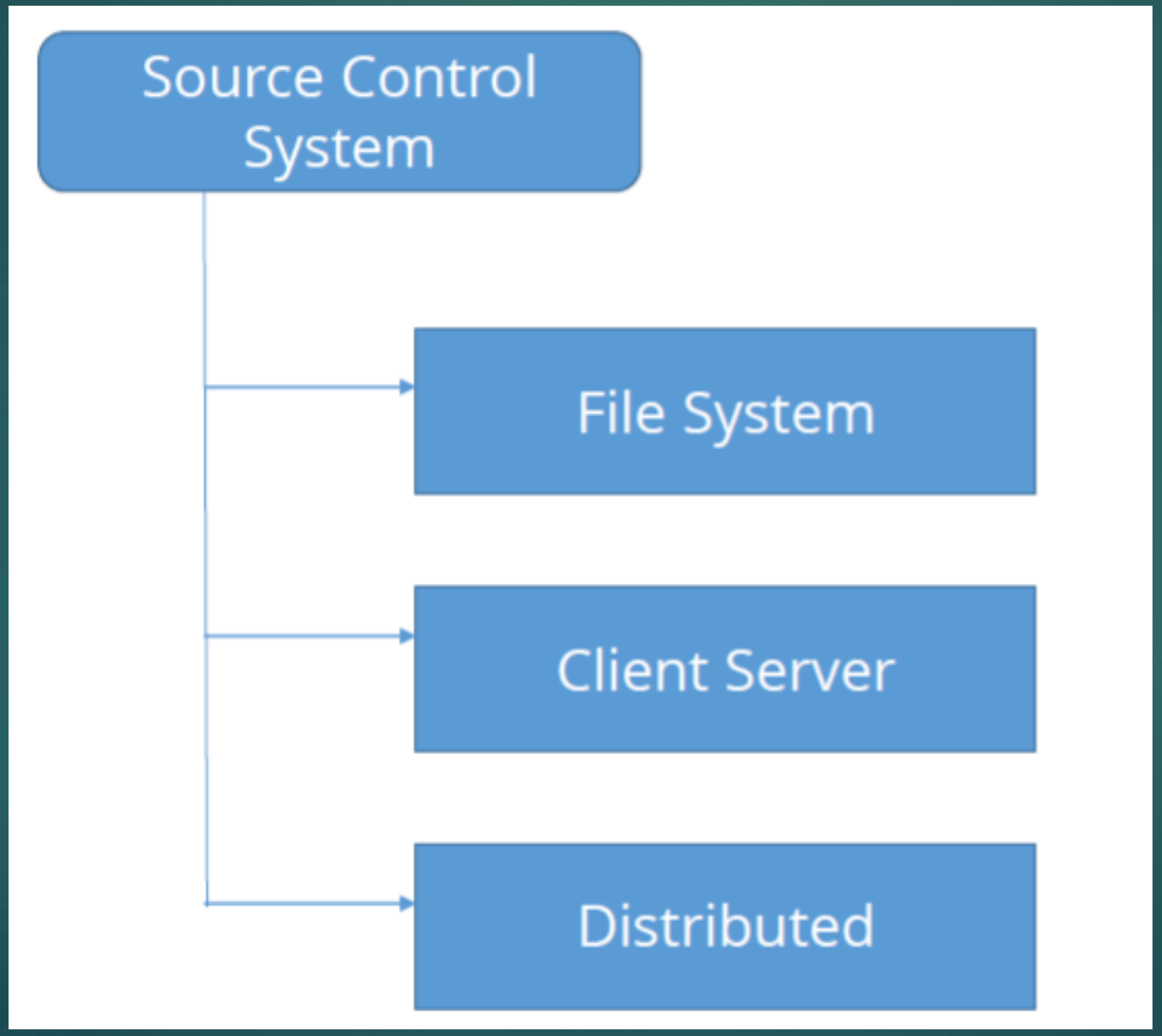


Repositories

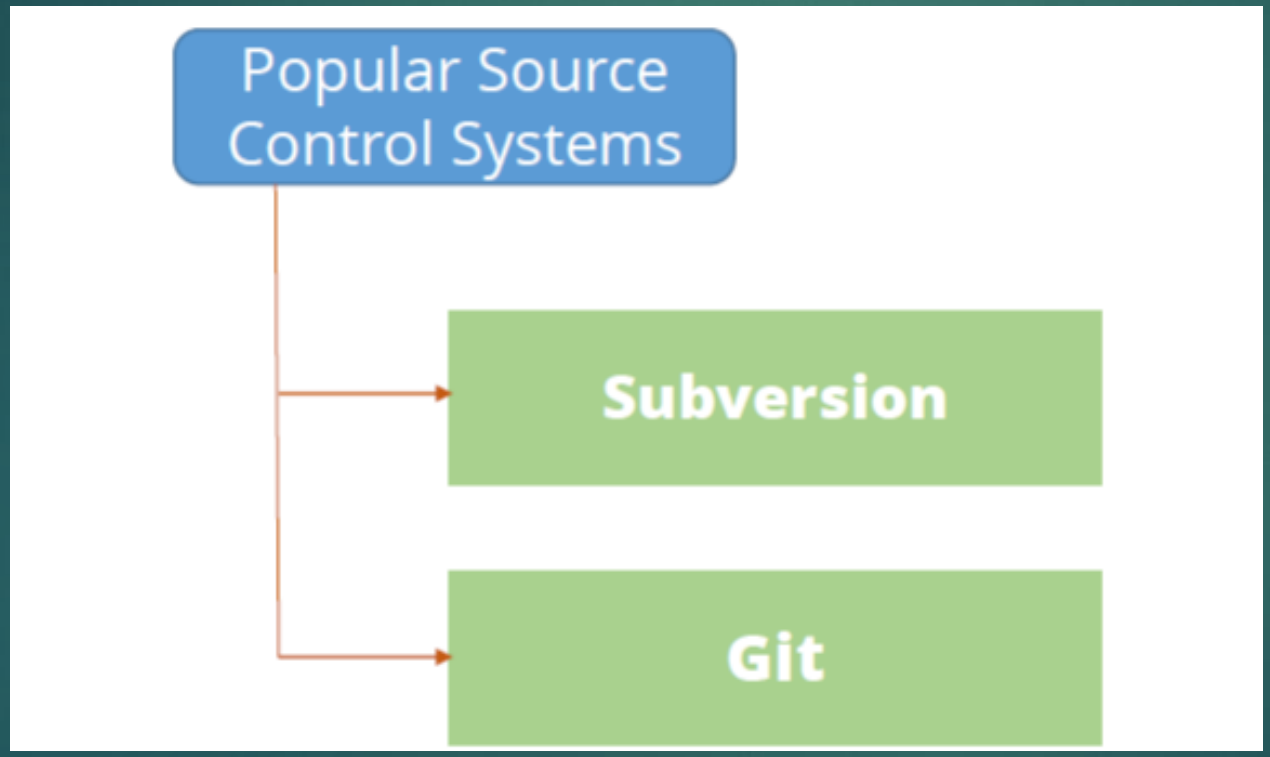


- Source Control Systems are organized into repositories / repos
- A repository can contain a single directory ,a library or a complete system

Types of Source Control Systems

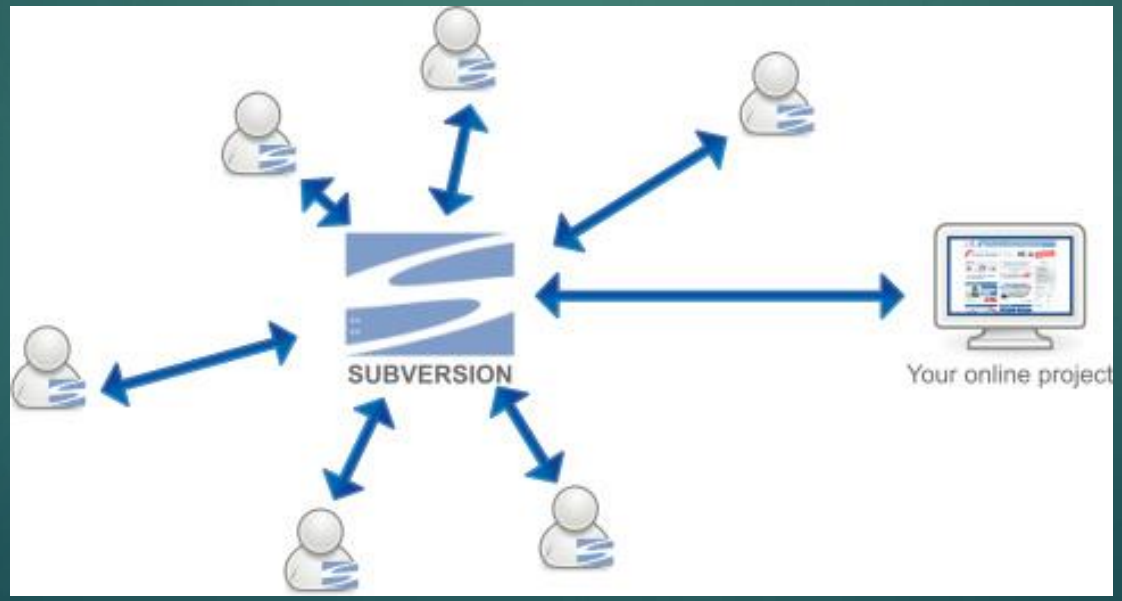


Subversion & Git

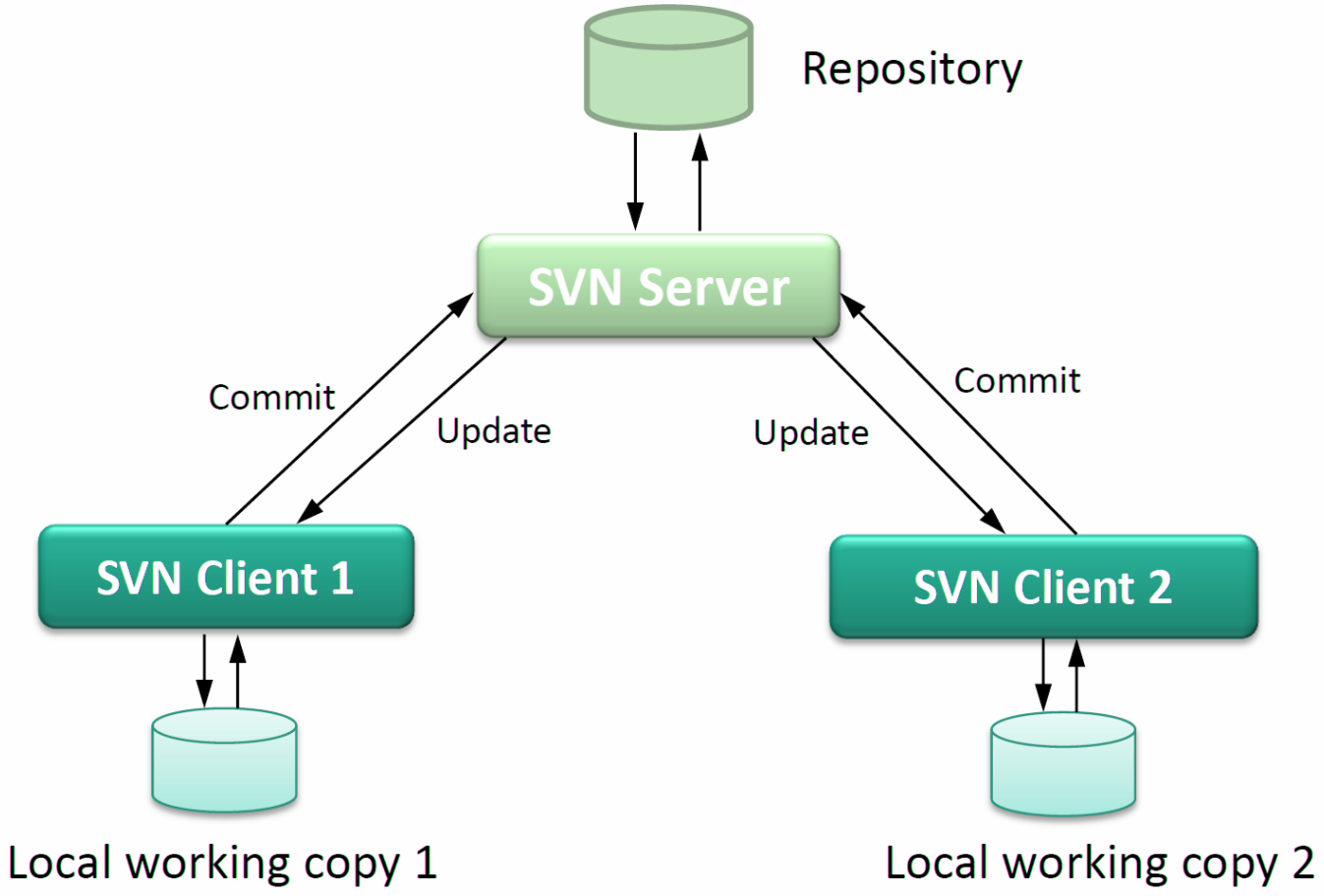


What is Subversion ?

- Open Source Control Version System
- Better than CVS
- Provides Command Line tools



Subversion Architecture



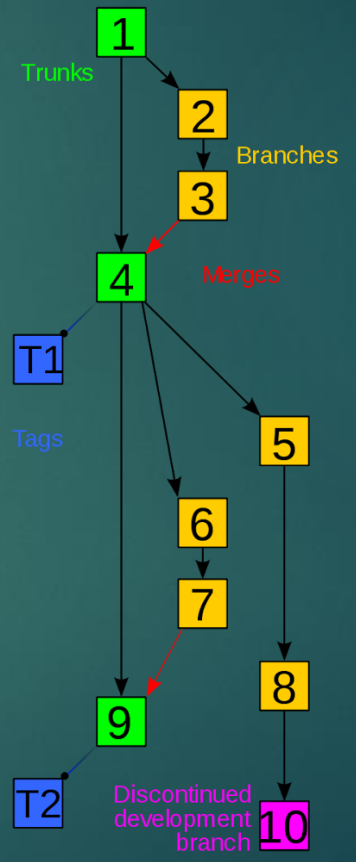
Working of SVN

Subversion usually uses structure with 3 folders:

Trunk: Contains latest source code, which is on development

Tags: Contains snapshot of project.

Branches: Contains different branches of project. Developers can work on multiple, simultaneous features without affecting others. Branches can be merged later after feature has been implemented



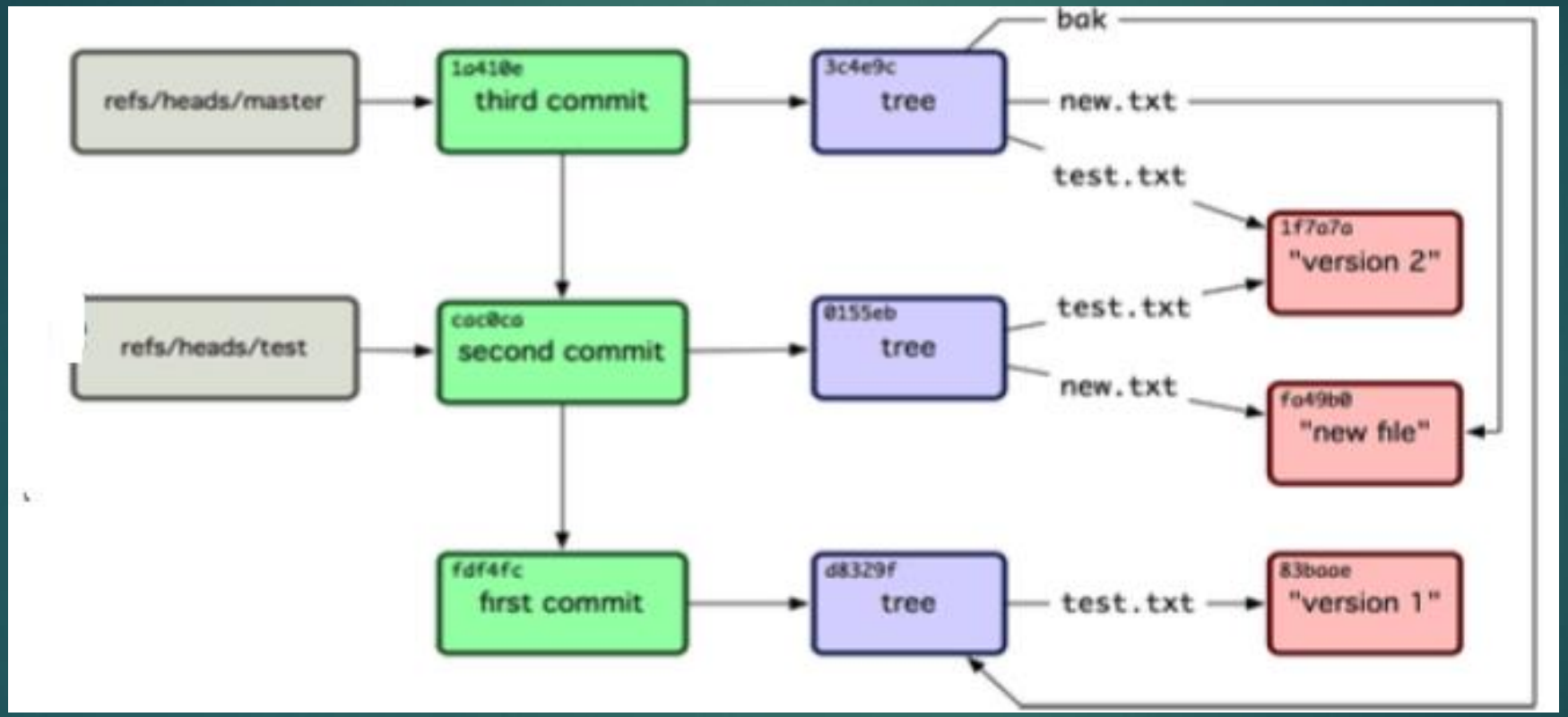
Working of SVN

- Trunk folder of repository
- From trunk folder, create new branch for development
- Work with new branch
- After new branch had completed its own feature, merge it with main branch. Project is stable now and will be tagged as T1
- Continue to develop
- Continue to develop
- New branch is created
- New branch is created
- New branch is completed, merge it onto main branch
- This branch is discontinued, development is no longer active

What is Git ?

- Distributed version control system developed by Linux Torvalds in 2005
- Most Popular and the fastest Open Source Version Control System

Git Design



Installing Git

- Installation of Git is very simple - https://git-scm.com/downloads
- Users should set email and names in the global git configuration

Provides information to be included in the commits

Useful for analysis of code changes over time

```
git config --global user.name "Your Name"
git config --global user.email user@domain
```

Creating Repository

- First create a Project which contains files
- Initialize git , which would create an empty .git directory
- Add Files
- Commit changes using commit command.

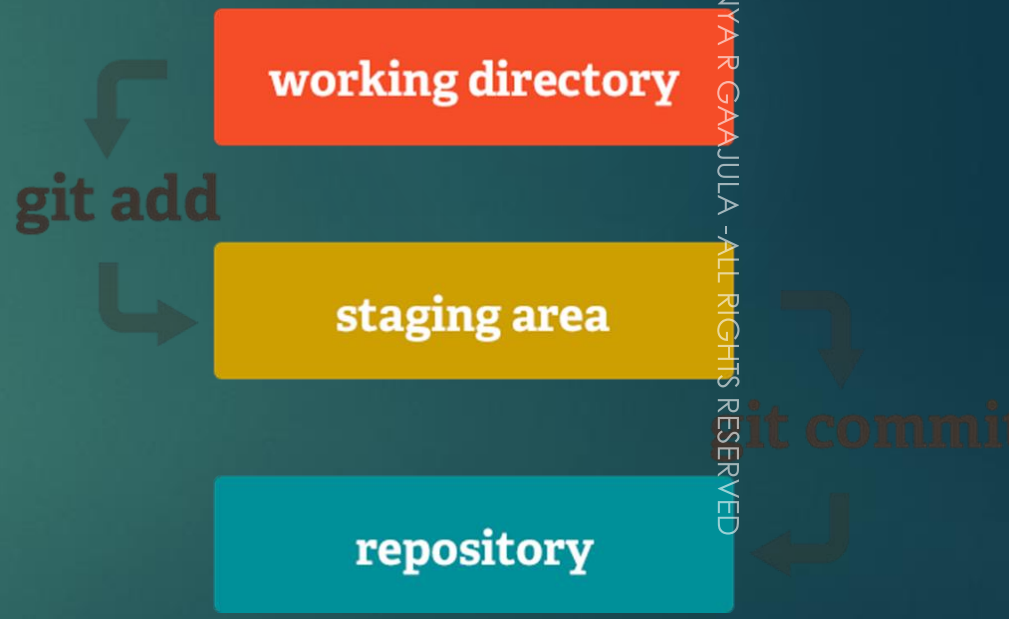
Adding and Committing Files

Git keeps tracks of all changes and additions –git status

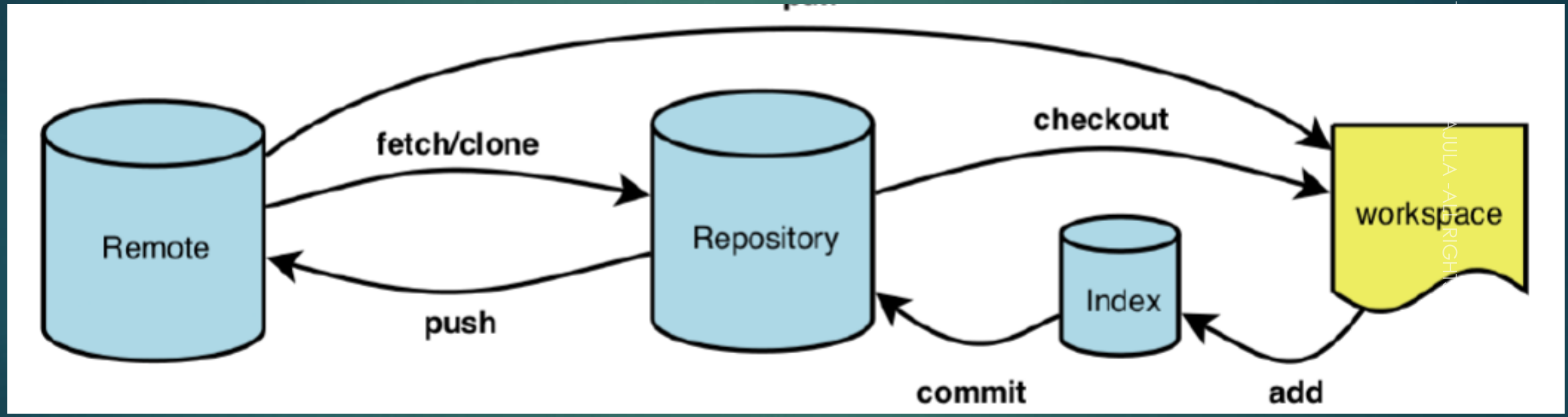
Changes must be first added - git add files

Changes must be committed to the repository – git commmit

```
git status
git add files
git status
git commit
```



Git Workflow

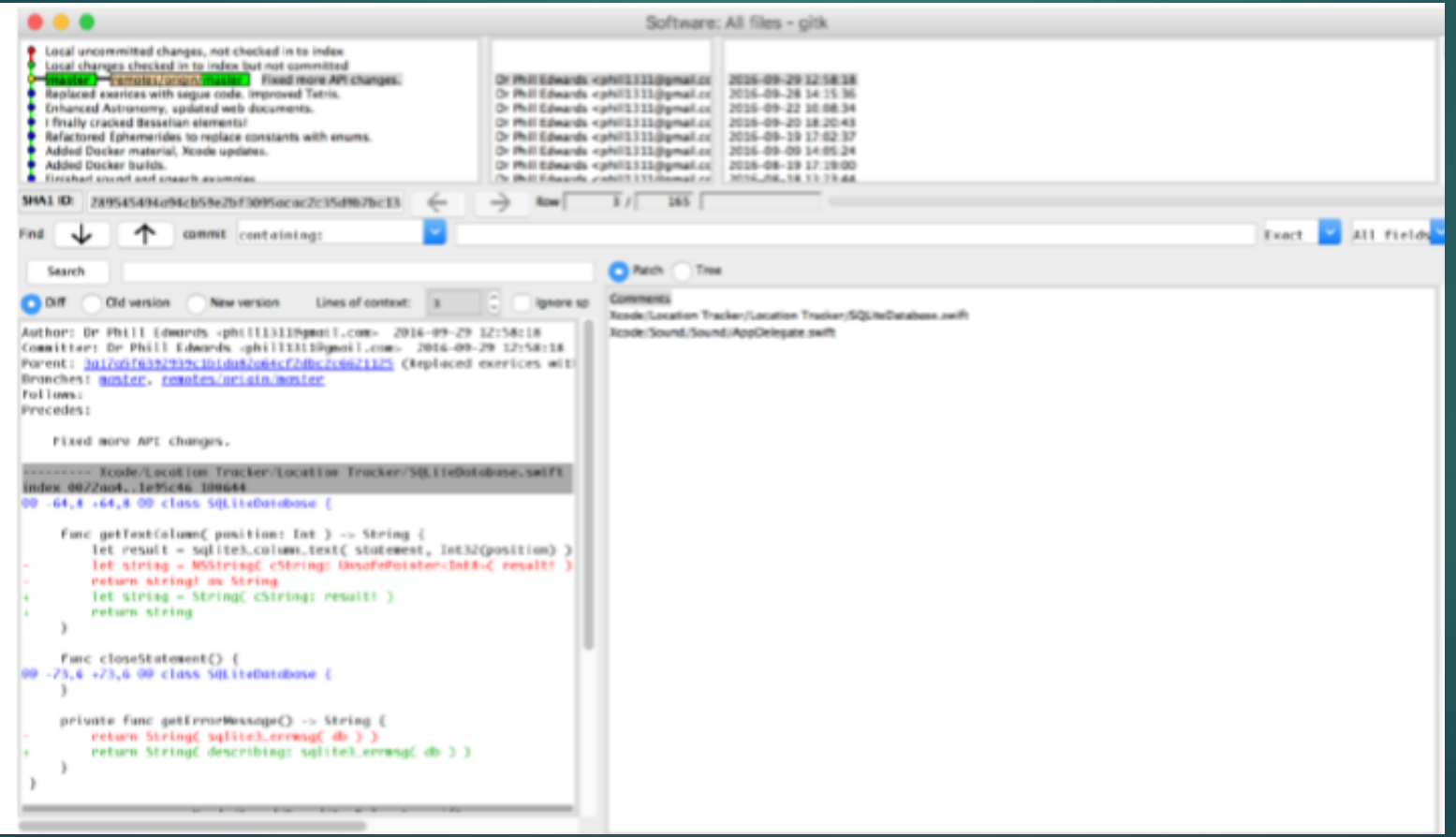


Git Branches

A branch in Git is simply a lightweight movable pointer to one of these commits. The default branch name in Git is master. As you start making commits, you're given a master branch that points to the last commit you made. Every time you commit, the master branch pointer moves forward automatically

```
git branch release1
git branch # Shows master as default
git checkout release1
# Make and changes
git commit -a
git checkout master
```

Gitk (GUI)



Git Repositories (Gitlab and Github)

What is GitHub?

- GitHub is a Git repository hosting service.
- GitHub provides a Web-based graphical interface.
- Also provides access control and several collaboration features such as a wikis and basic task management tools for every project.

THANK YOU