

Subversion & Mercurial: Version Control Systems

A DEMONSTRATION OF SVN & HG USAGE

By: ALVEERA AHMAD
500112367
CCVT B1

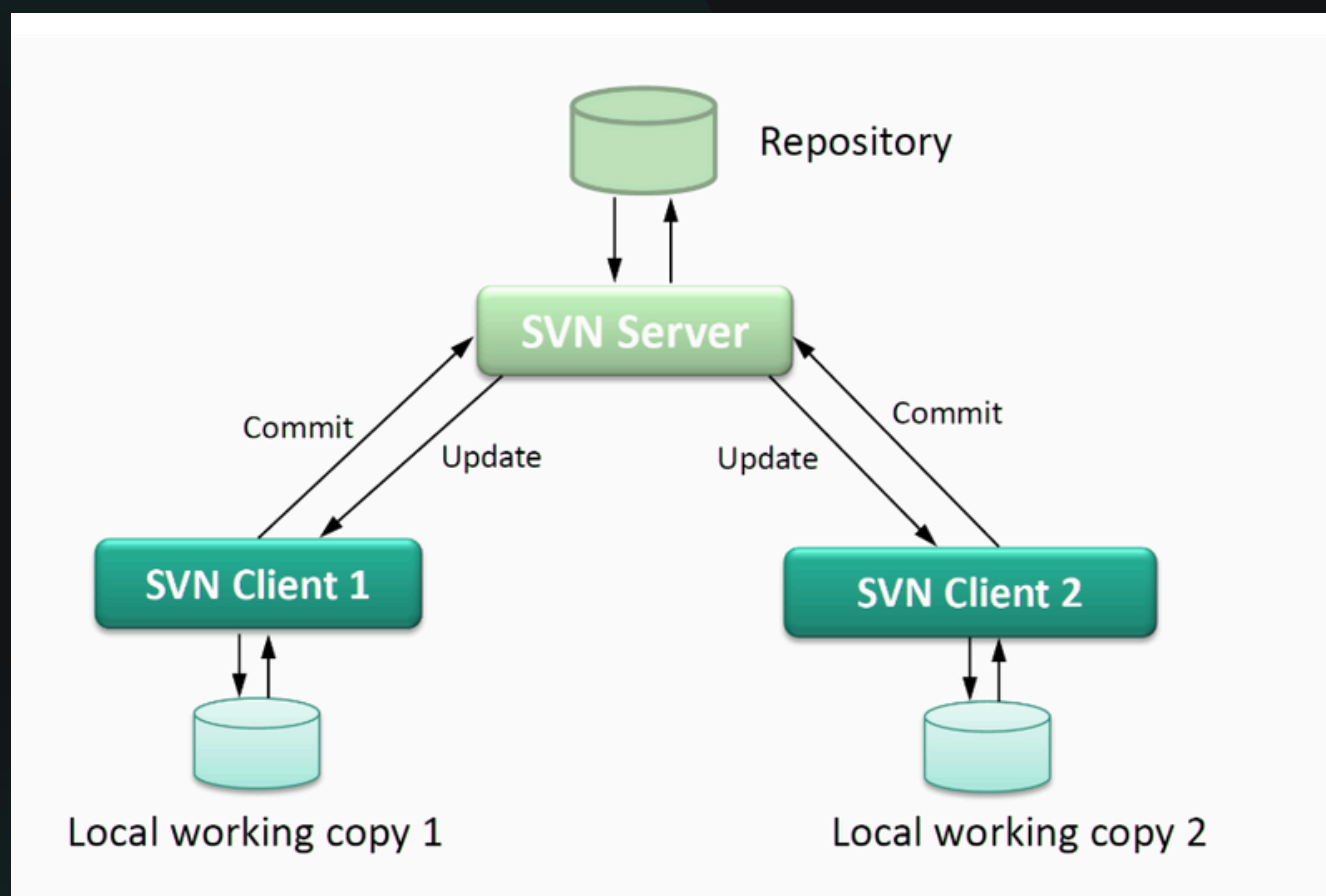
What is SVN?

- Centralized version control system.
- Maintains a single repository on a server.

Key Features:

- Version tracking
- Atomic commits
- Branching & tagging

SVN Architecture Diagram:



Installing SVN:

- macOS:
Install via terminal:
`brew install svn`

```
alv — -zsh — 80x24

Last login: Thu Feb 13 16:30:13 on console
alveera@Alveeras-MacBook ~ % svn --version
svn, version 1.14.5 (r1922182)
    compiled Nov 27 2024, 21:22:28 on arm-apple-darwin23.6.0

Copyright (C) 2024 The Apache Software Foundation.
This software consists of contributions made by many people;
see the NOTICE file for more information.
Subversion is open source software, see http://subversion.apache.org/

The following repository access (RA) modules are available:

* ra_svn : Module for accessing a repository using the svn network protocol.
  - with Cyrus SASL authentication
  - handles 'svn' scheme
* ra_local : Module for accessing a repository on local disk.
  - handles 'file' scheme
* ra_serf : Module for accessing a repository via WebDAV protocol using serf.
  - using serf 1.3.10 (compiled with 1.3.10)
  - handles 'http' scheme
  - handles 'https' scheme

The following authentication credential caches are available:
```

SVN Basic Commands:

- **svnadmin create <path>**

Create an SVN repository

- **svnserve -d -r <repo-path>**

Start SVN server

- **svn checkout <URL>**

Clone an SVN repository

- **svn add <file>**

Add file to SVN

- **svn commit -m "<message>"**

Save changes to repository

- **svn update**

Sync local copy with repository

- **svn status**

Check file status

- **svn log**

View commit history

- **svn delete <file>**

Remove file from SVN

```
svn: E200010: '/Users/alveera/..._alveera' path not found
[alveera@Alveeras-MacBook ~ % mkdir svn_repo
mkdir: svn_repo: File exists
[alveera@Alveeras-MacBook ~ % svnadmin create ~/svn_repo/my_project
[alveera@Alveeras-MacBook ~ % svnserve -d -r ~/svn_repo
```

```
alveera@Alveeras-MacBook ~ % svn checkout svn://localhost/my_project
```

```
Checked out revision 0.
```

```
alveera@Alveeras-MacBook ~ % cd my_project
```

```
alveera@Alveeras-MacBook my_project % svn add file.txt
```

```
alveera@Alveeras-MacBook my_project % svn commit -m "Added file.txt"
```

```
alveera@Alveeras-MacBook my_project % svn update
```

```
Updating '.':
```

```
At revision 0.
```

```
alveera@Alveeras-MacBook my_project % svn status
```

```
alveera@Alveeras-MacBook my_project % svn log
```

```
-----
alveera@Alveeras-MacBook my_project % svn delete file.txt
```

```
svn: E200005: Use --force to override this restriction (local modifications may be lost)
```

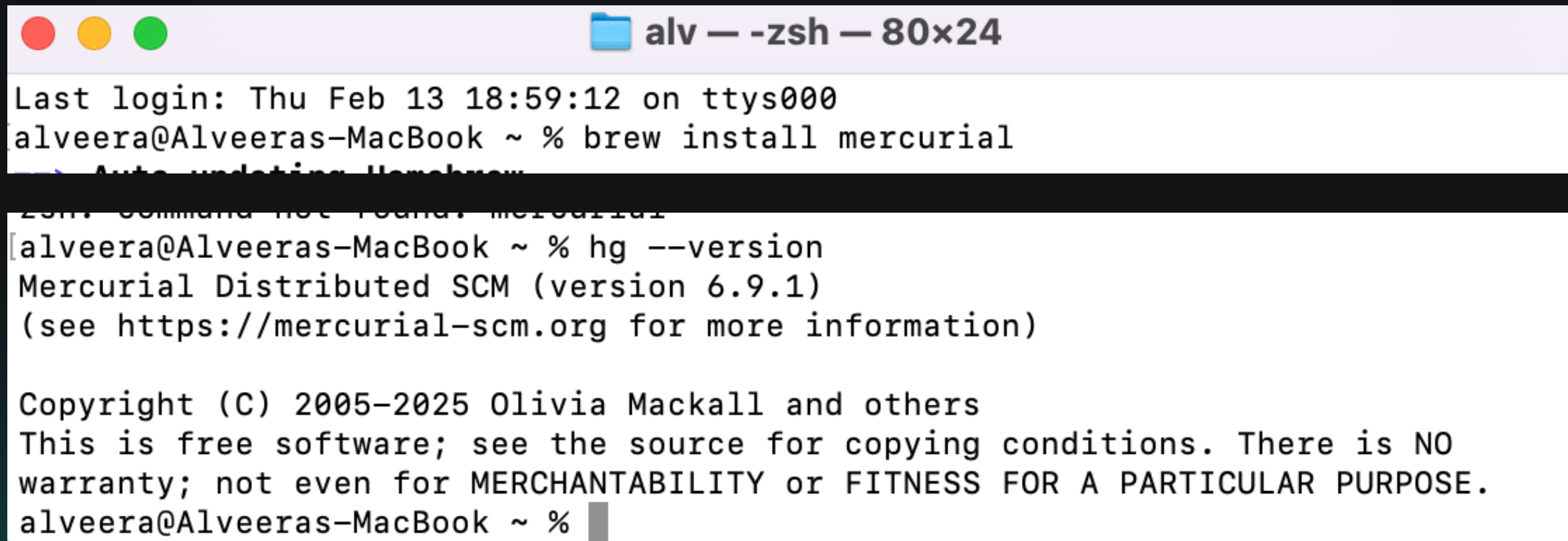
```
svn: E200005: '/Users/alv/my_project/file.txt' is not under version control
```

```
alveera@Alveeras-MacBook my_project % svn commit -m "Deleted file.txt"
```


Introduction to Mercurial (Hg)

- What is Mercurial?
 - A distributed version control system.
 - Alternative to Git, but easier to use.
- Key Features:
 - Simplicity
 - Fast performance
 - Branching and merging support.

Installing Mercurial:
brew install mercurial



```
alv — -zsh — 80x24
Last login: Thu Feb 13 18:59:12 on ttys000
alveera@Alveeras-MacBook ~ % brew install mercurial
Auto-updating Homebrew...
Error: Command not found: mercurial
alveera@Alveeras-MacBook ~ % hg --version
Mercurial Distributed SCM (version 6.9.1)
(see https://mercurial-scm.org for more information)

Copyright (C) 2005-2025 Olivia Mackall and others
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
alveera@Alveeras-MacBook ~ %
```

Mercurial Basic Commands

- `hg init` – Initializes a new Mercurial repository in the current directory.
- `hg clone <repo_url>` – Clones a remote repository to your local machine.
- `hg status` – Shows the status of files in the working directory (tracked, untracked, etc.).
- `hg add <file>` – Starts tracking a file in the repository.
- `hg commit -m "message"` – Commits changes to the repository with a message.
- `hg push` – Pushes local changes to a remote repository.
- `hg pull` – Fetches changes from a remote repository.
- `hg update` – Updates the working directory to a specific commit or branch.
- `hg log` – Views the commit history.
- `hg diff` – Shows differences between the working directory and the last commit.
- `hg branch <branch_name>` – Creates a new branch or switches to an existing branch.
- `hg merge` – Merges changes from another branch.