



DevOps

Roll No.: R2142220310

SAP ID:500105040

Batch: B2

Semester: VI

Submitted by: Anugrah Narayan Kumar

Submitted to: Dr. Prateek Raj Gautam

VERSION CONTROL ASSIGNMENT-1

SUBVERSION

Subversion software is also known as SVN. It is an open source version control system. Through subversion we can look at the previous version of the file and track the changes over time.

There are two types of Version Control System:

- **Centralized Version Control System(CSCV):** There is a single central server that stores all the versions.
- **Distributed Version Control System(DVSC):** Each user have the copy of full repository.

Why do we use Version Control?

- To track all the changes and keep the history.
- We can rollback to the previous version when needed.
- We can merge new features.

Installing & Setting up SVN on windows:

1. Download and install SVN ([TortoiseSVN](#))
2. After the installation restart your system.
3. Verify the installation.

svn --version

```
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Jitendra Kumar>svn --version
svn, version 1.14.5 (r1922182)
   compiled Dec 16 2024, 22:40:41 on x86_64-microsoft-windows6.2.9200

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.
  - 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:

* Wincrypt cache in C:\Users\Jitendra Kumar\AppData\Roaming\Subversion
```

SVN Commands:

Step 1: Initialize the repository.

`svnadmin create ~/svn_repo/my_project`

```
C:\Users\Jitendra Kumar>svnadmin create D:\Anugrah_repo
```

Step 2: Checkout the repository

`svn checkout file:///path/to/svn_repo/my_project`

```
C:\Users\Jitendra Kumar>svn checkout file:///D:/Anugrah_repo Anugrah-project
Checked out revision 0.
```

Step 3: Add files to the directory

- Navigate to your working directory.
- Add new files.
- Commit the changes.

`cd my_project`

`svn add file.txt`

`svn commit -m "Added file.txt"`

```
C:\Users\Jitendra Kumar>cd Anugrah-project
C:\Users\Jitendra Kumar\Anugrah-project>echo "Hello SVN" > Anugrah_file.txt
C:\Users\Jitendra Kumar\Anugrah-project>svn add Anugrah_file.txt
A
    Anugrah_file.txt
```

```
C:\Users\Jitendra Kumar\Anugrah-project>svn commit -m "Initial commit by Anugrah"
Adding      Anugrah_file.txt
Transmitting file data .done
Committing transaction...
Committed revision 1.
```

Step 4: Update your working copy and view the logs

svn update

svn log

```
C:\Users\Jitendra Kumar\Anugrah-project>svn update
Updating '.':
At revision 1.

C:\Users\Jitendra Kumar\Anugrah-project>svn log
-----
r1 | Jitendra Kumar | 2025-02-15 18:49:17 +0530 (Sat, 15 Feb 2025) | 1 line
Initial commit by Anugrah
-----
```

Step 5: Reverting the changes

svn revert file.txt

```
C:\Users\Jitendra Kumar\Anugrah-project>svn revert Anugrah_file.txt
```

Step 6: Creating the branch and merge the changes.

svn copy file:///C:/svn_repos/my_repo/trunk

file:///C:/svn_repos/my_repo/branches/feature-branch -m "Creating feature branch"

svn merge file:///C:/svn_repos/my_repo/branches/feature-branch

```
C:\Users\Jitendra Kumar\Anugrah-project>svn mkdir file:///C:/Anugrah_repo/branches -m "Creating Branches Directory"
svn: E170013: Unable to connect to a repository at URL 'file:///C:/Anugrah_repo'
svn: E180001: Unable to open repository 'file:///C:/Anugrah_repo'

C:\Users\Jitendra Kumar\Anugrah-project>svn mkdir file:///D:/Anugrah_repo/branches -m "Creating Branches Directory"
Committing transaction...
Committed revision 3.

C:\Users\Jitendra Kumar\Anugrah-project>svn copy file:///D:/Anugrah_repo/trunk file:///D:/Anugrah_repo/branches/new-features -m "Creating a branch by Anugrah"
Committing transaction...
Committed revision 4.

C:\Users\Jitendra Kumar\Anugrah-project>
```

MERCURIAL (HG)

Mercurial is a distributed version control system (DVCS) designed for efficient handling of the projects of all sizes. The functionality of Mercurial is similar as Git but it emphasizes simplicity and ease of use. Mercurial is written in python and it is known for its intuitive commands, robust performance and cross-platform compatibility.

Features of Mercurial:

- **Distributed Version Control:** Every developer has a fully copy of the repository, enabling offline work and independent branching.
- **Lightweight and Fast:** Efficient handling of large projects and binary files.
- **Cross-Platform:** It works on Windows, macOS and Linux.
- **Extensible:** It supports plugins for additional functionality.
- **Simple and Intuitive Commands:** Commands are easy to learn and use, with a consistent syntax.

Installing & setting up Mercurial on Windows:

1. Download and install Mercurial ([TortoiseHg](#))
2. After the installation restart your system.
3. Verify the installation.

hg –version

```
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Jitendra Kumar>hg --version
Mercurial Distributed SCM (version 6.5.1)
(see https://mercurial-scm.org for more information)
```

```
Copyright (C) 2005-2023 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.
```

Mercurial Commands

Step 1: Creating and Initializing the Repository

hg init my-hg-repo

```
C:\Users\Jitendra Kumar>hg init Anugrah_hg_repo
C:\Users\Jitendra Kumar>cd Anugrah_hg_repo
C:\Users\Jitendra Kumar\Anugrah_hg_repo>
```

Step 2: Adding files and committing the files

hg add file.txt

hg commit -m "Added newfile.txt"

```
C:\Users\Jitendra Kumar\Anugrah_hg_repo>echo "Hello Mercurial" > Anugrah_file.txt
C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg add Anugrah_file.txt
```

```
C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg commit -m "Initial commit by Anugrah"
nothing to commit, working tree clean
```

Step 3: Cloning, Updating and Reverting

hg clone https://example.com/repo

hg pull

hg update

hg log

```
C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg clone "C:/Users/Jitendra Kumar/Anugrah_hg_repo" "C:/Users/Jitendra Kumar/Anugrah_hg_clone"
updating to branch default
1 files updated, 0 files merged, 0 files removed, 0 files unresolved

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg pull
pulling from default
abort: repository default not found

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg pull "C:/Users/Jitendra Kumar/Anugrah_hg_repo"
pulling from C:/Users/Jitendra Kumar/Anugrah_hg_repo
searching for changes
no changes found
1 local changesets published

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg update
0 files updated, 0 files merged, 0 files removed, 0 files unresolved

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg log
changeset:  0:e23a57b6dfbd
tag:        tip
user:       Anugrah <anugrahkvs@gmail.com>
date:       Sat Feb 15 19:21:53 2025 +0530
summary:    Initial Commit by Anugrah

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg revert Anugrah_file.txt
no changes needed to Anugrah_file.txt
```

Step 4: Branching and Merging

hg branch new-feature

hg merge

```
C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg branch Anugrah-feature
marked working directory as branch Anugrah-feature
(branches are permanent and global, did you want a bookmark?)

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg commit -m "Creating a new feature branch by Anugrah"

C:\Users\Jitendra Kumar\Anugrah_hg_repo>echo "This ia an update from Anugrah-feature branch" >> Anugrah_file.txt

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg commit -m "Updated Anugrah_file.txt in Anugrah-feature branch"

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg update default
1 files updated, 0 files merged, 0 files removed, 0 files unresolved

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg merge Anugrah-feature
1 files updated, 0 files merged, 0 files removed, 0 files unresolved
(branch merge, don't forget to commit)

C:\Users\Jitendra Kumar\Anugrah_hg_repo>hg commit -m "Merged Anugrah-feature into default"
```