SVN

SVN

- SVN is an open-source centralized version control system
- Designed to handle minor to major projects with speed and efficiency. It is developed to co-ordinate the work among programmmers.
- SVN is used to manage both current and past versions of files, such as source code, documentation, and other types of files.

Features of SVN

- **Atomic Commits:** All changes in a commit either succeed together or fail together, preventing incomplete changes.

- **Centralized Repository:** All the data is stored in a single central repository, allowing multiple users to collaborate on the same project.

- **File Locking Mechanism:** Useful when working with non-mergeable files like images or binary data, preventing conflicts when multiple users access the same file.

- **Support for Multiple Protocols:** SVN works with multiple access protocols, such as **HTTP(S)**, **SSH**, and its own **SVN protocol**, offering flexibility in how users interact with the repository.

- **Branching and Tagging:**

- **Efficient Large File Handling:** Providing better performance and storage management for large files compared to some other version control systems.

- **Cross-Platform Support:** SVN is compatible with multiple platforms, including **Windows**, **Linux**, and **macOS.**

- **Versioning for Files and Directories:** Both files and directories maintain full track of the changes done on the project.

Components of SVN

There are two core components of SVN

- SVN Server
- SVN Client

SVN Server

- It store the whole project in the centralized remote devices.
- The server operates on the central master copy of your projects in a database.

The usage of SVN server is as follows:

- It provides security to customer.
- It offers remote access.
- It provides the read and write access to multiple destinations through the SVN client-server.
- It can store the complete version of the requirements.

SVN Client

- It provides the interface that helps and allows us to interact with the server. SVN client is free and open-source.

The usage of SVN client is as follows:

- It communicates with the SVN server.
- It provides remote and manual access.
- It provides the interface.
- It is free and open-source.

SVN vs Git

1. Centralized (SVN) vs Distributed (Git)
2. Branching: Heavy (SVN) vs Lightweight (Git)
3. Speed: Slower (SVN) vs Faster (Git)
4. Offline Work(for commit): Limited (SVN) vs Full (Git)
5. Merge Handling: More complex (SVN) vs Easier (Git)
6. File Locking: Available (SVN) vs Not built-in (Git)
7. History: Linear (SVN) vs Non-linear (Git)

When to Use SVN Over Git

- Centralized Control and Governance
- Handling Large Binary Files
- Simpler Learning Curve
- Single Source of Truth
- File Locking Needs
- Legacy Projects and Existing Infrastructure
- Smaller Teams or Projects
- Compliance and Auditing

Pros

- Centralized Control
- Atomic Commits
- Efficient Handling of Large Files
- Simple Workflow
- File Locking
- Branching and Tagging
- Partial Checkouts
- Good Integration with Legacy Tools
- Revision History

Cons

- Centralized Model
- Limited Offline Work
- Slower Performance
- Branching Complexity
- Less Efficient for Distributed Teams
- Higher Server Load
- Merge Conflicts
- Less Popular/Modern