9. Version Control

9.1 Introduction

- Record all changes and modifications
- Source Control *or* Source Code Management
- Add - Modify/Update - Delete
- **Benefits**
  - *Revision history*
    - Provides a record of all changes in a project
    - Ability to revert back to the stable point in case something causes bugs or issues
    - Roll-back
    - When
  - *Identity*
    - Changes are stored with the identity of people who made them
    - Enhances responsibility
    - Who
  - *Collaboration*

        - Allows better collaboration with other team members
        - Code Inspection - Code Reviewed by Others
    - *Automation*
    - *Efficiency*

9.2 Systems of Version Control

- Some Examples
    - Subversion
    - Perforce
    - AWS Code Commit
    - Mercurial
    - Git
- Types
    - Centralized Version Control Systems (CVCS)
        - Server-Client Model
        - Server : Contains main repository
        - Client : Needs to pull down code
        - Changes needs to be pushed back
        - Needs to maintain connection with server
        - Easier to learn
        - More access control
        - But slower
    - Distributed Version Control Systems (DVCS)
        - Each client behaves as server
        - Allows users to work in offline state
        - Better speed