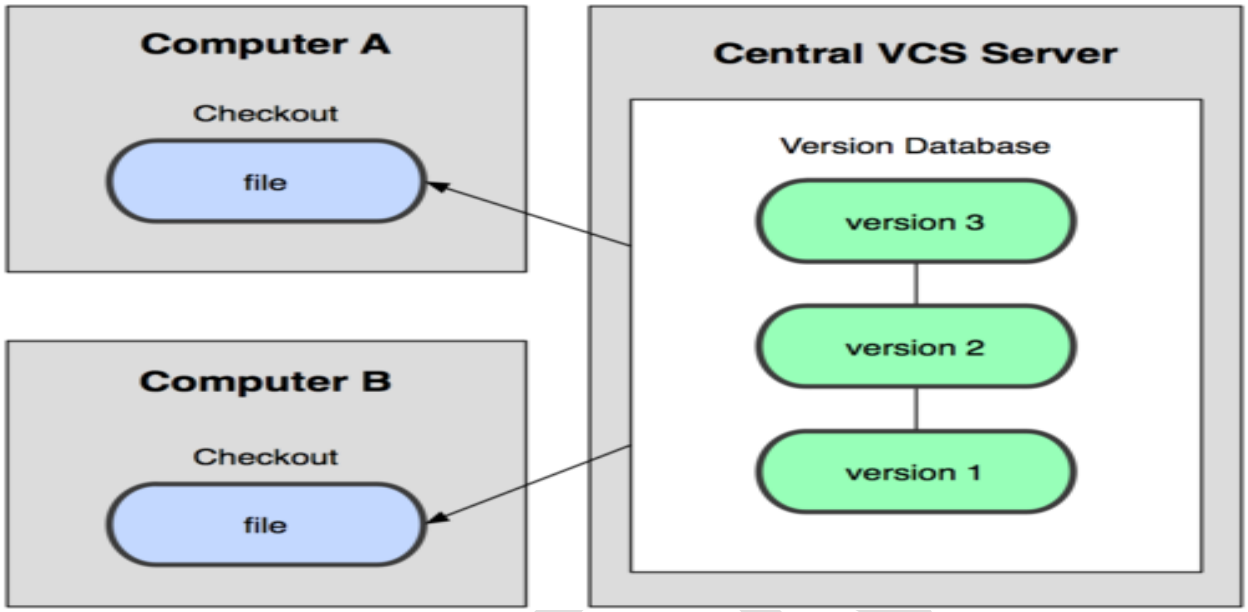
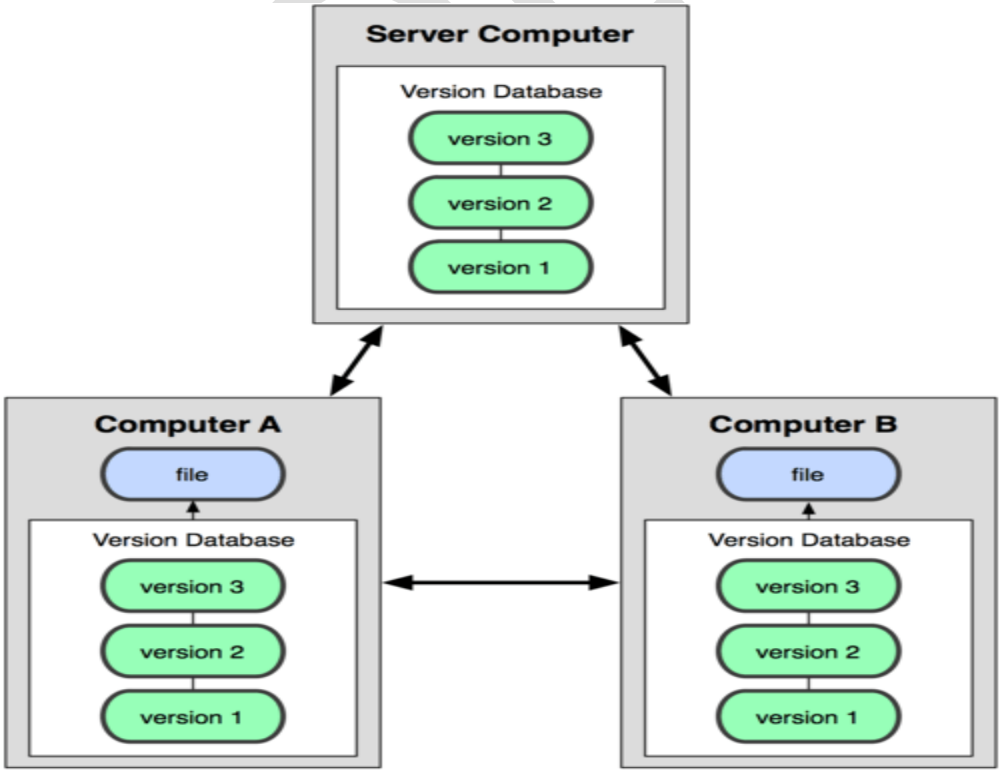
Why only Git?

- Speed
    - Snapshots concept
- Parallel branching
    - Multiple branches at a time unlike other SCM tools
- Fully Distributed
    - Backup copy is available in multiple locations.
    - No need Internet connection. So no network latency.
    - No need central server separately
    - Each work space will have its own repo internally
    - Can create any no of branches
    - Can share code without using central repo
    - That's why we call GIT as DVCS (Distributed Version Control System)

CVCS vs DVCS



CVCS



DVCS

Important Terminology

Snapshots:

- Get any previous version (Backup)
- Represents some data of particular time
- Stores the changes (appended data) only. Not whole copy.

Commit:

- Store changes in repo (Will get commit ID)
- 40 Alpha-numeric characters
- Concept - Checksum (It's a tool in Linux generates binary value equal to data present in file)
- Even if you change one dot, Commit-ID will be changed
- Track the changes

Git stages

Work space:

- Physically see file & Modify
- Staging/Indexing area
- Buffer area
- Takes snapshot

Repository (Local)

- Store changes locally

Repository (Central)

- Store changes Centrally

Types of Repositories

Bare Repositories (Central)

- Store & share only
- All central repositories are bare repositories

Non – Bare Repositories (Local)

- Where you can modify the files
- All local repositories are non-bare repositories