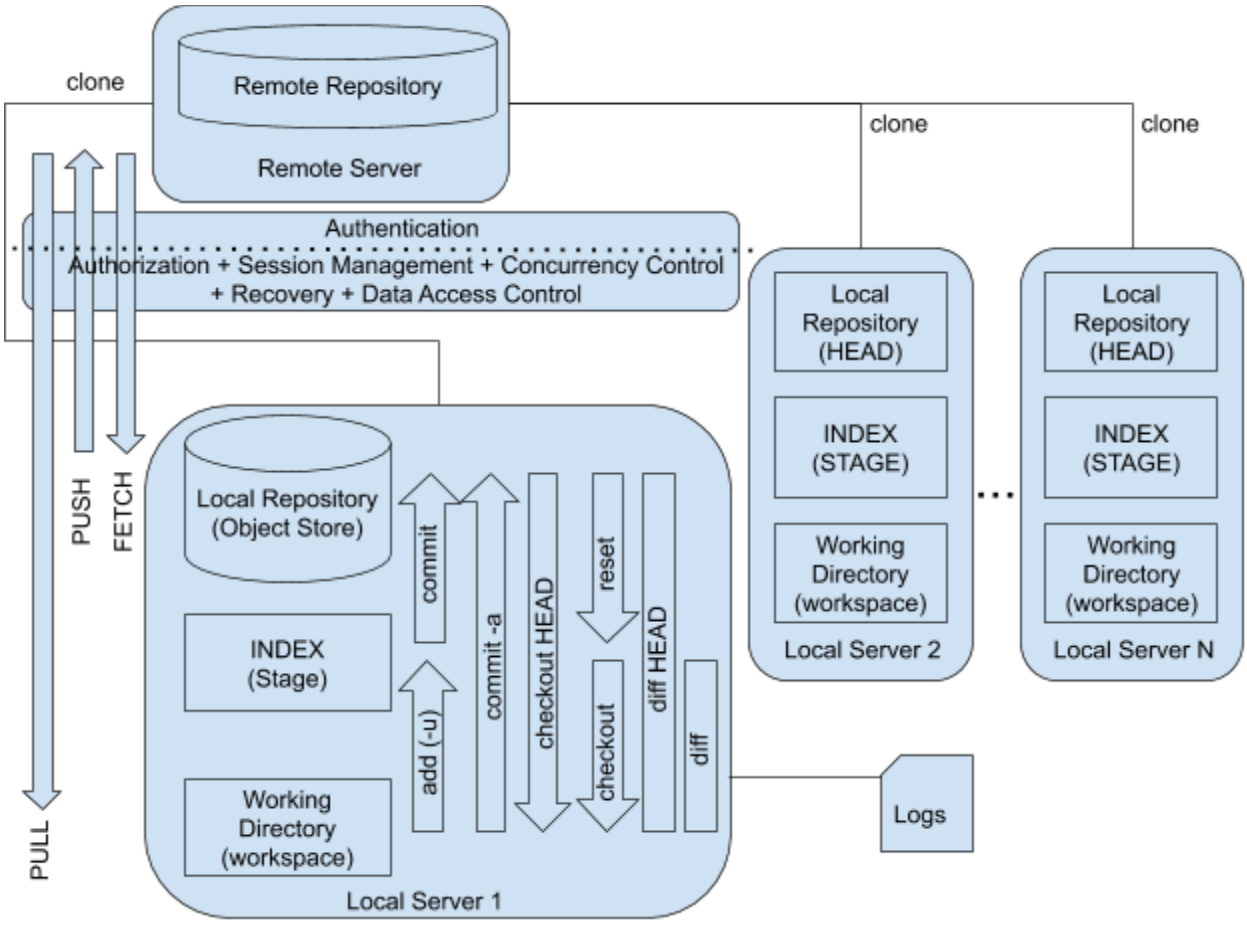
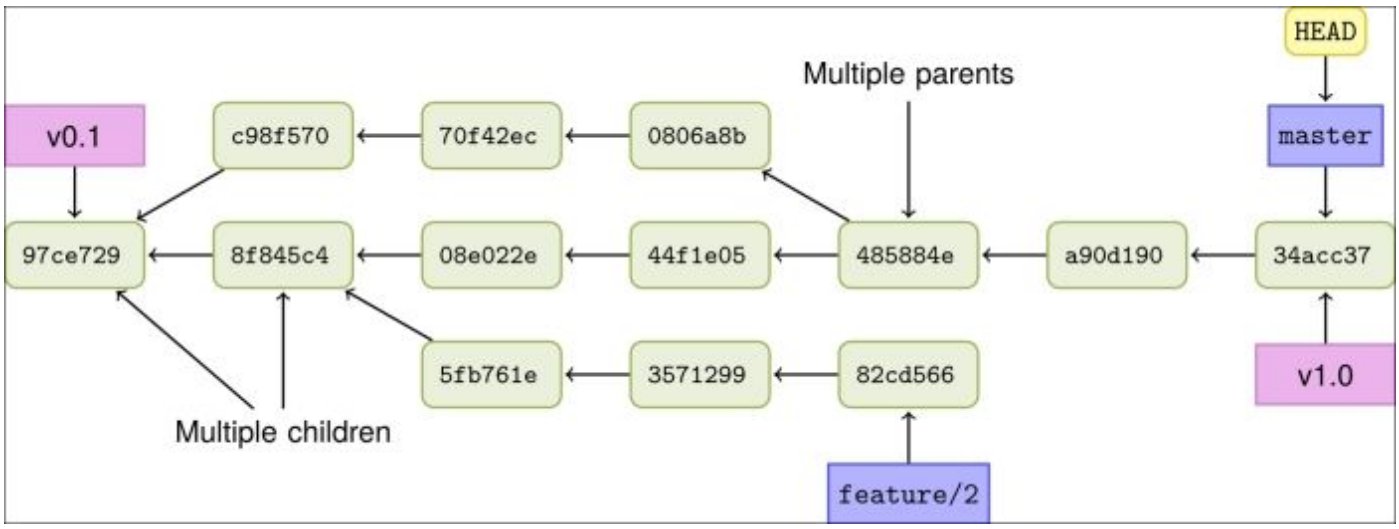
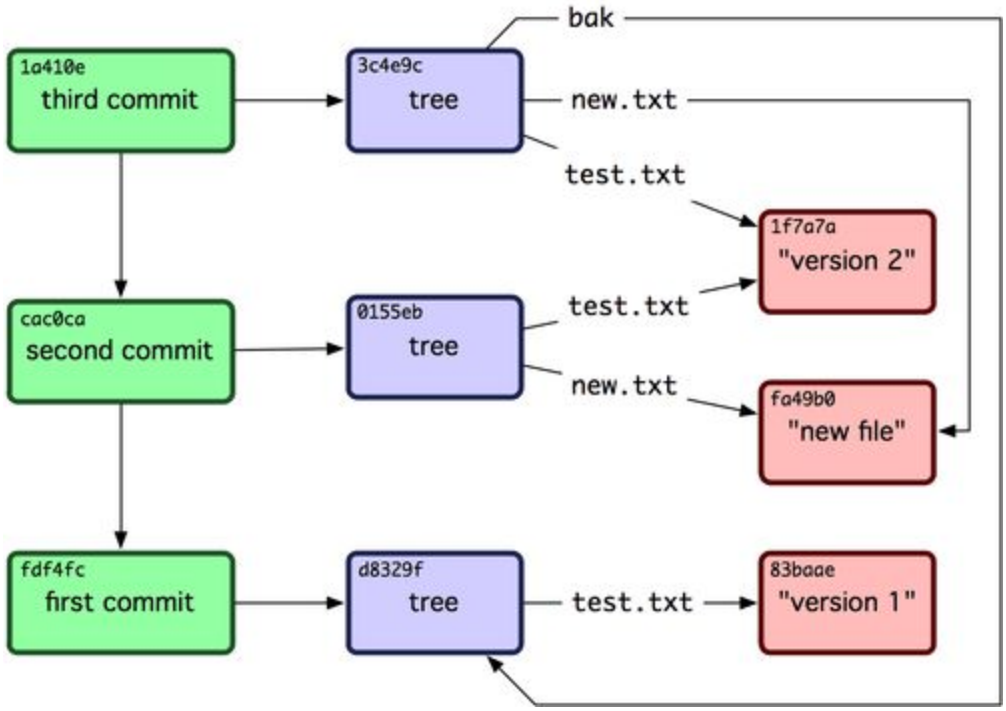
1. **Distributed version control architecture:**



2. **Object Store (Head pointer to Directed Acyclic Graph of Commits):**



3. **Commit (pointer to previous commit and current git tree representing hierarchical file structure for current version of files):**



Definitions:

Working Directory: Files that have been modified but not committed yet
Remote Repository: Hosted repository on a shared server
Local Repository: Local copy of the remote repository
Index: Pointer to Git Tree which has been generated due to some addition/modification in file
Head: Pointer to current branch
Master: Default branch we get on local machine post cloning a repo from remote
Ignore: List of all the files which we do not want being tracked inside git repo
Commit: Commit object contains hash of git tree for the files being tracked
Branch: Pointer to commit along with metadata attached to it.
Log: List of all commits along with message and author name sorted by timestamps
Object: Directed Acyclic Graph of commits
Hooks: Scripts that run whenever any particular event occurs in git repo
Blob: Object which stores the SHA-1 hash of content of some file in repo
Git tree: It is a **Hash tree (Merkle tree)**, which is a hierarchical representation of directory and files along with the hash of content at each level. A new git tree gets indexed everytime we modify/add a file. A git tree creates a new file only for files which have been modified/added thus points to some existing (unchanged) files and a few (newly added or) modified files. Difference can be computed by comparing two git trees.

Thanks
Nitin Mishra
Created on 4th July 2020