**GIT**

Git is a distributed version control system(DVCS) and source code management system with an emphasis on speed. Git was initially designed and developed by Linus Torvalds for Linux kernel development.

DVCS clients not only check out the latest snapshot of the directory but they also fully mirror the repository. If the server goes down, then the repository from any client can be copied back to the server to restore it. Every checkout is a full backup of the repository. Git does not rely on the central server and that is why you can perform many operations when you are offline. You can commit changes, create branches, view logs, and perform other operations when you are offline. You require network connection only to publish your changes and take the latest changes.

**Advantages of Git**

### Free and open source

### Implicit backup

### Security

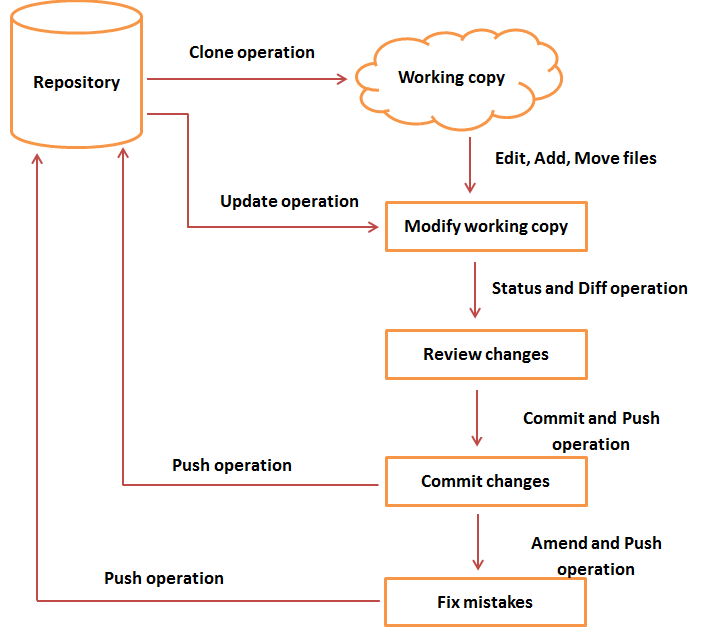
### No need of powerful hardware

### Easier branching

**Working directory and staging area**



**Git Life Cycle**



**Git Installation**

**Git server Installation(Linux):**

**Step-1**: Initial pre-requisite packages:

#yum install openssl-devel-\* -y

#yum install curl-devel-\* -y

#yum install expat-devel-\* -y

#yum groupinstall ‘Development Tools’ -y

**Step-2:** Git installation

#tar –xzv git-x.x.x.

#cd git

#mkdir /git1

#make prefix=/git-home all

#make prefix=/git-home install

**Step-3:** PATH

#vi ~/.bashrc

export PATH=/git1/bin:$PATH

**Step-4:** Check

#git --version

**Git Client side Installation(Linux)**

Follow the same steps as git server Installation

**Git Client side Installation(Windows)-gitbash**

Install git-x.x.x windows version

**Setting up Git(bare) repository on server**

#mkdir git\_repo

#cd git\_repo

#git init - -bare

The above 3 commands lets you create a server repository

**Setting up repository on client side**

**Linux client**

#mkdir git-client

#cd git\_client

#git clone [user@server ip:/git\_repo](mailto:root@192.168.242.133:/git_repo)

**Windows client**

-open git-bash and run the following commands

#mkdir git-client

#cd git\_client

#git clone [user@server ip:/git\_repo](mailto:root@192.168.242.133:/git_repo)

**Setting up Environment**

### Setting username

This information is used by Git for each commit.

**#**git config --global user.name "yourname"

### Setting email id

This information is used by Git for each commit.

**#**git config --global user.email "yourname@mail.com"

### Setting default editor

By default, Git uses the system default editor, which is taken from the VISUAL or EDITOR environment variable. We can configure a different one by using git config.

**#**git config --global core.editor vim

### Setting default merge tool

Git does not provide a default merge tool for integrating conflicting changes into your working tree. We can set default merge tool by enabling following settings.

**#**git config --global merge.tool vimdiff

### Listing Git settings

To verify your Git settings of the local repository, use **git config –list**command as given below.

#git config --list

The above command will produce the following result.

user.name=yourname

user.email=yourname@mail.com

push.default=nothing

branch.autosetuprebase=always

color.ui=true

color.status=auto

color.branch=auto

core.editor=vim

merge.tool=vimdiff

**Creating a bare(server) repository**

Server side:

#mkdir server\_repo

#cd server\_repo

#git init --bare

**Cloning a git repository**

Client side:

#mkdir client\_repo

#cd client\_repo

#git clone user@server ip:/server\_repo

**Adding files**

**Cycle:**

1. Create a file

>touch hello.txt

>git status

1. Stage the file

>git add hello.txt

>git status

1. Commit the file

>git commit hello.txt –m “committing hello.txt”

1. Push the file to remote repository

>git push

>git pull