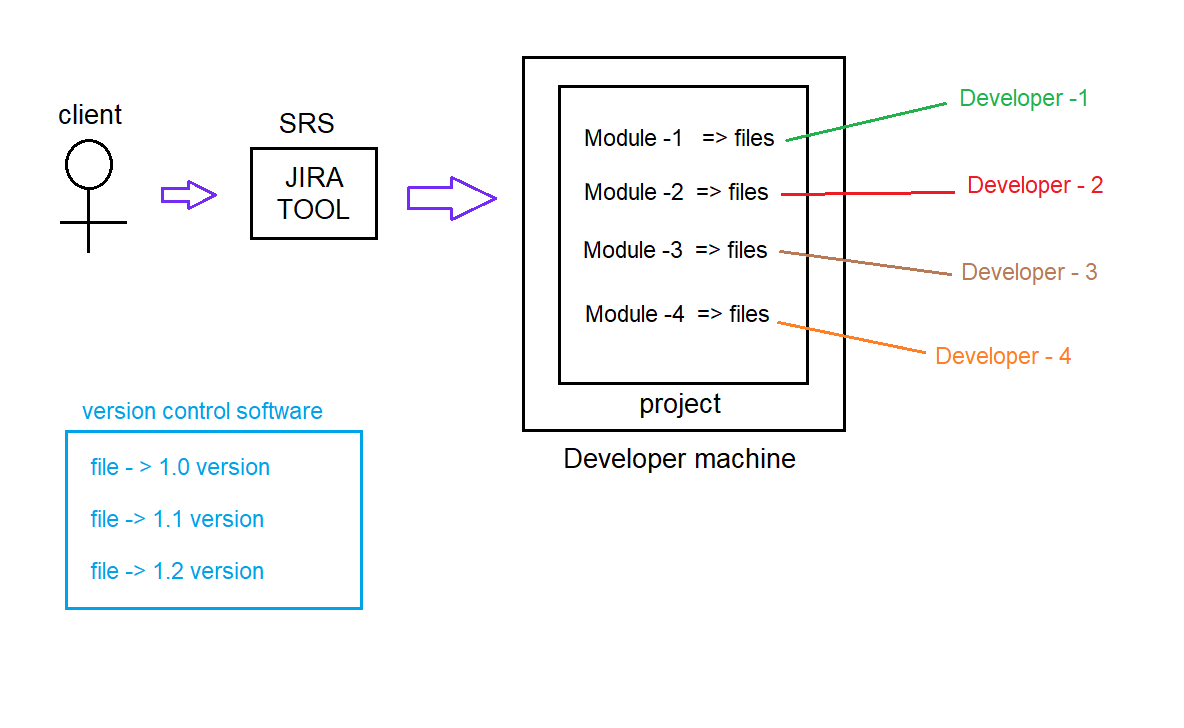
* Git is a version control system (vcs )
* Created by linus Torvalds in 2005
* since 2005 junio hamano is the core maintainer .

Git is used for

1. Tracking code changes
2. Tracking who made changes like history of files
3. Making coding collaborations



Client : a person who has requirement

Keeping the track of changes made to the code by the developer as per the client requirement would be difficult to maintain

To resolve this problem we use version control system .

Srs -> software requirement specification

For client requirement developer will write srs , srs will be maintained by jira tool.

Vcs (version control system)

It is a system that records the changes made to a file or a group of files , so that we can recall specific version if we want

i.e for every source code changes in a file a new version will be created

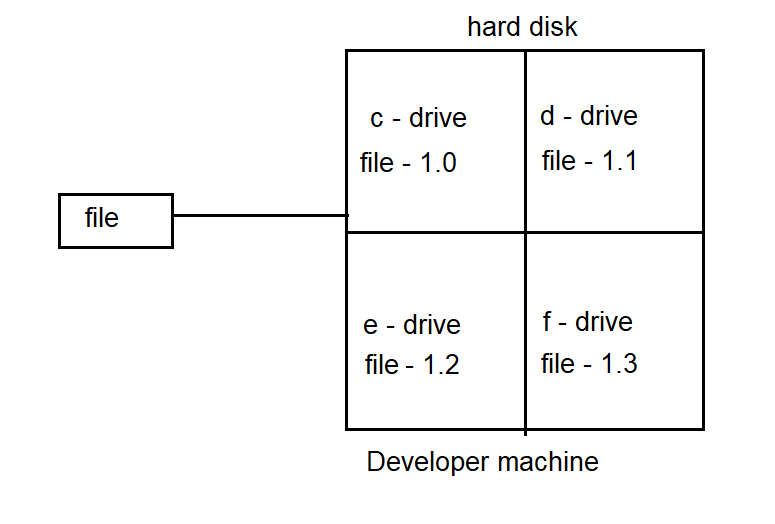
eg: jdk 1.0 ,1.1,…

Types of version control system

1. Local version control system
2. Central version system
3. Distributed version control system

Local version control system

It is used to maintain file version and retrieve files based on specific version .



Drawbacks

1. There is a situation that you will try to save the file in any drive and but already that file exists and you will overwrite it save .
2. If the hard disk is corrupted there is a possible loss of data .
3. By mistake we can delete some files also .

Centralized version control system (cvcs)

To overcome drawbacks of local version control system we use centralized version control system .

Developers can collaborate with code in one repository and do the change .

Eg: Subversion (svn) , perforce

Centralized version control system contains only one server and maintains all version files

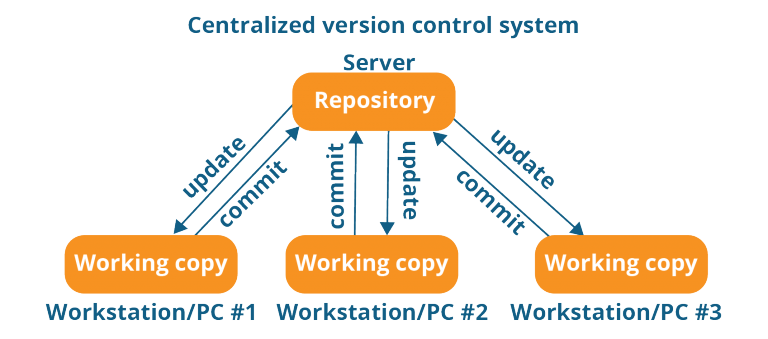
For many years it is a standard version control system

More no of developers would connect to cvcs to checkout files

Note :

Checkout : taking code from repository to local machine

Push : sending code from local machine to server



Advantages :

1. Every developer in the project knows what changes that other developer had made to the code.
2. Administrator will have full control and he will give limited acess rights

Drawbacks:

1. There is possibility of Single point failure
2. All the data will be stored in the computer which is managed by the admin . what if hard disk fails (i.e sever repository ) called as spf single point failure
3. If server goes down then it would be difficult to collaborate with the server (or) Save changes

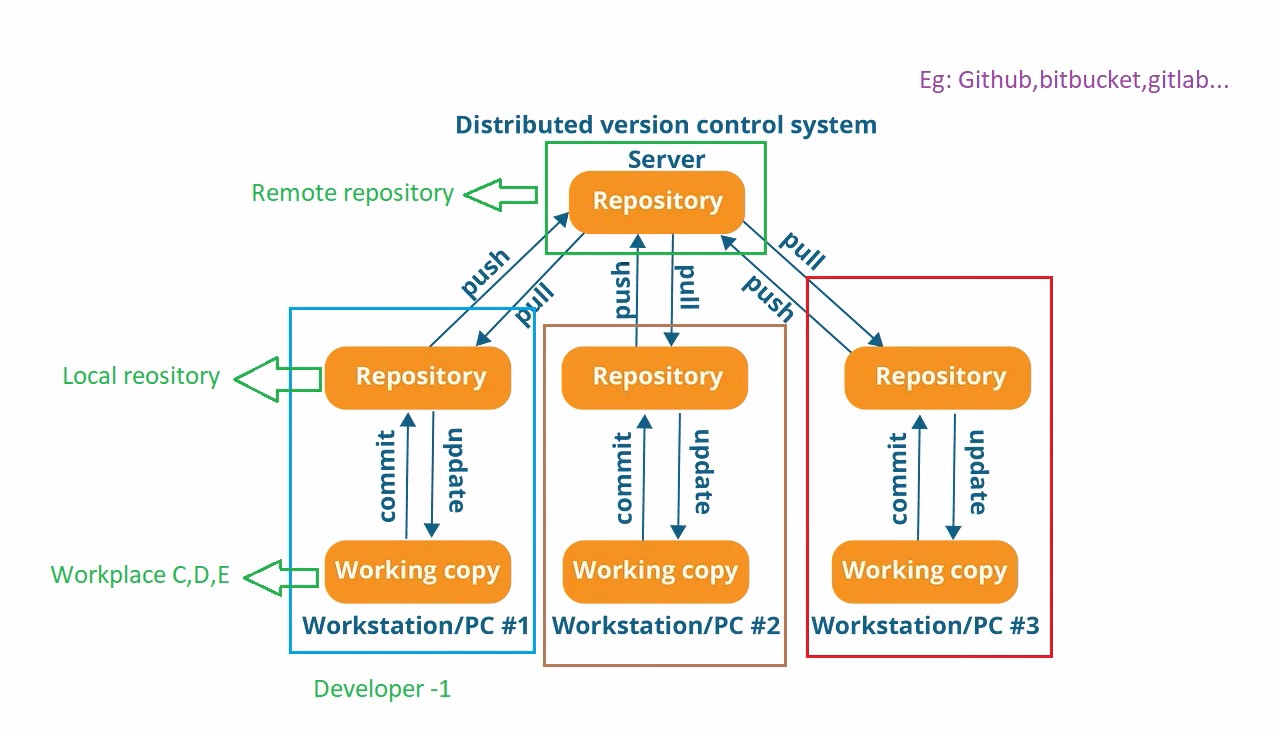
Note :

In Lvcs , Cvcs getting up complete history of changes is not possible

It is possible to only get the latest version but not the entire history

Push will not happen w.r.t to version , rater push will happen w.r.t changes

Distributed version control system



Eg: github, gitlab,bit bucket

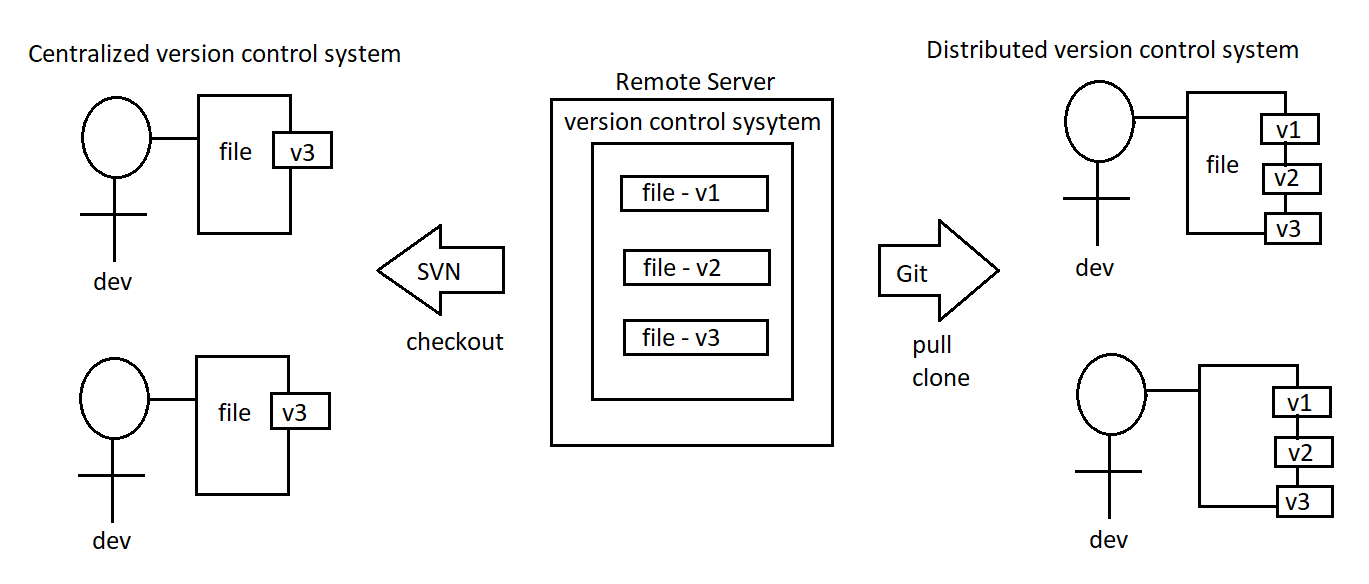
Eg: git, mercurial , baazar , Darcs

Developers will not only get latest version of files rather but also the compile history of files

Push will not only happen to latest snapshot of files but also they will push old files also

If the remote server is down , there is a local repository where it maintains copy of remote repository (which has same code as server)

If remote repository is down you can still make changes in the loca repository , whenever the server is up its is pushed into the remote repository.



There are 2 types of git software

1. Git server ( github )
2. Git client (git)

Git server

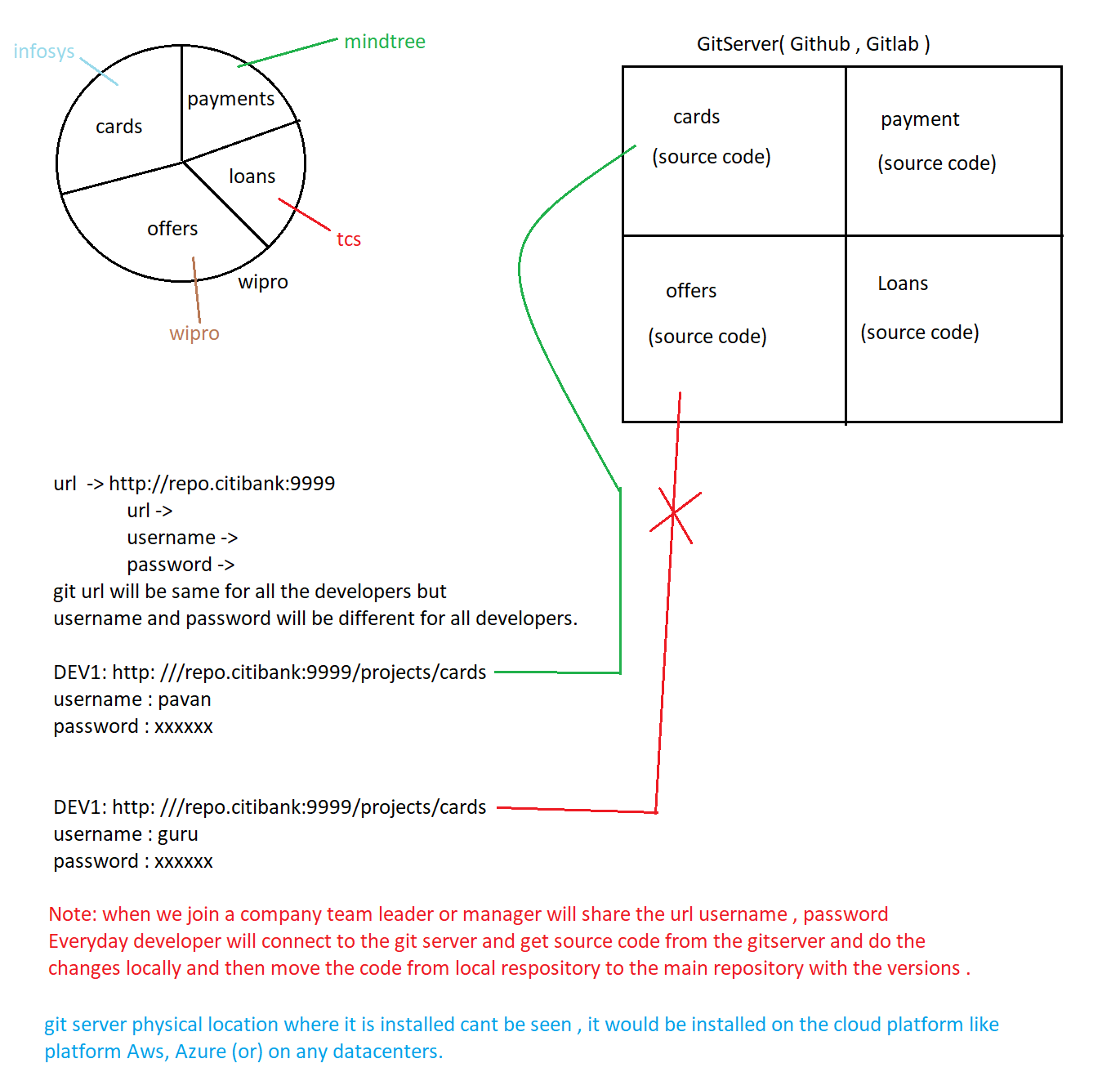
Its consists repositories

It stores large amount of source code around world

It is used to store / manage source code of the project

Some of git server tools are Eg: github , bitbucket, gitlab

Gihub is a server software where repositories are maintained .



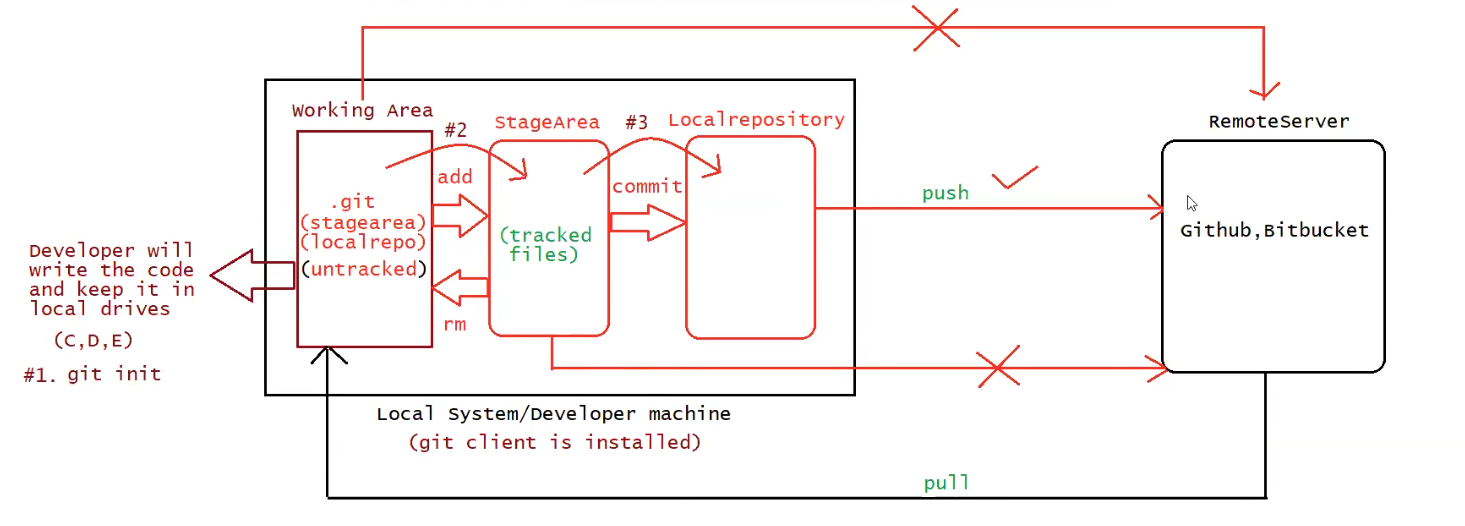
Gitclient

It is a tool which is used to connect with our git server .

We should type username , password in the gitclient and it will connect with our git

If we install git client we get some tools they are git bash , git gui, git cmd

Git Architecture :



There are 3 regions in the git architecture

1. Working area : place where developers maintain the source code
2. Stage area : once the code is ready then it will be added to the stage area (indication to git software )
3. Local repository : once the code is in stage area, we commit it to local repository with some standard message , from local repository we push it to main repository by providing url , username and password .

