L’utilisation standard de Git se passe comme suit :

**1.Working space**

On modifie des fichiers dans le répertoire de travail :

vim fichier.ext

**2.staging space**

Git calcule une empreinte( on dit un index) pour le fichier ,à l’aide de la fonction SHA-1, ce qui ajoute des instantanés de ce fichier dans la zone d’index :

git add fichier.ext

**3.repository**

On valide, por basculer les instantanés du fichier dans la base de données du répertoire Git :

commit -m message

un commit est une validation d'un fichier modifié puis indexé.

cette trace enregistrée se compose essentiellement de :

-un SHA-1 : une certaine identification du commit

-la date d'enregistrement

-le nom de l'auteur(eventuellement son email)

-une annotation sous forme d'un message decrivant la

modification en question

-un pointer qui contient la reference de l’instantané du

fichier(snapshot)

-un pointer vers le commit précedent(le commit init n’a pas de

parent)



La base de données ,qui est stockée dans le repository,est formée de plusieurs instantanés successifs dans le temps.On accède à un element via son index (SHA-1) : l'affichage de l'historique de tous les commits sur un (ou plusieurs) fichier(s),se fait avec git log.

On peut formatter cette liste de commits à nôtre gré :

$ git log --oneline --decorate

f30ab (HEAD, master, test) add feature #32 - ability to add new

34ac2 fixed bug #ch1328 - stack overflow under certain conditions

98ca9 initial commit of my project

-

une 'branch' est une sorte de repertoire ,oû toutes les modifications d'un fichier sont enregistrées.