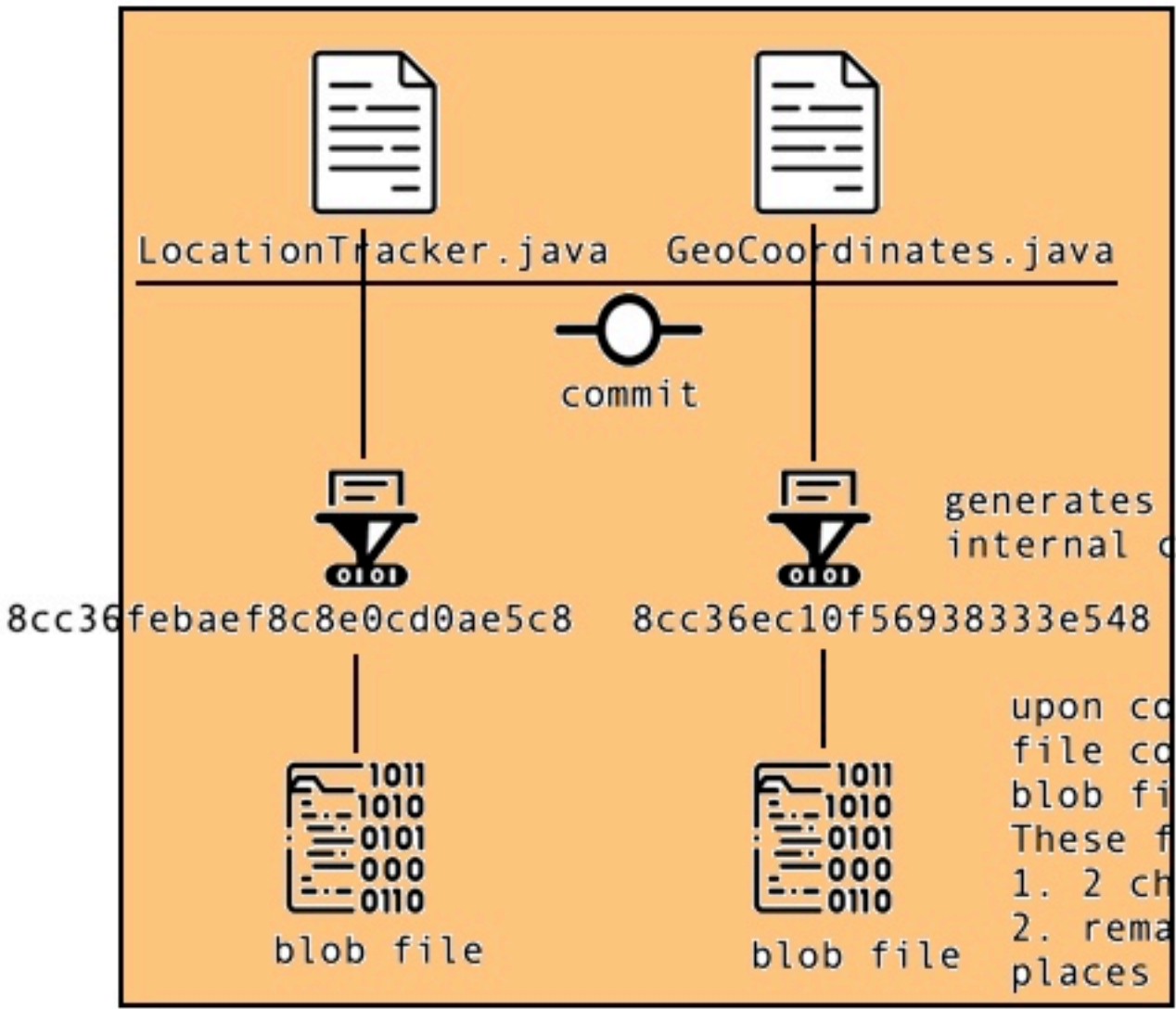


.git  
|-objects/  
|-logs/  
|-refs/  
|-hooks/  
|-HEAD  
|-config

#1.



generates the hashvalue for each file in the commit  
internal command: git hash-object filename

upon computing the hash value for each of the files based on the  
file contents. it encrypts and stores the contents of the file in  
blob files.  
These files are created using naming convention:  
1. 2 characters will be taken as directory name under .git/objects  
2. remaining hashvalue will be used for creating the filename and  
places the binary/ecrypted file contents inside it

#2.



To keep track of what files are part of an commit, the  
GIT creates an Tree file per each commit in which it  
stores commit hash and filename

hashvalue	filename
8cc36febaef8c8e0cd0ae5c8	LocationTracker.java
8cc36ec10f56938333e548	GeoCoordinates.java



8cc36ec10f2737848



The GIT has to store this Tree file also inside the  
repository. For this based on the contents of this file it  
generates hashvalue. converts the file contents into binary  
and encrypted blob file. Stores the file inside the .git/  
objects directory with  
1. first 2 characters of hashvalue as directory  
2. rest of the characters in hashvalue with filename

#3



commit info

Per each commit we make in the  
repository, the GIT creates an  
Commit Info file in which it holds  
the details of the commit like  
1. pointer pointing the hashvalue  
of the Tree file  
2. author info  
3. committer info

Tree 8cc36ec10f2737848  
Author tech.sriman@gmail.com  
Commier tech.sriman@gmail.com



8cc36ec10f27u2y444



.git/objects

It stores the commit info file  
also inside the repository  
by generating the hashvalue  
based on the commit info and  
stores as an blob file within  
the .git/objects directory of  
the repository same as the way  
we described earlier

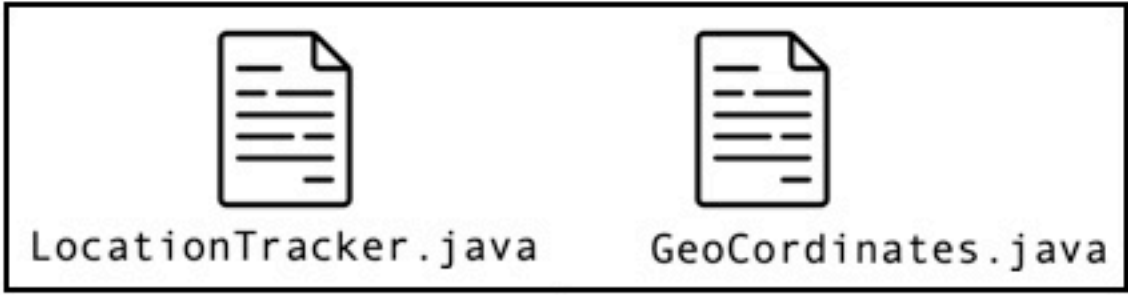
To keep track of each branch and at what commit these branches  
are at. git creates one file per branch with BRANCHNAME under  
.git

|-refs  
  |-heads  
    |-master  
    |-feature1  
    |-feature2

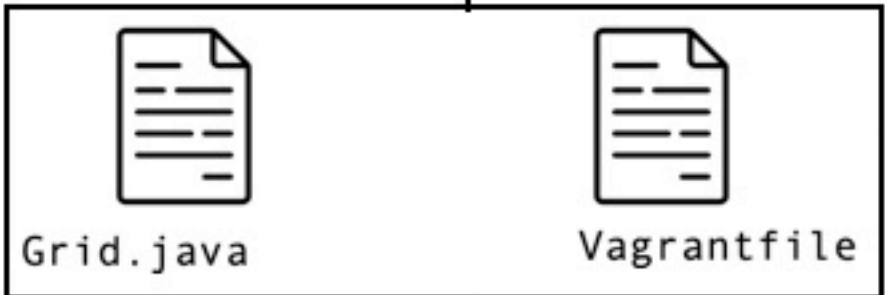


BRANCHNAME

in these files the git stores the COMMIT INFO file hash pointing  
to the COMMIT at which the branch exists.



C1



C2

h1 = LT.java  
h2 = GC.java  
h3 = GR.java  
h4 = VF



C3

h5 = LT.java  
h2 = GC.java  
h3 = GR.java  
h4 = VF



C4

feature1

```
.git/logs/refs/heads/master
|-00000000000 C1
|-C1 C2
|-C2 C3
|-C3 C4
```

```
.git
|-refs
  |-heads
    |-master = C4
    |-feature1 = C3
```



GIT LOG

.git/logs/refs/heads/master  
all the commits in the repository are kept/stores inside  
the Log file

it keeps track of all the commits in the repository and  
their order in which the commits are made by placing  
parent -> current

For the initial commit/root commit of the repository the  
parent hashvalue - 0000000000000000  
there after for each commit, the previous commit hash will  
be its parent.

git clone -b feature1 repoURL  
git clone -b master repoURL