**DevOps Demo**

Developpers ---🡪 Test ----🡪 Released to production (IT Operations)

- Devops is combination of set of tools ,practices which automates the process between dev and ops

BEFORE Devops

* There was no full automation, this leads to less productivity
* There was no communication or collaboration between Dev and Ops team. (no communication means process will have some struggles)
* Releases cycles used to about 6 months, time to market is slow (customers has to wait till the feature gets to the prod)
* There was no continuous integration, which gives late feedback and that causes complete rework.

Devops:

* Fully automated
* Dev and ops team must work as a single team.
* We follow Agile, i.e shot releases (1-2 Weeks)
* We follow continuous integration ( developpers commits changes sevral times a day and it is automatically built and tested ,if any bugs found an automated email is sent.)
* We follow continuous Deployments. (
* We follow comtinuous delivery. (some one needs to be approved ,manual Aprroval is required before deploying to production) **IQ**
* Configurtion Management( For Example Ansible 🡪ex: patching )
* Microservices,Dockers,Kubernetes. ( fb divided in to smaller services earlier it was big app)
* Monitoring
* Log management
* Etc ..

17-10-19

Any operation

I am deleting a branch in local and I am pushing that changes t0 remote

# git push -d origin task-5

**IQ** )Git Merging strategies

Git has different strategies :

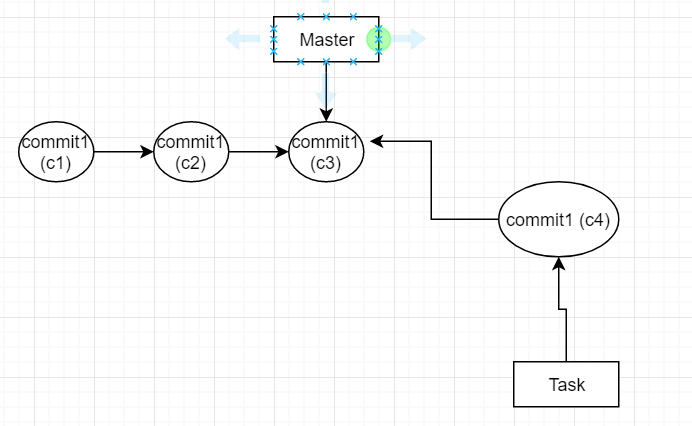
Git depending on the context uses one of the following meging strategies

* Fast forward strategies
* Recursive or 3 way merge
* Rebase

**Fast forward Merge :**

**Before FFM:**

%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22Master%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22240%22%20y%3D%2240%22%20width%3D%2270%22%20height%3D%2230%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E



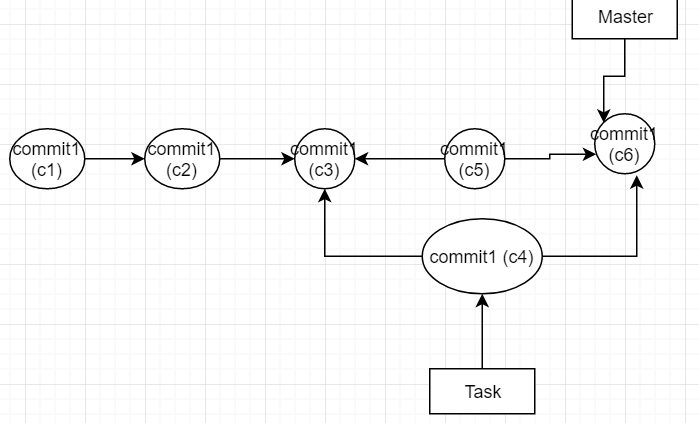
**After t**ask branch is created from master at c3 there are no new commits in the master In this case if we merge git uses fast forward merge

After ffwd merge

Recursive merge:

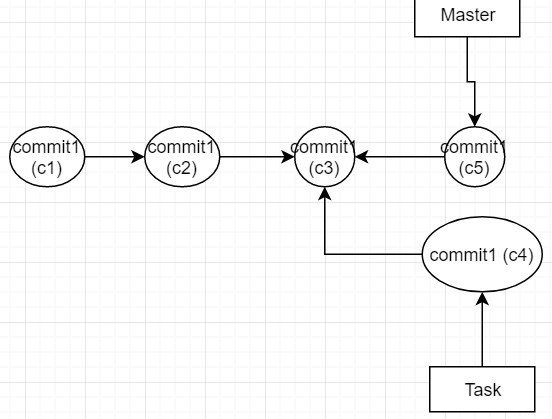
After task branch is created from master at c3, master has new commits so git can’t use Fast forward, the option left is recursive merge. It creates a new commit

After recursive :

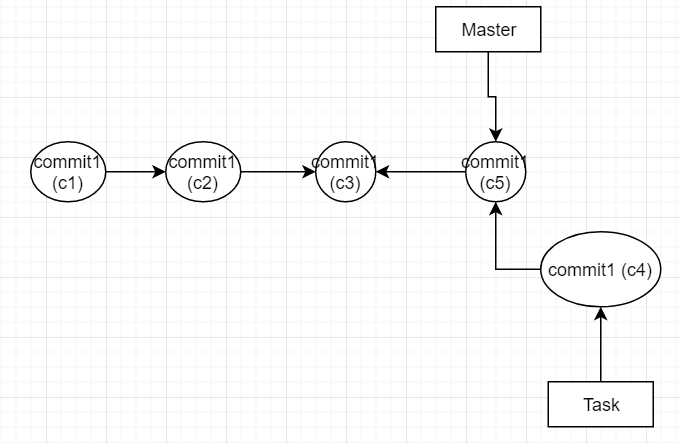


Rebase:

Before rebase



After rebase



Task : check practically how to use rebase.

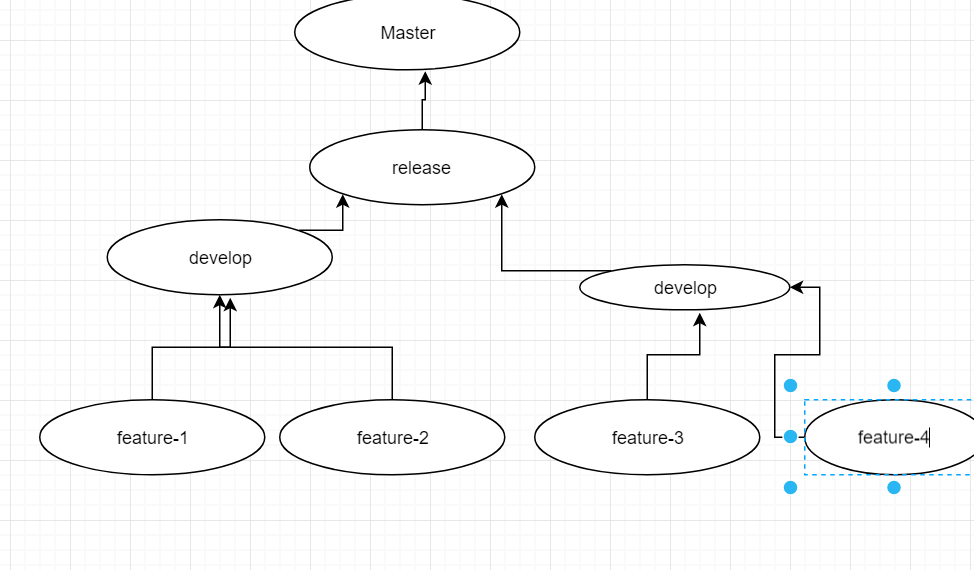
**IQ**)What is Git init :

Converts local folder to Git repository

**IQ)** what is bare repository in git:

It’s a repository which doesnot contain working area and staging area (i.e we can’t do changes to bare repository)

**IQ) : which branching strategy are you using in project : imp**



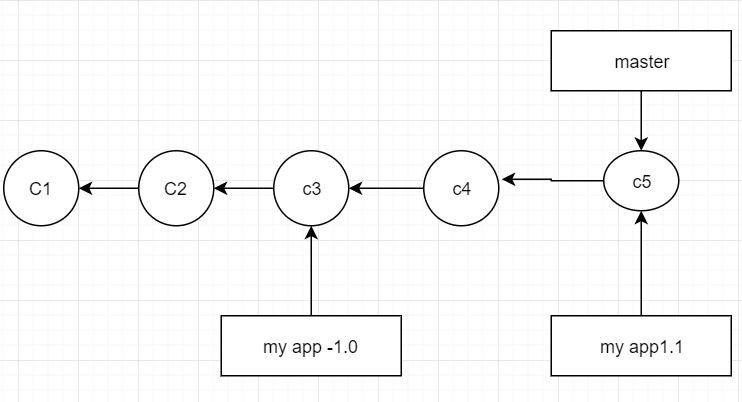
There will be development teams with different names.

**Cherry pick:**

Cherry pick pics only specific commit from the other branch and integrates with current branch

**Tag in Git :**

Tag is a light weight pointer which is used for marking important comits in the history ,it is typically used for versioning releases



[ same branch commands just to replace }

Command : # git tag myapp-1.0.0 -m ‘releaseone’ ( git tag tagname commit message

# git push origin myapp-1.0.0 ( to push to remote)

# git push origin myapp-1.0.0 -m ‘release 0.1’ 574c4b9 ( for tagging from particular point )

# git tag -d myapp-1.0.0 ( to delete a tag)

**IQ)What is difference between Branch and tag ?**

We create new branch to implement new features and fix defects

We use tags to mark new releases

**IQ)What is Git stash ?**

**-------------------------------------------------------------------------------------------------------------**