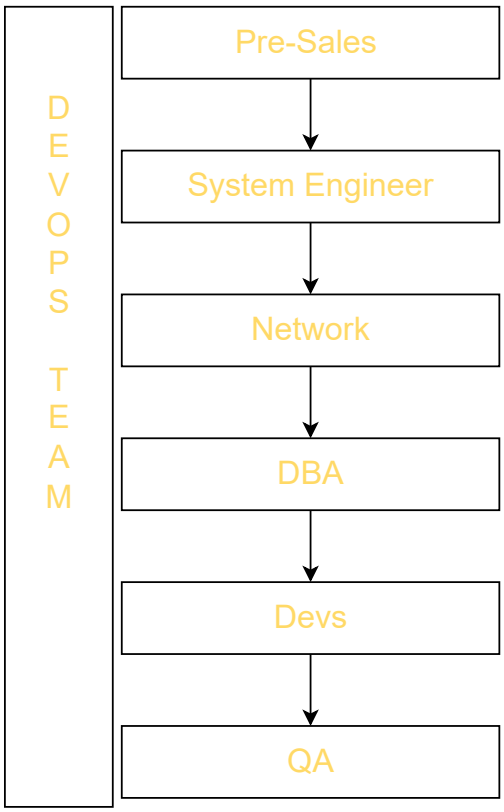
What is Devops?

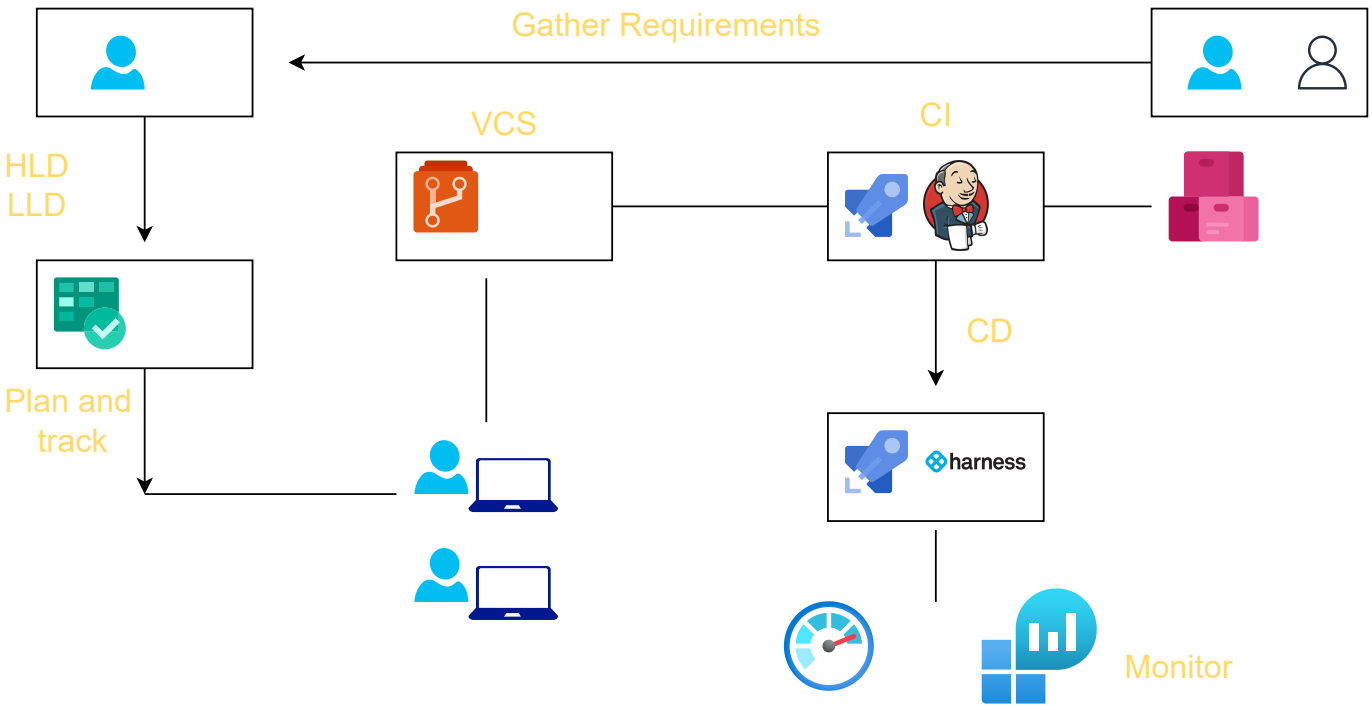
1. faster releases
2. Quality and efficiently
3. Continuous Improvement

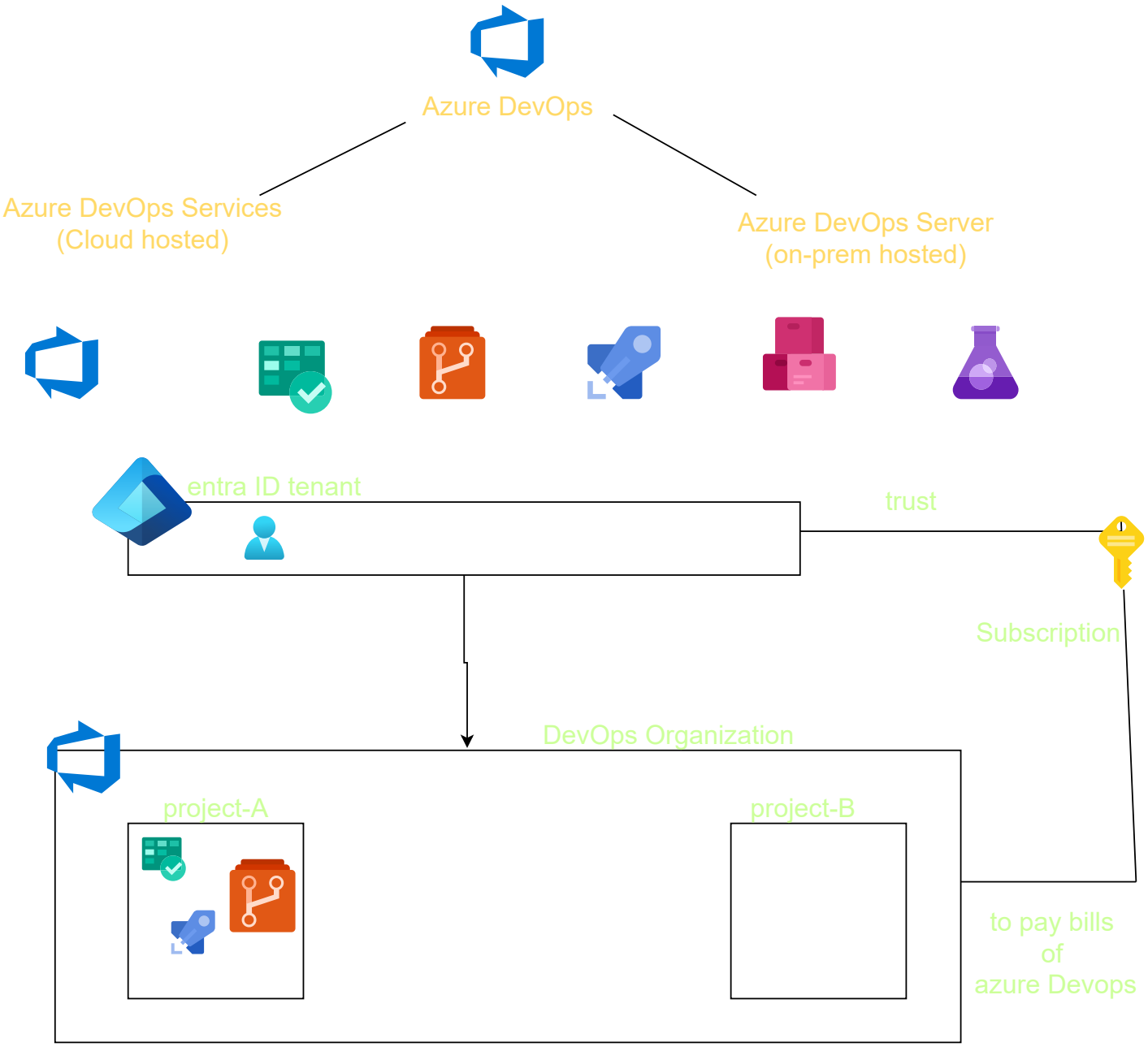
People, Process and Products

We cannot buy or Sell DevOps
build it ....



SDLC (software Dev Lifecycle):





Access Level in DevOps:
What can you see

All users in Azure DevOps belong to one or more default security groups.
Security groups get assigned
permissions that either Allow or Deny access to features or tasks.

Product management is an organizational function that guides every step of a product's lifecycle — from development to positioning and pricing — by focusing on the product and its customers first and foremost.

Running the scrum calls, creating those user stories ==> Scrum Masters

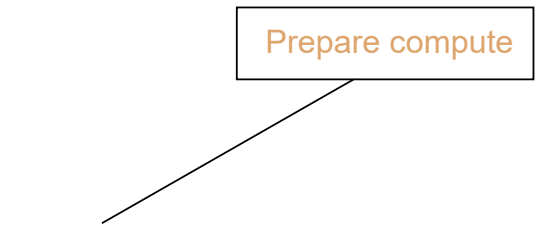
- Use **Agile** when you need flexibility and can afford to adapt to change constantly.
- Use **Scrum** when you need a more structured, time-boxed approach with specific roles and regular feedback loops.

Anything you create inside azuree boards ==> work item
example: task, epic, bug etc

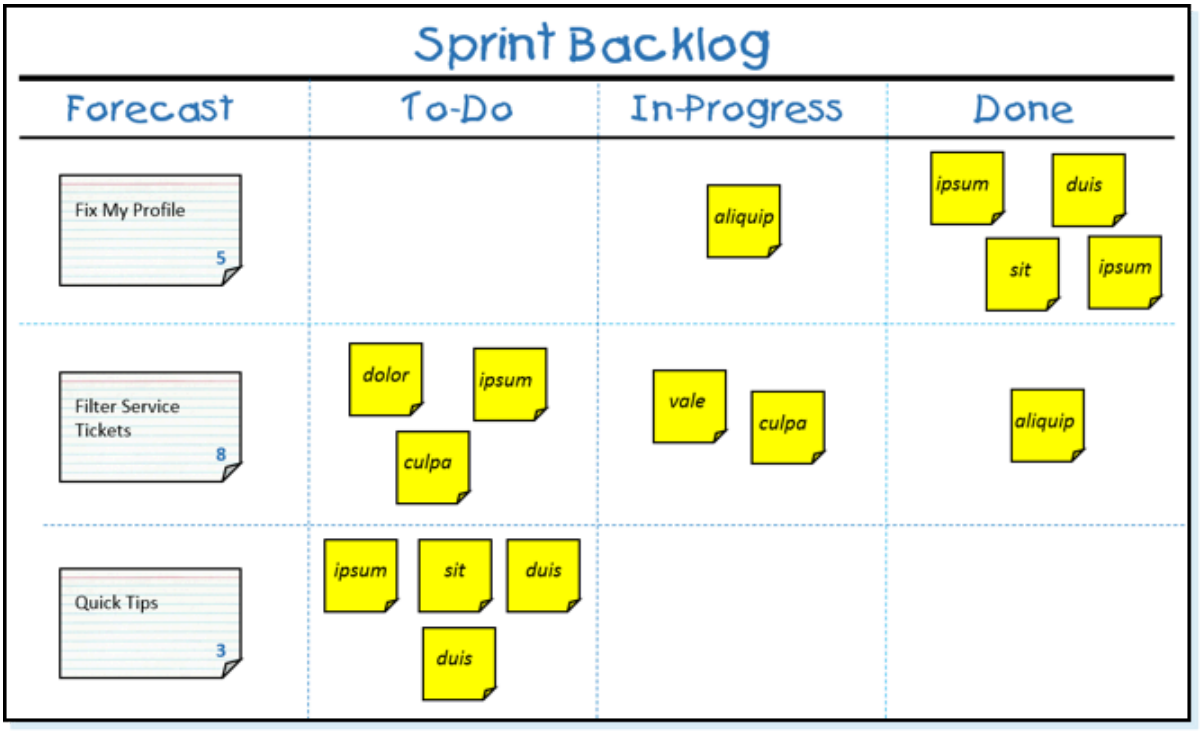
wrong work item process type. No problem. You can change it after creation.

Select Work item Process type
a, change it after creation (possible)

Game Development:
1. Game Character
2. Game Story
3. Create Animation
4. prepare compute
5. Prepare Networking database
6.



What is a product backlog? A product backlog is a prioritized list of work for the development team that is derived from the product roadmap and its requirements. The most important items are shown at the top of the product backlog so the team knows what to deliver first

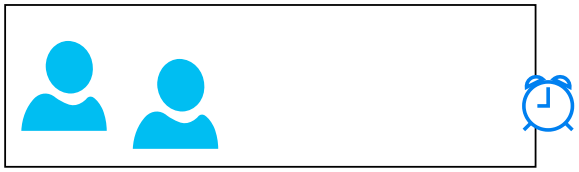


A kanban board is an agile project management tool designed to help visualize work, limit work-in-progress, and maximize efficiency (or flow). It can help both agile and DevOps teams establish order in their daily work.

Collection of work item is called as Area

Task if you have to perform in your entire graduation
Product backlogs

Team - group of people having common work items



sprint backlogs

sprint

Queries - filter work items

A burndown chart is a graph that represents the work left to do versus the time it takes to complete it. It can be especially useful for teams working in sprints, as it can effectively show whether your deadlines are able to be met along the way.

https://learn.microsoft.com/en-us/azure/devops/boards/work-items/guidance/media/alm_pt_wits_testexperience.png?view=azure-devops

Frontend - ReactJS, ExpressJS, HTML, CSS, Kotlin, Swift

Backend - Python, Java, Node

Database: MongoDB, MySQL

Terraform, Ansible, Bash

Dockerfiles, K8s manifests

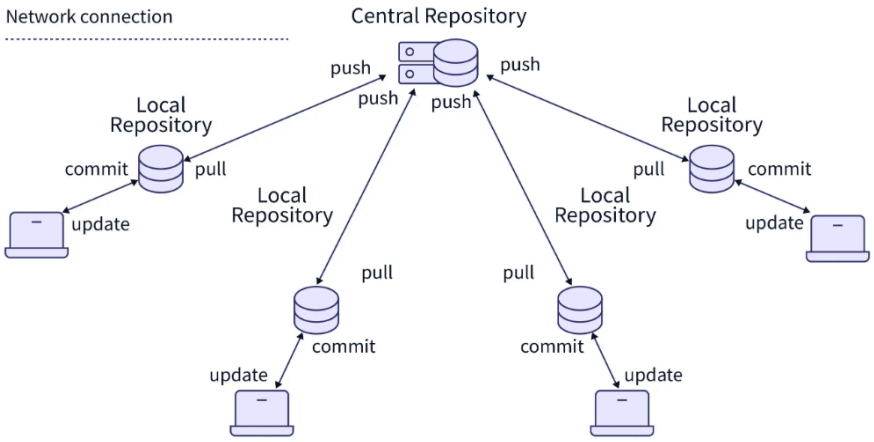
1. Central Place
2. avoid conflicts
3. version control

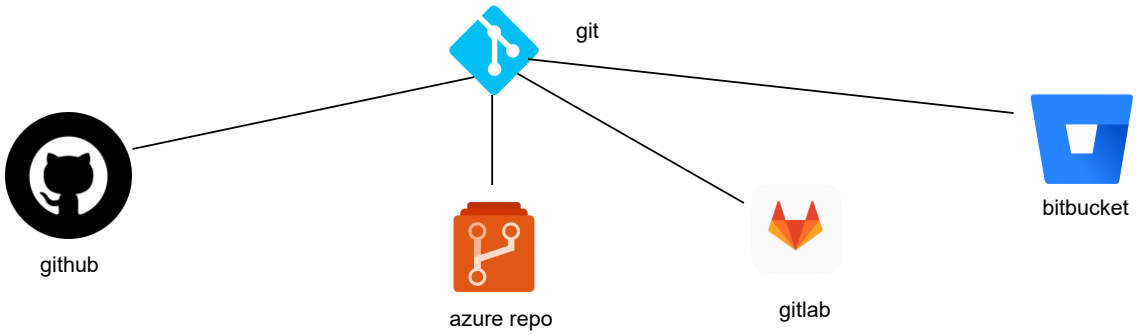
→ Version control system (VCS)
aka
SCM (source code management)

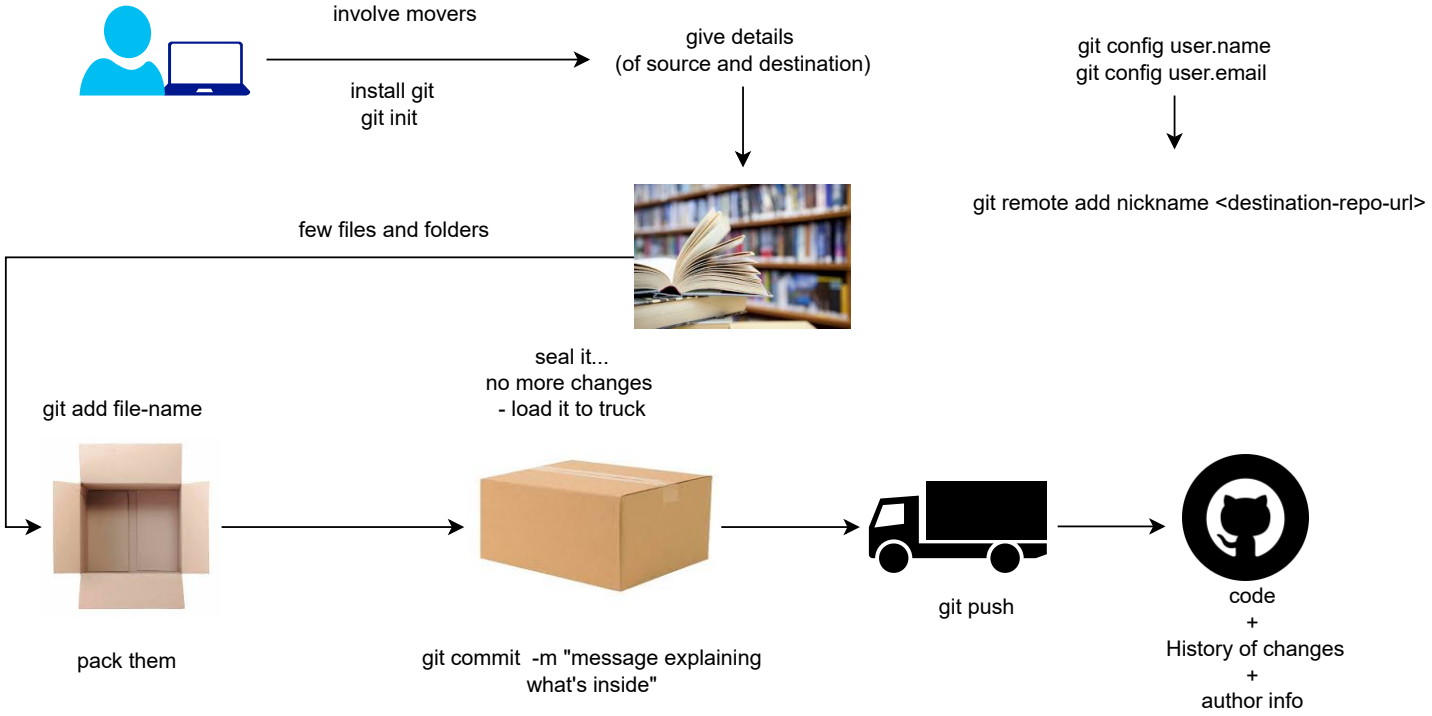
↓

git : distributed VCS

A distributed version control system (DVCS) brings a local copy of the complete repository to every team member's computer, so they can commit, branch, and merge locally. The server doesn't have to store a physical file for each branch — it just needs the differences between each commit.

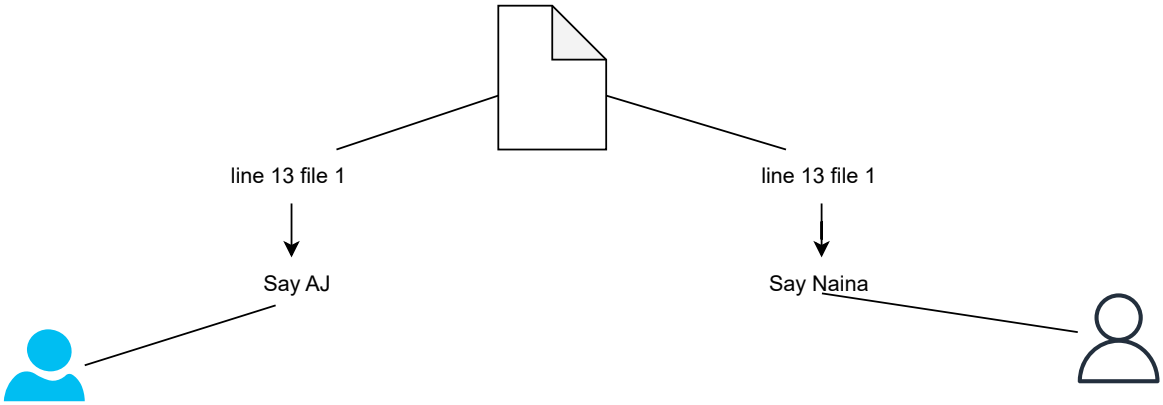

Network connection
Central Repository
push
push
push
push
Local Repository
Local Repository
Local Repository
Local Repository
commit
pull
pull
commit
update
update
pull
pull
commit
commit
update
update

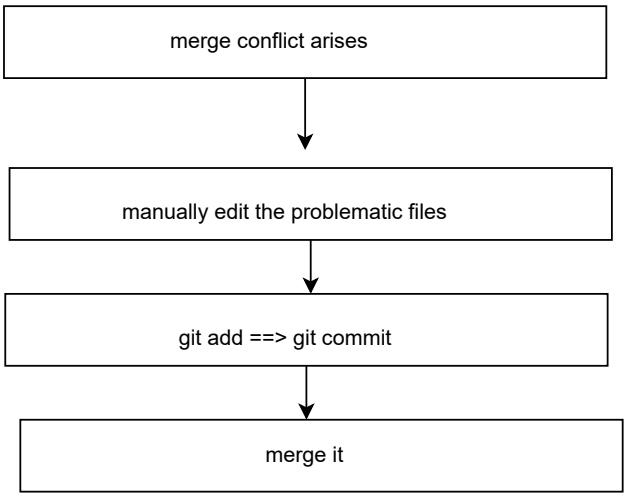

git
github
azure repo
gitlab
bitbucket


involve movers
install git
git init
give details
(of source and destination)
git config user.name
git config user.email
git remote add nickname <destination-repo-url>
few files and folders
seal it...
no more changes
- load it to truck
git add file-name
pack them
git commit -m "message explaining what's inside"
git push
code
+
History of changes
+
author info

Trying to combine (merge) code of two different branches

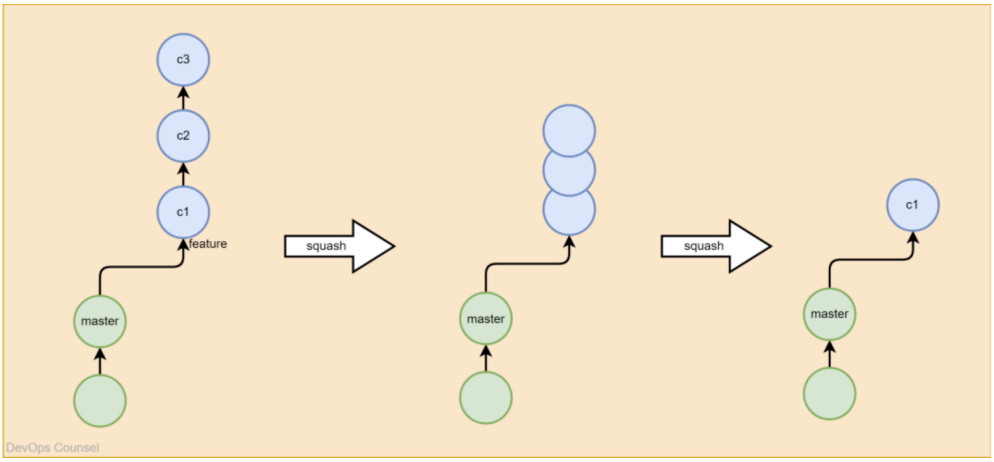
two different people try to make change on the same line of the same file. ==> Merge Conflicts



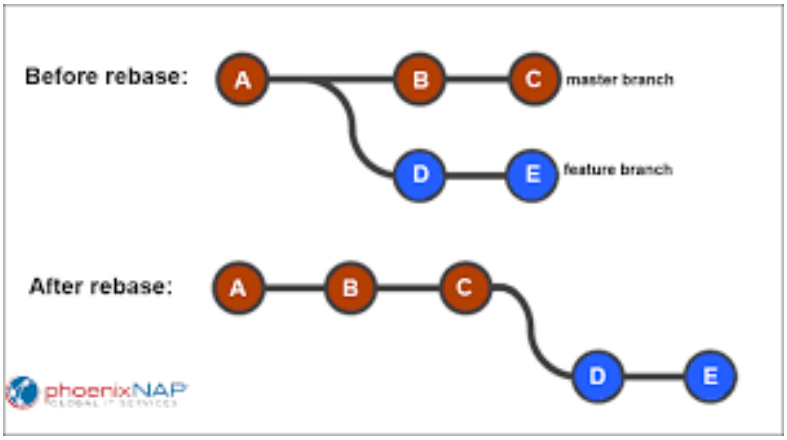



```
<<<<<<< chnage2-user2
   bucket = "AJ-the-user-tfstate"
   key    = "test/test.tfstate"
   region = "us-east-1"
   # use_lockfile = true
   dynamodb_table = "mumbai-dynamodb"
=======
   bucket = "tokyo-dev-env-tfstate"
   key    = "test/tokyo.tfstate"
   region = "ap-northeast-1"
   use_lockfile = true
   dynamodb_table = "tokyo-dynamodb-table"
>>>>>>> chnage1-user1
```

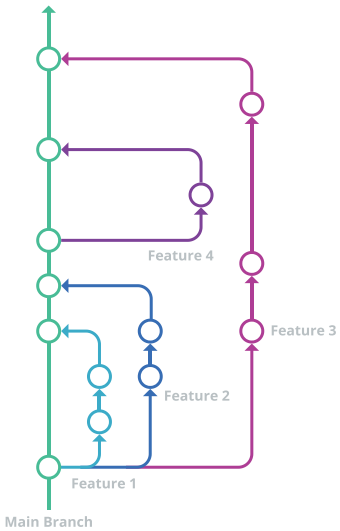
git squash

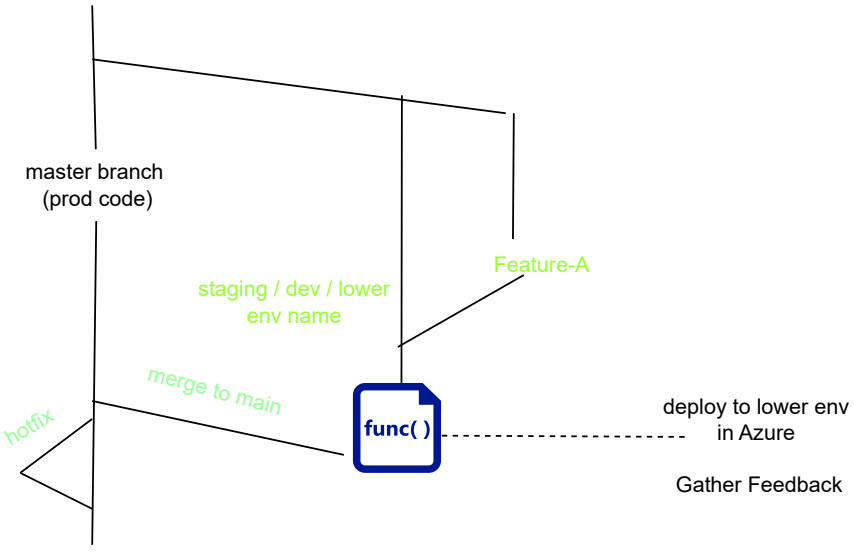


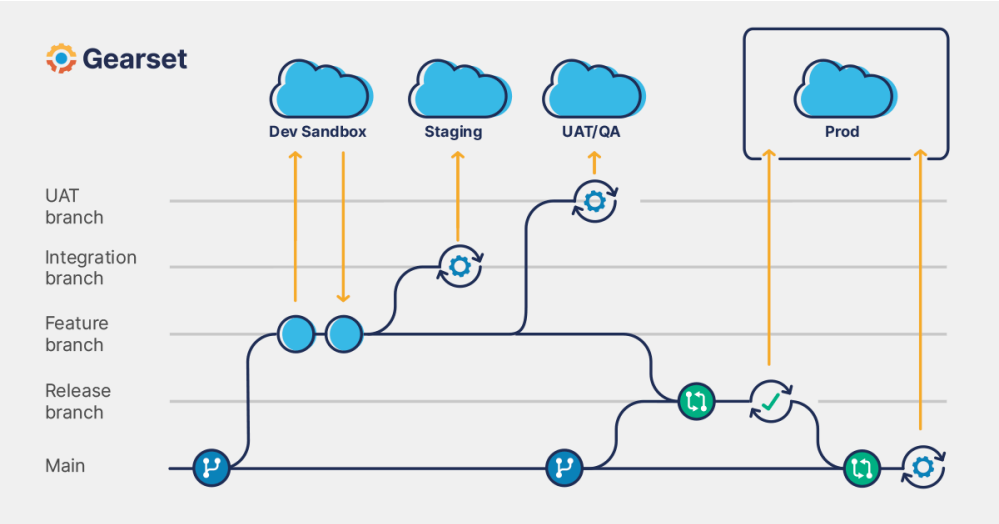
git rebase



git merge commit







Pipeline

Manual - time and efforts

When you code :
1. No more manual efforts
2. Idempotence : Produce same result every time
3. Easy to share and collaborate

→ DO NOT REPEAT YOURSELF so Create A TEMPLATE

↓

specific event occurs ==> run this template

translator in between, correct? Who translates my English into the binary for computers, you understand?

↓

build Process (Java, CS, Golang)

↓

exe

⇩

deploy it to non prod ==> test ==> get approval

↓

Deploy to Prod

Deploy to Env

build ==> test ==> Produce Output (aka Artifacts)

Build pipeline

Source Code → download it → code downloaded
Task-1
task-2
→ Output

CPU, Memory etc to RUN pipeline tasks → Pipeline agent → Collection of such agents => agent Pool

```
##[error]No hosted parallelism has been purchased or granted. To request a free parallelism grant, please fill out the following form https://aka.ms/azpipelines-parallelism-request
```

Attach billing and purchase paid parallel job fir pipeline