

Agile Development :

- #Continuous availability of increments of product

- #Variety of activity needs to be performed to make it available at various places

- #These activity must implemented and planned so that process shall remain smooth / time-boxed (agile)

- ==>Activity of building the product increment (

Continuous Integration :Continuous Build(practical)->Continuous Deployment->Continuous Deployment

- #Continuous Integration : process of Automated build and Automated Test (in sync)

- ==> helps to to detect error quickly...

JENKINS...

- #After every commit (or any custom) in S/C an auto build is triggered : then auto deployed on test server

- #if test result shows some error : dev only have to check the last commit

- #automated system can configured to provide relevant and timely feedback to the team

JENKINS :

need a setup on our machine

#Desktop solution : works in integration with online jenkins servers

1. require create user accounts (atleast 1 user account (admin))

2. Jenkins will run on local machine (by default run on 8080)

#launch the local jenkins server from browser (<http://localhost:8080>)

#ask for login credentials

JENKINS have 2 components:

Jenkins Master

Jenkins Slave

Master :

#Schedule the Build Job

#Dispatches job to slave component for execution

#Monitor the slave and record the build result

Slave/Slave Agent

#Executes the build job (separate programs)

==>master launches and registers a slave agents when a build job is required to be performed

Node : Jenkins env: machine setup for different jenkins activities

can master/slave

Complete jenkins env spreaded across multiple node (local/remote/cloud)

Jenkins Dashboard:

- #create a new build job

- #manage users

- #manage profile

- #option for complete config of jenkins

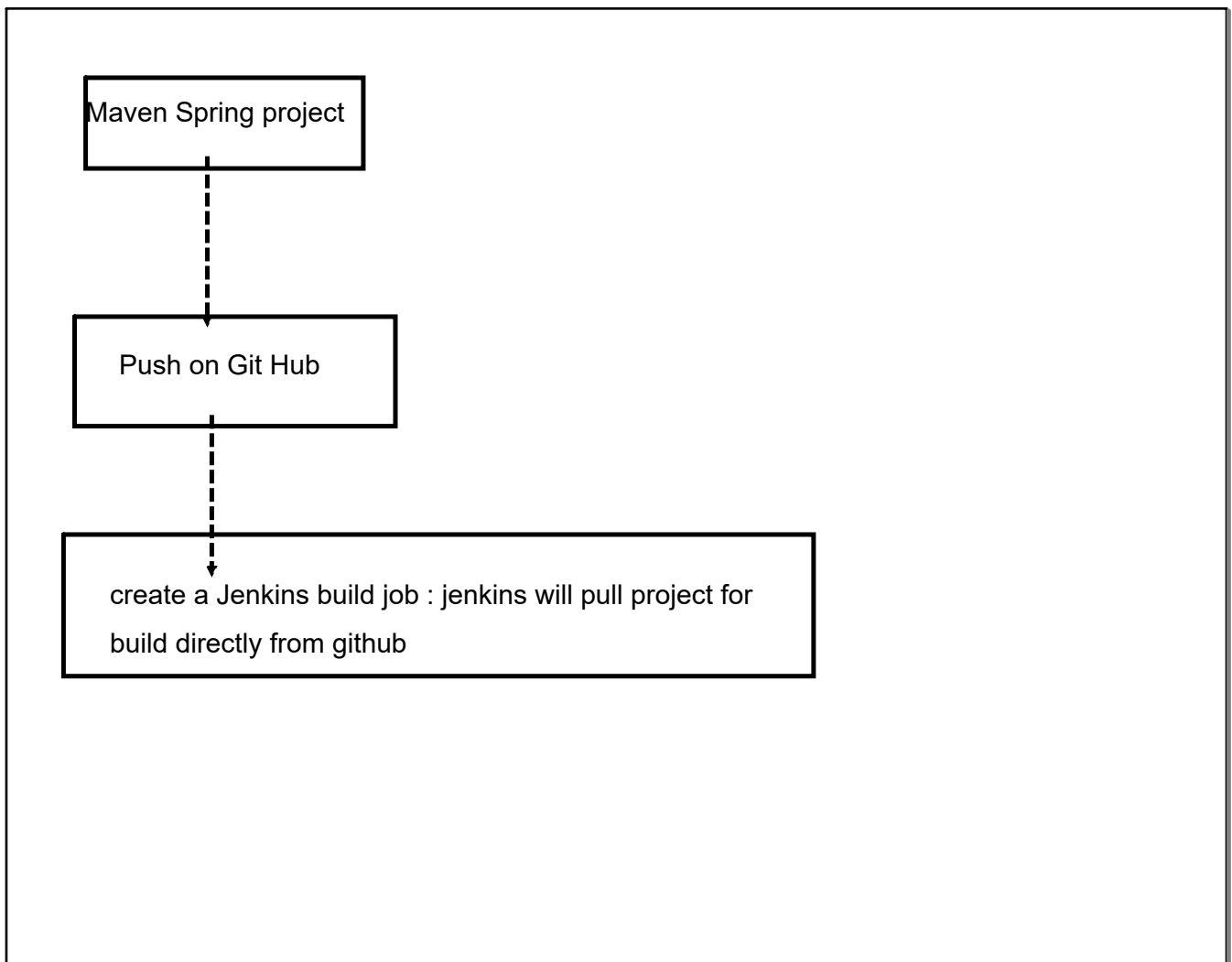
- #can create multiple view for quick access and easy navigation/info

==>Before creating build need to configure the env for build

eg:

- need to create build job for a maven spring project

- #need to configure jenkins for java support,maven support



Initiated local git repo for project : git init

adding to staging area : git add .

committed the changes for final artifact : git commit -m "---"

#create git hub project (empty)

#connect the local git repo to remote link :

git remote add origin <https://github.com/Navin-Gupta/spring-unit-test-project.git>

push project on git hub

git push -u origin master

jenkins : Maven life-cycle :

Jenkins able to follow diff stages of maven life cycle

1. Validate : project structure is validated and checked for all necessary info
2. Compile : Compile the S/C
3. Test : Test the compiled code using a suitable unit Testing Framework
4. Package : Take up the compiled code and package (jar/war....)
5. Install : Jenkins also manages a local-repository to store build project..

Install package in Local Repo : can be used locally

6. Deploy : Copy the final package to remote repository (testing server/production/other developer systems) : configure for creating docker image as well

#need to configure jenkins build to specify the phase we want

#can be defined upto what phase build should take place

eg:

package phase : Validate,Compile,Test,Package

Periodic build pattern :

* * * * * :Alaises can be used

1. minute (0-59)
- 2.hour (0-23)
3. day of month (1-31)
4. month (1-12)
- 5.day of week (0-6)

eg:

TO auto trigger build after every Hour

H * * * *

Build every 20 min

H/20 * * * *

Build every 20 min (8am -6pm), (MON-FRI)

H/20 8-18 * * 1-5