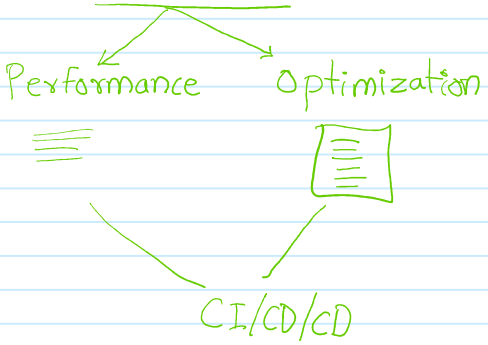
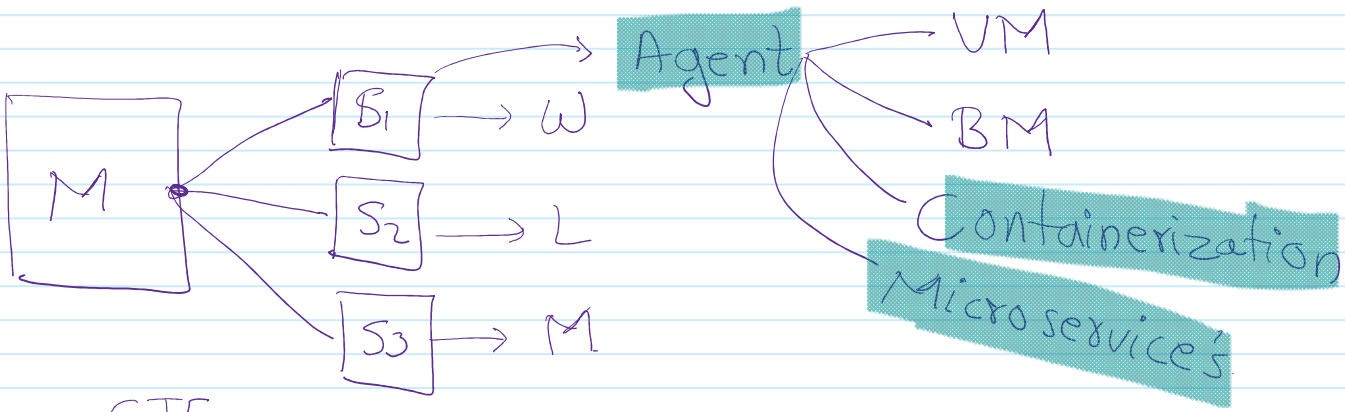
32.05Aug.Jenkins Best Practices & Integrations(JavaApp) Part1

5 August 2023 04:46 PM

- Jenkins Master Slave Architecture.





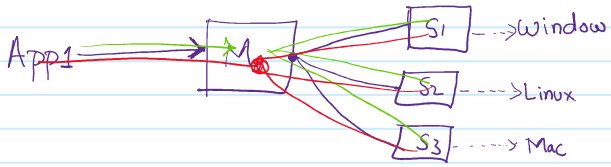
CJE

CCJE = Certified
Cloud Bees
Jenkins
Engineer.

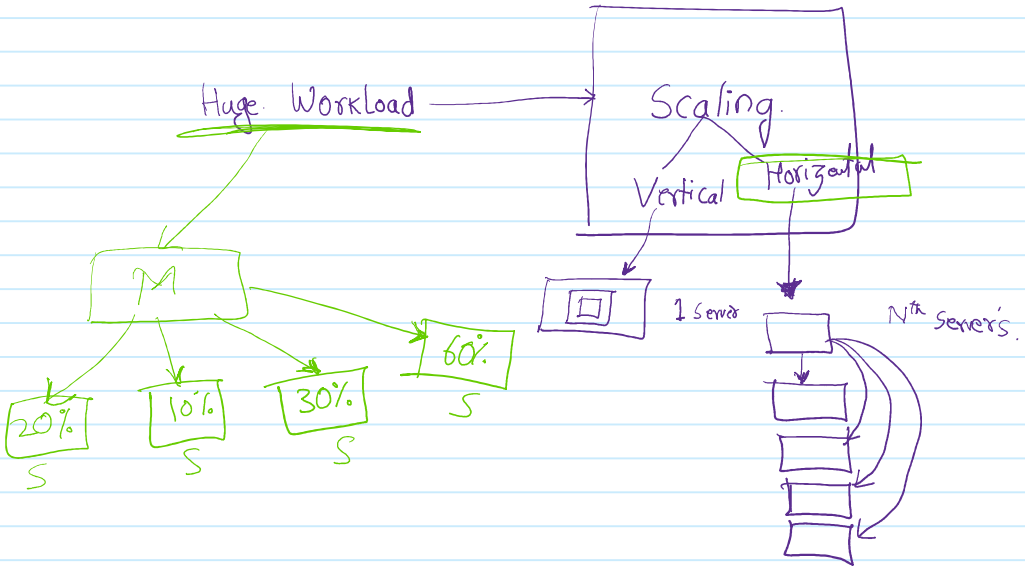
Challenges

1] Parallel Build & Distributed Build.

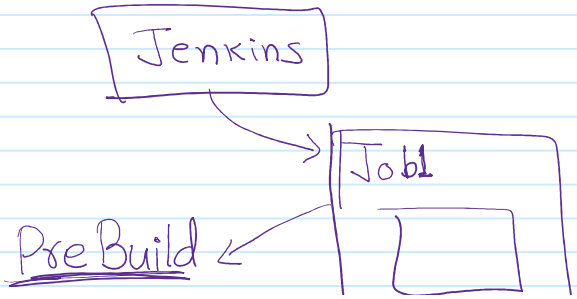
- Parallel Execution
  - Splitting the build Processes Across multiple ~~executors~~/agent's

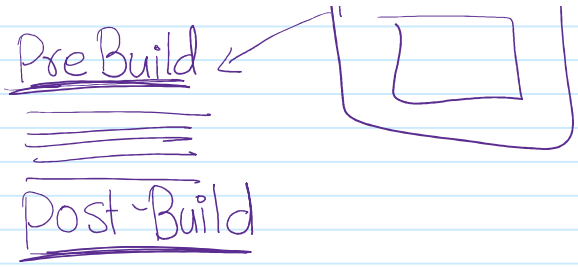


2] Distributed Build



3] Workspace Cleanup





4] Caching Dependencies

- Cache Memory

5] Agent Scaling & Dynamic Provisioning

- EC2 → HA/FT
- Docker → K8s

6] Pipeline Optimization

- Remove Unnecessary Step's
- Structure
  - Conditional
  - Parallel Stages

7] Artifact Management

8] Avoid frequent things

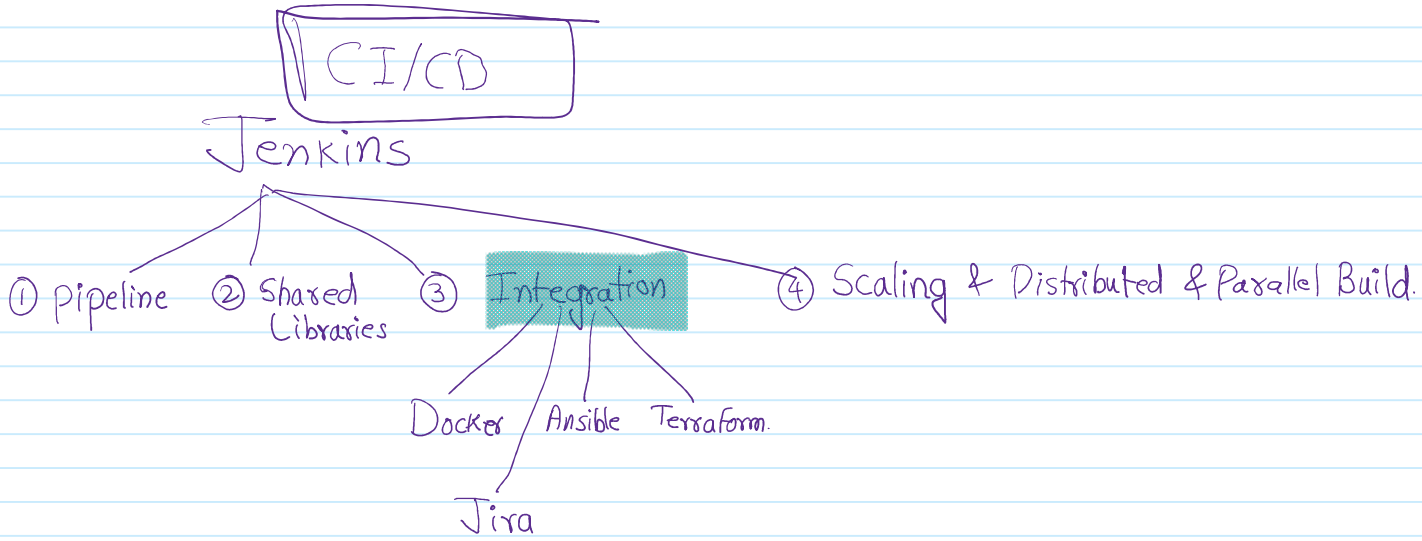
9] Jenkins Master-Slave.

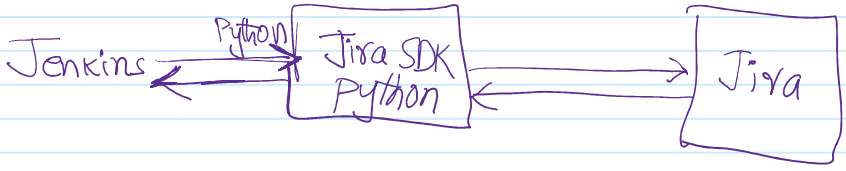
- JVM Config
- Head Size
- Executors
- Agents

- JVM Config
- Heap Size
- Executors
- Agents

10] Scheduled Maintenance & Upgradation

- Deprecated
- Plugins
- Tools
- DataBase





① Github ⟶ Jenkins

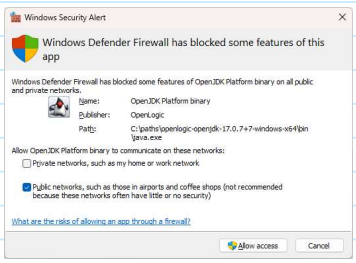
② Github ⟶ Jenkins ⟶ Java. Console App.

③ Git ⟶ Jen ⟶ Java ⟶ Mvn.

④ Git ⟶ Jen ⟶ Java ⟶ Mvn. ⟶ Junit

TestNG. ⟶ Selenium.

1. VCS to CI Server

1. Git hub Account , Repository
   a. git clone https://github.com/NubeEra-Samples/Java-Welcome-HelloWorld.git
2. Java(Installed, Configure Paths)
   - java -version
     openjdk version "17.0.7" 2023-04-18
     OpenJDK Runtime Environment OpenLogic-OpenJDK (build 17.0.7+7-adhoc..jdk17u)
     OpenJDK 64-Bit Server VM OpenLogic-OpenJDK (build 17.0.7+7-adhoc..jdk17u, mixed mode, sharing)
   - javac -version
     javac 17.0.7
3. Jenkins
   a. Install Jenkins(Single/Master-Slave)
      i. Linux/Windows
   b. Start Single/Multiple Servers(Master Slave Architecture)
      - Windows # java -jar jenkins.war --httpPort=8081
      - Unix(Linux/Mac) # systemctl restart jenkins

   c. Configure Credentials
      i. Type : Username & password
      ii. Username : GitHub-User-Name
      iii. Password/Token : Token
   d. Install Plugins
      i. GitHub Integration
   e. Create Pipeline