Jenkins

Machine Learning Architects Basel

Bassem Ben Hamed

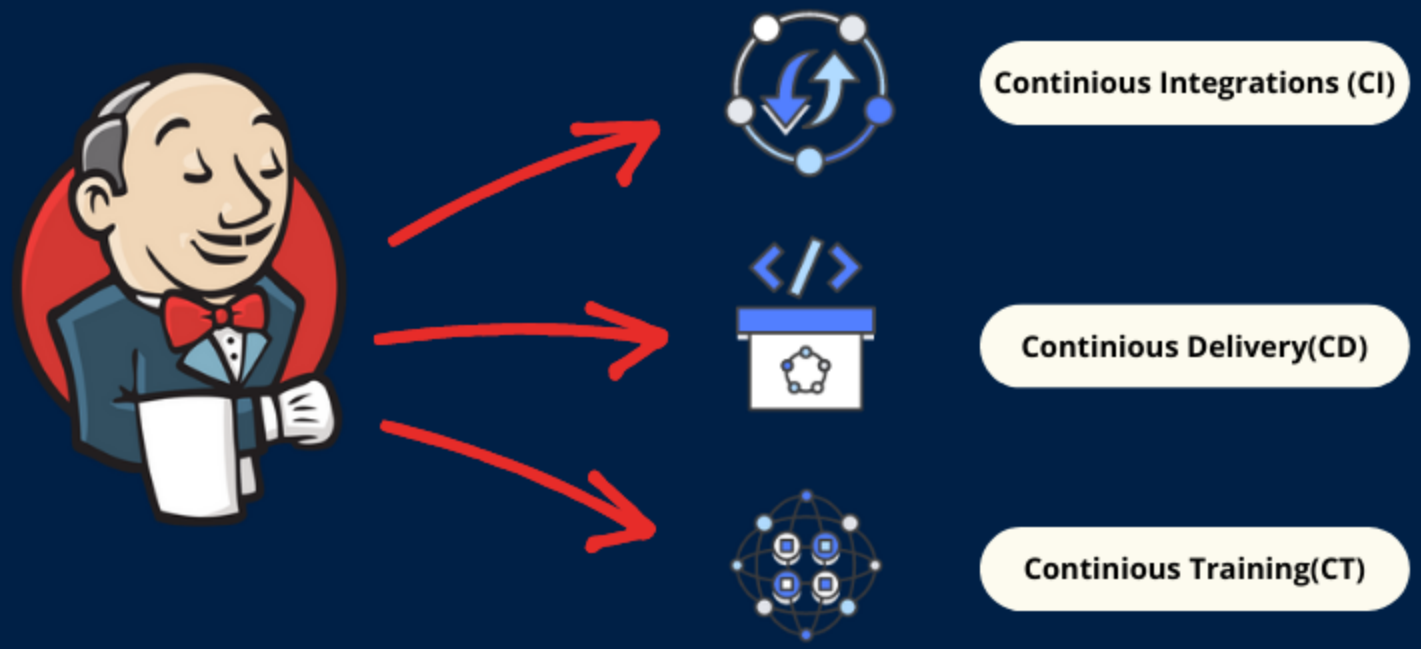
December 2022

ML ARCHITECTS
BASEL

Agenda

- What is Jenkins
- Before and After Jenkins
- Jenkins' Master and Slave Architecture
- Jenkinsfile Pipeline
- Understanding The Jenkinsfile Pipeline

What is Jenkins



Before and After Jenkins

Before Jenkins	After Jenkins
The entire source code was built and then tested. Locating and fixing bugs in the event of build and test failure was difficult and time-consuming, which in turn slows the software delivery process.	Every commit made in the source code is built and tested. So, instead of checking the entire source code developers only need to focus on a particular commit. This leads to frequent new software releases.
Developers have to wait for test results	Developers know the test result of every commit made in the source code on the run.
The whole process is manual	You only need to commit changes to the source code and Jenkins will automate the rest of the process for you.

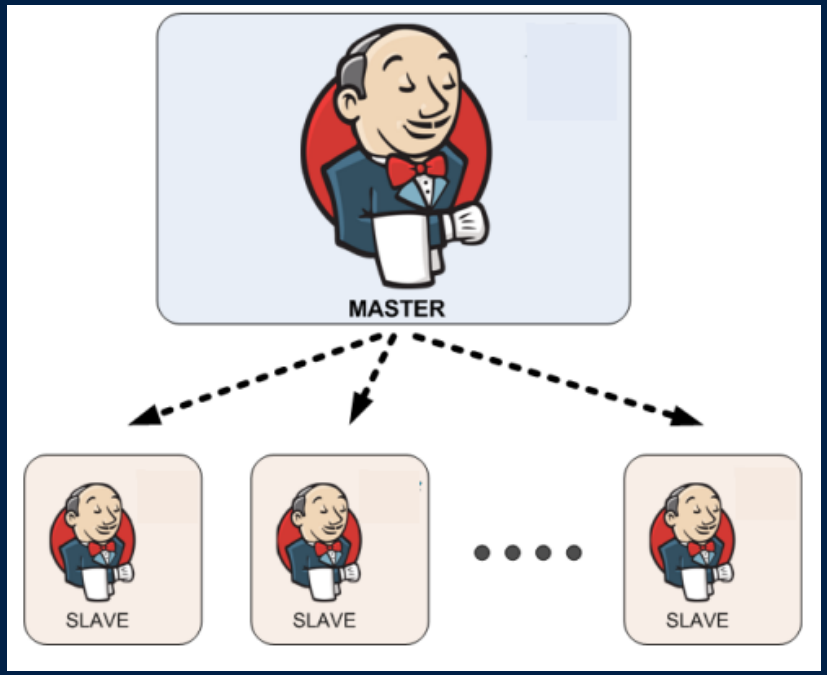
Jenkins' Master and Slave Architecture

Master:

- Schedule build jobs.
- Dispatch builds to the slaves for the actual job execution.
- Monitor the slaves and record the build results.
- Can also execute build jobs directly.

Slave:

- Execute build jobs dispatched by the master.



Jenkinsfile Pipeline

Why should you use Jenkinsfile pipeline?

- Pipeline can be run in a loop.
- It supports larger projects that may involve a high CPU job, provided the Jenkins infrastructure is scalable enough to support it.
- Since Jenkins pipeline is written in code, any number of users can use it as a template, modify it and run customized tests and processes.
- Multiple jobs can be run in parallel.
- Jenkins Pipeline is robust.
- Pipeline can automatically be resumed from it might have stopped for any reason.

Understanding The Jenkinsfile Pipeline

Pipeline:

•A Pipeline's code defines your entire build process, which typically includes stages for building an application, testing it and then delivering it.

Node:

•A node is a machine which is part of the Jenkins environment and can execute a Pipeline.

Stage:

•A stage block defines a conceptually distinct subset of tasks performed through the entire Pipeline

•Step:

•A single task. Fundamentally, a step tells Jenkins what to do at a particular point in time (or "step" in the process).

Understanding The Pipeline

1) Execute this Pipeline or any of its stages, on any available agent.
2) Defines the "Build" stage.
3) Perform some steps related to the "Build" stage.
4) Defines the "Test" stage.
5) Perform some steps related to the "Test" stage.
6) Defines the "Deploy" stage.
7) Perform some steps related to the "Deploy" stage.

Jenkinsfile (Declarative Pipeline)

```
pipeline {
    agent any 1
    stages {
        stage('Build') { 2
            steps {
                // 3
            }
        }
        stage('Test') { 4
            steps {
                // 5
            }
        }
        stage('Deploy') { 6
            steps {
                // 7
            }
        }
    }
}
```

Implementing reliable
machine learning solutions

ML ARCHITECTS
BASEL

Operating Models – Technologies – Culture & Skills

Consulting – Engineering - Training