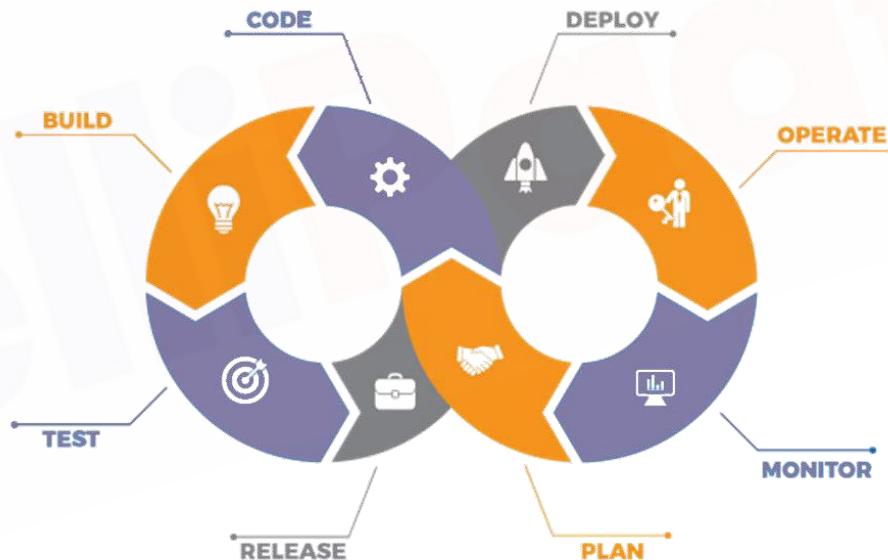




Continuous Integration Using Jenkins



Agenda

01

Introduction to
Continuous Integration

02

What is
Jenkins?

03

Installing Jenkins
on AWS

04

Jenkins
Architecture

05

Managing Nodes
on Jenkins

06

Jenkins Integration
with DevOps Tools

07

Understanding
CI/CD Pipelines

08

Creating an End-to-end
Automated Pipeline in AWS

Why Continuous Integration?

Before Continuous Integration



Version 1



Developer 1



Version 1



Developer 2



Version 1



Developer 3

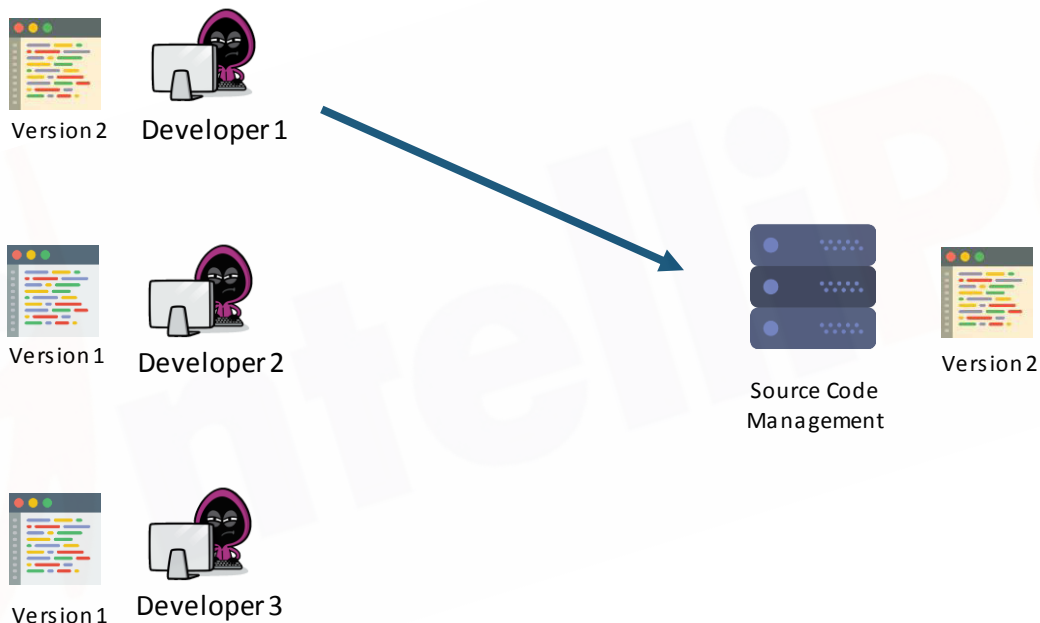


Source Code
Management

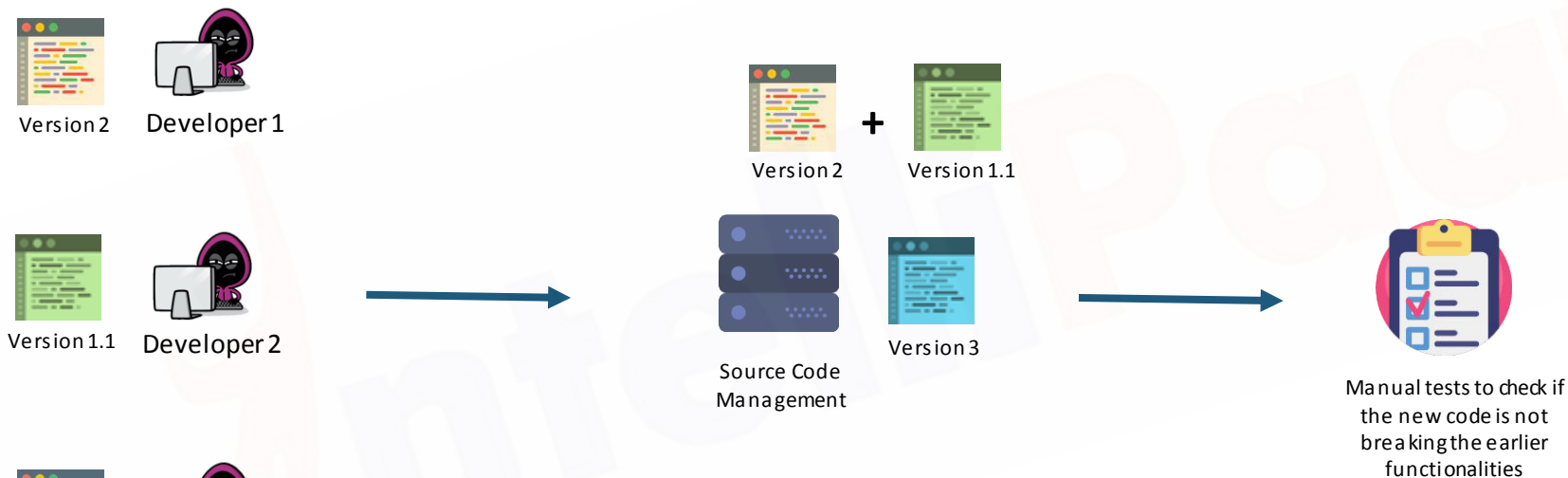


Version 1

Before Continuous Integration



Before Continuous Integration



Problems before Continuous Integration

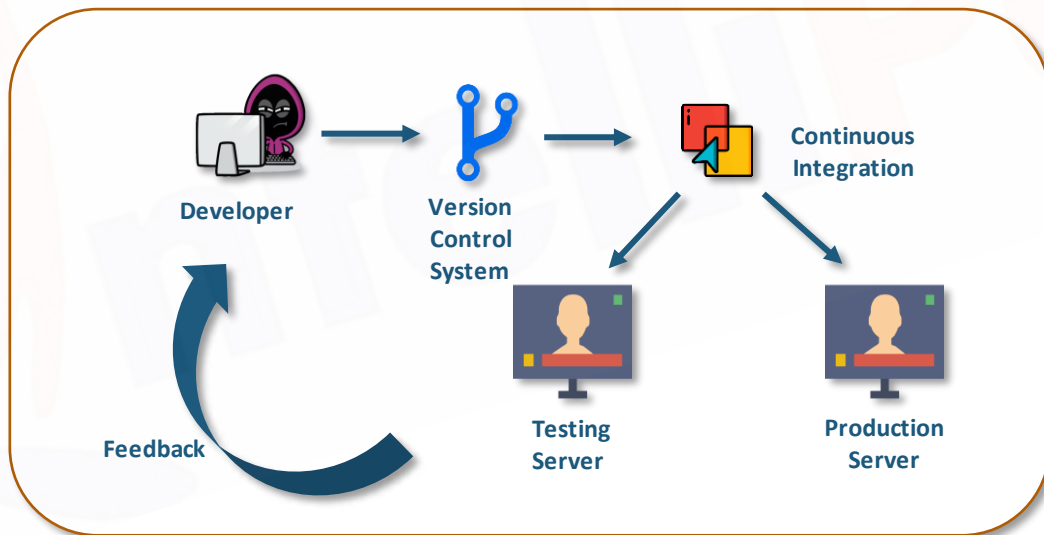


- ❌ Infrequent Commits led to bigger releases in one go leading to complex integration.
- ❌ Manual testing took a lot of time.
- ❌ Feedback took a lot of time to reach the developer.
- ❌ It involved high risk and uncertainty.

What is Continuous Integration?

What is Continuous Integration?

The process of having shorter release cycles (sometimes, several times a day), i.e., creating small features and integrating them to the source code and employing automated build and test processes for quicker feedback is called Continuous Integration.



Advantages of Continuous Integration



- ✓ Frequent Commits, hence small feature release
- ✓ Automated Build and Testing
- ✓ Instant feedback to the developer
- ✓ Low risk and faster delivery

What is Jenkins?

What is Jenkins?

Jenkins is an open-source automation server written in Java. Jenkins helps to automate the non-human part of the software development process, with continuous integration and facilitating technical aspects of continuous delivery.



Features of Jenkins



Adoption: Jenkins is extremely popular among the open-source community; hence, there are more than 147,000 active installations throughout the world and 1 million people are using it.



Plugins Support: With an extremely active open-source community, Jenkins has around 1000 plugins that allow it to integrate with most of the development, testing and deployment tools.



Advantages of Jenkins

Before Jenkins

- ★ Locating and fixing bugs in the event of build and test failure was difficult and time consuming.
- ★ Tests were triggered manually.
- ★ No central place for triggering jobs on remote systems.

After Jenkins

- ★ Smaller and automated continuous build and testing make the task accurate and faster.
- ★ Developers have to just commit the code to the remote repository, build, test and deployment happen automatically.
- ★ All builds or tests on multiple remote systems can be controlled from one place.

Installing Jenkins on AWS

Installing Jenkins on AWS



1. Launch an AWS Instance
2. Connect through SSH
3. Execute the following commands:

Jenkins Installation:

```
$> sudo apt-get update
```

```
$> sudo apt install openjdk-8-jdk
```

```
$> wget -q -O - https://pkg.jenkins.io/debian/jenkins.io.key | sudo apt-key add -
```

```
$> sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ >
```

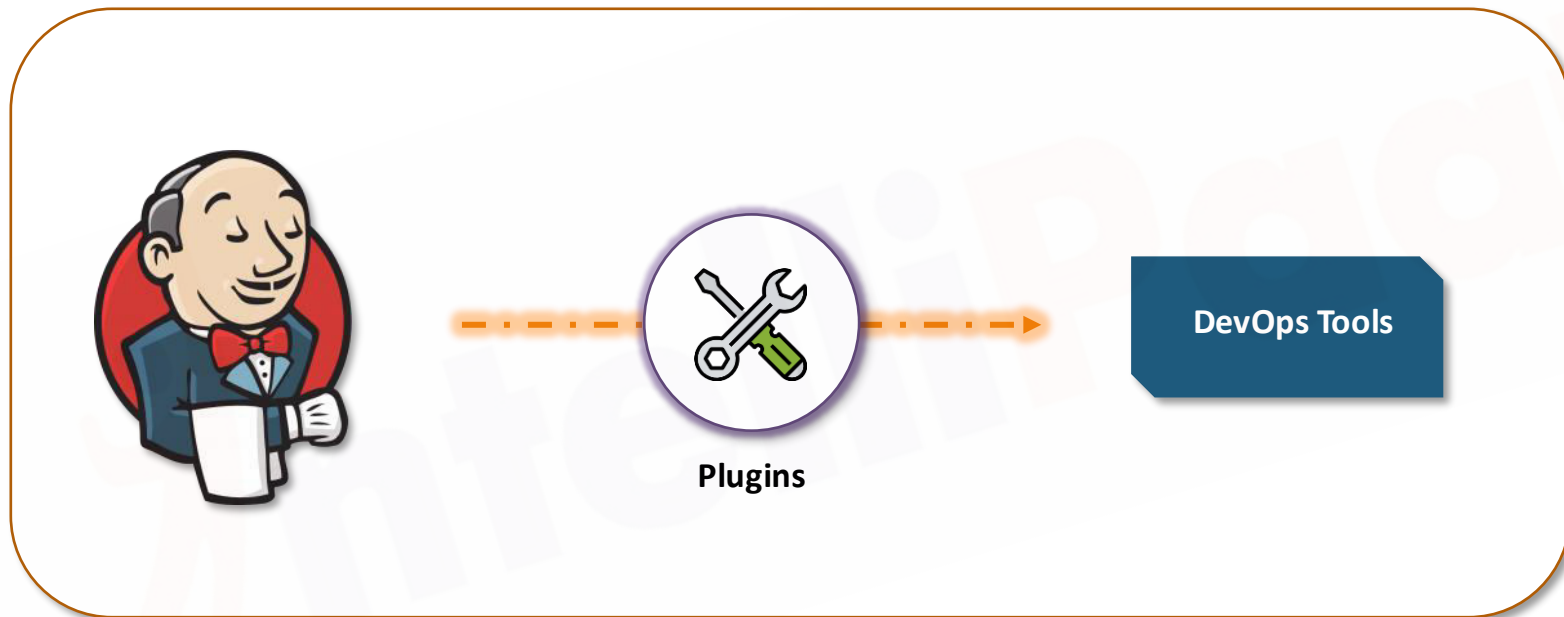
```
/etc/apt/sources.list.d/jenkins.list'
```

```
$> sudo apt update
```

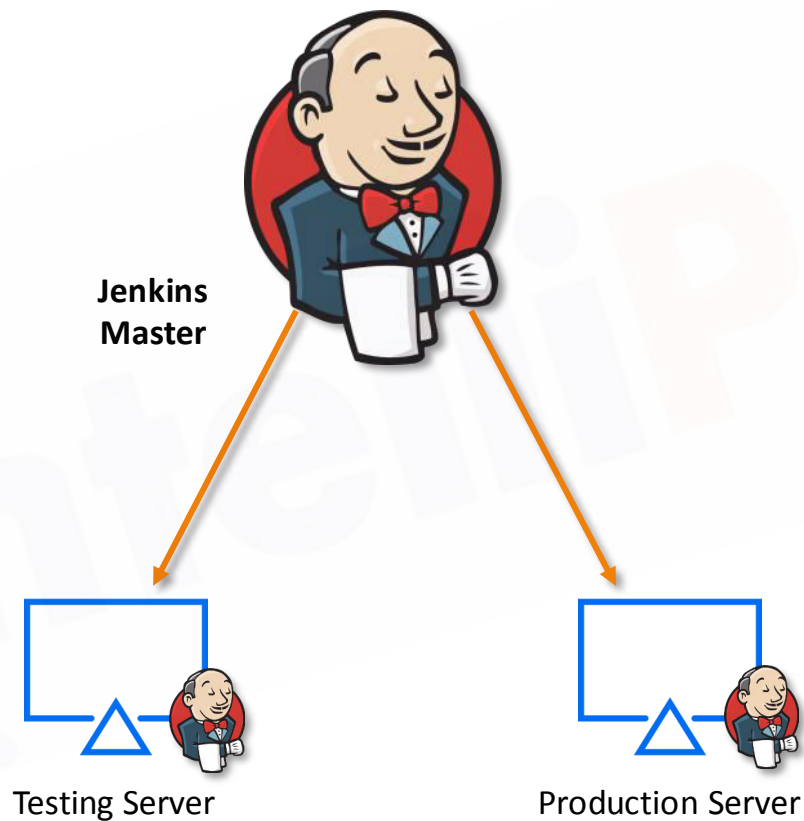
```
$> sudo apt install jenkins
```


Jenkins Architecture

Jenkins Architecture



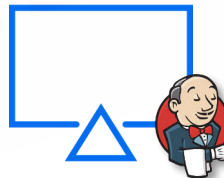
Jenkins Architecture



Managing Nodes on Jenkins

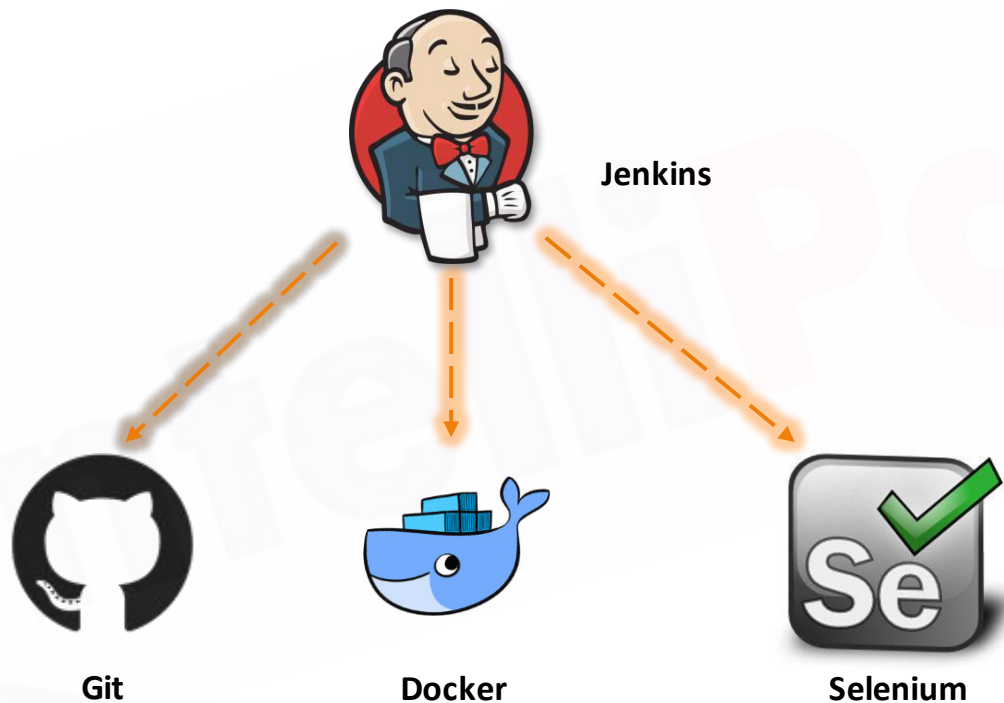
Managing Nodes on Jenkins

Add a slave node to Jenkins using JNLP connection



Jenkins Integration with DevOps Tools

Jenkins Integration with DevOps Tools



Jenkins Integration with DevOps Tools



Git



Docker



Selenium

Copy a Git repository to the slave's filesystem from Jenkins master



Jenkins Integration with DevOps Tools



Git

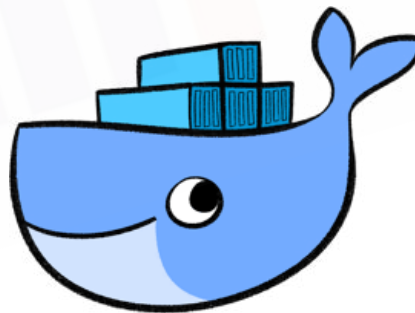


Docker



Selenium

Containerize the website in the previous step to a Docker Container using Jenkins



Jenkins Integration with DevOps Tools



Git



Docker



Selenium

Create a test case for the website in the previous step and execute the test on the slave using Jenkins



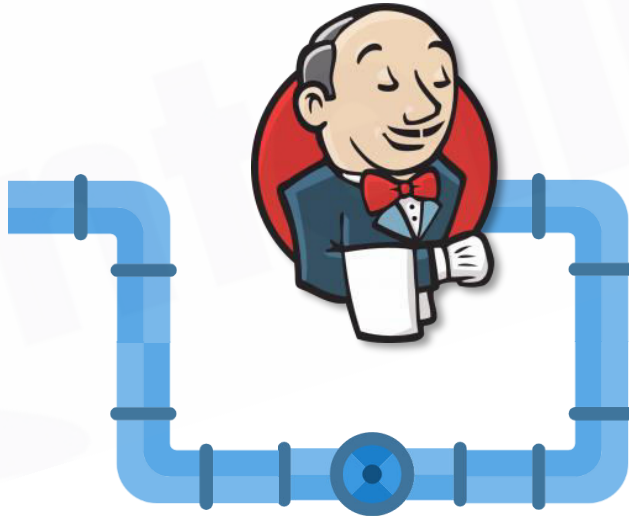
Understanding CI/CD Pipelines

What are CI/CD Pipelines?

CI/CD Pipelines, i.e., Continuous Integration, Continuous Delivery and Deployment pipelines, are a way of running Jenkins jobs in a sequence, which resembles a pipeline view.



Finished Application



What are CI/CD Pipelines?

CI/CD Pipelines, i.e., Continuous Integration, Continuous Delivery and Deployment pipelines, are a way of running Jenkins jobs in a sequence, which resembles a pipeline view.

For Example:



GitHub



Git-Job

Downloads Git Code
to Repository



Build-Website

Containerizes the
Git Code into
Apache Container



Test-Website

Validates the
Website

Creating an Automated CI/CD Pipeline

Creating an Automated CI/CD Pipeline



1. Initiate a Git Webhook for the Jenkin's git-job repository
2. Trigger the jobs after the completion of previous jobs with the following map: Git-Job → Build-Website → Website-Test
3. Install the plugin for the pipeline view
4. Make changes to the website and commit the job to see the changes



Quiz

1. Can Jenkins execute jobs without slaves?

A. Yes

B. No

C. Minimum one slave is required

D. None of these

1. Can Jenkins execute jobs without slaves?

A. Yes

B. No

C. Minimum one slave is required

D. None of these

2. Which plugin in Jenkins helps us see the pipeline view?

A. Build Pipeline

B. Create Pipeline

C. Pipeline View

D. None of these

2. Which plugin in Jenkins helps us see the pipeline view?

A. Build Pipeline

B. Create Pipeline

C. Pipeline View

D. None of these

3. Which Protocol in Jenkins is used to connect to Jenkins Slave?

A. SSL

B. SSH

C. JNLP

D. None of these

3. Which Protocol in Jenkins is used to connect to Jenkins Slave?

A. SSL

B. SSH

C. JNLP

D. None of these

4. Which of these will help you in triggering jobs from Git automatically?

A. Git Commit

B. Git Rebase

C. Git Webhook

D. None of these

4. Which of these will help you in triggering jobs from Git automatically?

A. Git Commit

B. Git Rebase

C. Git Webhook

D. None of these

5. Do the slaves need Jenkins installed on them?

A. Yes

B. No

5. Do the slaves need Jenkins installed on them?

A. Yes

B. No



India: +91-7847955955

US: 1-800-216-8930 (TOLL FREE)



support@intellipaat.com



24/7 Chat with Our Course Advisor