Jenkins as CI-CD tool

Ganesh Palnitkar

Agenda

- What is Continuous Integration?
- Installing and Configuring Jenkins as a CI Tool
- What are plug-ins?
- Using plug-ins to Integrate Build and VCS tools.
- Creating first job
- Jenkins and workflow management

What is Continuous Integration and Delivery

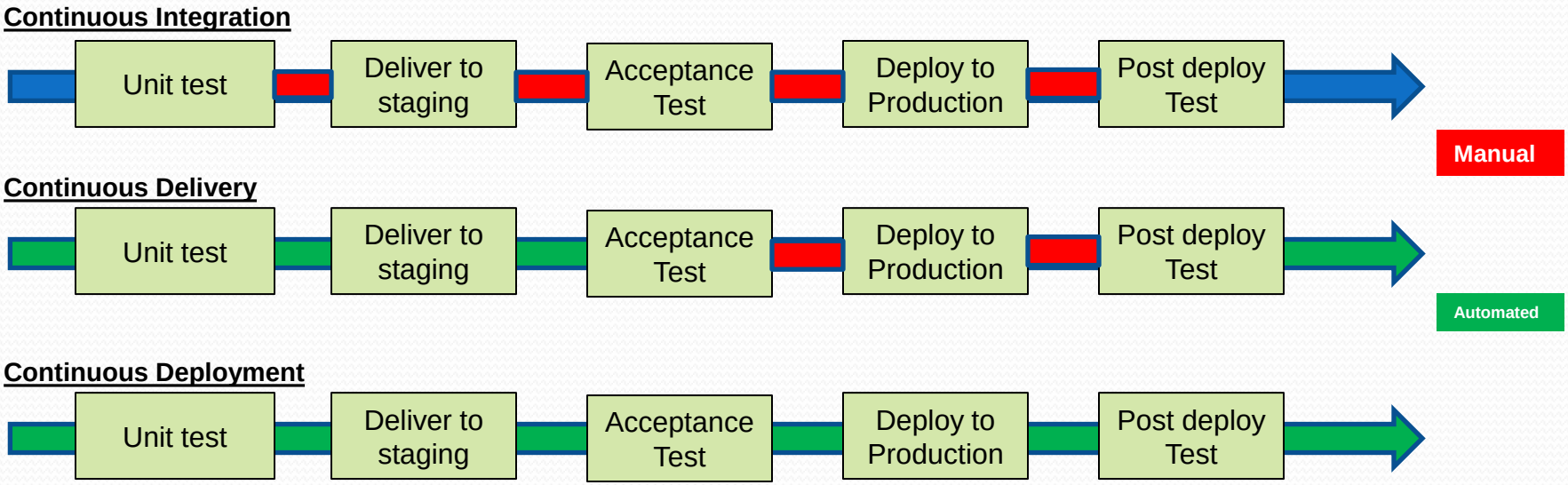
Continuous Integration (CI): Continuous integration is software development practice in which team members integrate their work frequently, leading multiple integrations per day. Each integration helps to reveals integrations errors in build success / failures as quickly as possible. This helps in significantly reducing integration problems and delivery timeline.

Continuous Delivery (CD) can be thought of as an extension to the continuous Integration. In this the teams always ensure that the code change is always releasable.

CDs ensures fast feedback from the test environments.

Continuous Deployment (CD): This is a next step of Continuous delivery where in every change that passes the automated testing is deployed to production environment automatically.

In the practical scenario Continuous deployment is still thought to be **impractical**.

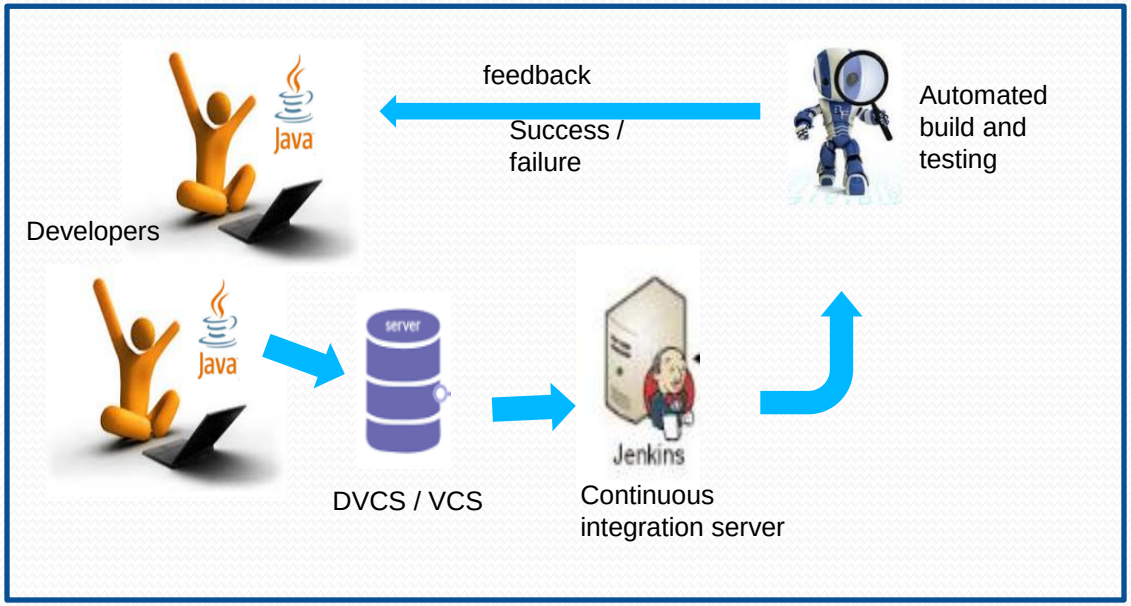


CI / CD Tools

CI / CD tools	Build Automation	Auto Code Analysis	Auto Testing	Deployment
Jenkins	gradle	sonarqube	cucumber	CHEF
Bamboo	maven	Flawfinder	MERCURY	puppet labs
circleci	MSBuild		Selenium	Octopus Deploy
				ANSIBLE

What is Continuous Integration

Jenkins Kohsuke Kawaguchi,



Recommended Configuration: • Java 8 • 1GB+ free memory • 50GB+ free disk space

- Available for installation as a 'package' or '.war' file.
- Jenkins can be used as a CI server or can be turned into a continuous delivery (CD) Hub.
- Jenkins can distribute work across multiple machines for build, tests and deployments across multi platform.
- Jenkins can be configured to run on port other than 8080 as well. This is done by changing setting in jenkins.xml file.

Benefits:

- Early feedback resulting in bug-free builds.
- Automated testing and build operation- eliminate human error.
- Enabler for success of Agile projects.
- Enabler for imbibing DevOps culture.

Advantages:

- Cross platform CI tool.
- Easy to use open source tool.
- Wide range of plugins available that supports almost all DevOps tools in the use.
- Widely used across industry, availability of expertise.

Jenkins is used by many big names as, Linked in, Ubuntu, Openstack, AngularJS, etc.

Installing and Configuring Jenkins

- Download Jenkins from https://jenkins-ci.org.
- Also refer to the Jenkins official page for details documentation on Jenkins installation various flavors of OSs. https://wiki.jenkins-ci.org/display/JENKINS/Installing+Jenkins+as+a+Windows+service
- Jenkins can be installed as a service or a .war file to be used by any Webapp server like Tomcat etc.
- Post install, create **Admin** user and Password for the Jenkins.
- **Plug-ins** need to be installed to start using Jenkins. By default during installation, few of the plug-ins will get installed along the Jenkins package.
- Setup **global security** options for the installed version.
- One can create Jenkins user database or can integrate with **LDAP** database.
- Install plugins for **SVN / GIT** that we can use it as the VCS solution.
- Install Build tool plug-in, in this case for **Maven**. Jenkins supports variety of build tools, such as ANT etc.,
- Jenkins can also be used over command line using '**Jenkins CLI**'.
- Plugins can be configured to be **updated automatically**, or one can write a own plug-in for if one is not available for in the list for download.

Official Jenkins documentation:

- https://wiki.jenkins-ci.org/display/JENKINS/Installing+Jenkins+on+Ubuntu
- https://wiki.jenkins-ci.org/display/JENKINS/Installing+Jenkins+as+a+Windows+service

For Jenkins installation using WAR file on any operating system, setup yhe JAVA_HOME, once the Java_home is set, run the command from command prompt.

C:\java –jar jenkins.war

Installing and Configuring Jenkins

For Jenkins installation on Ubuntu, follow below process.,

```
$ wget -q -O - https://pkg.jenkins.io/debian/jenkins-ci.org.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-get update
$ sudo apt-get install jenkins
```

For Jenkins installation on Centos 7.0

```
$ sudo yum install epel-release
$ sudo yum update
$ sudo yum install java-1.8.0-openjdk.x86_64
$ sudo wget -O /etc/yum.repos.d/jenkins.repo http://pkg.jenkins-ci.org/redhat-stable/jenkins.repo
$ sudo rpm --import http://pkg.jenkins-ci.org/redhat-stable/jenkins-ci.org.key
$ sudo yum install jenkins
```

While the Jenkins installation is done, the configuration are stored in /var/lib/Jenkins incase of a Linux installation, or /program file/Jenkins/ incase of Windows installation.

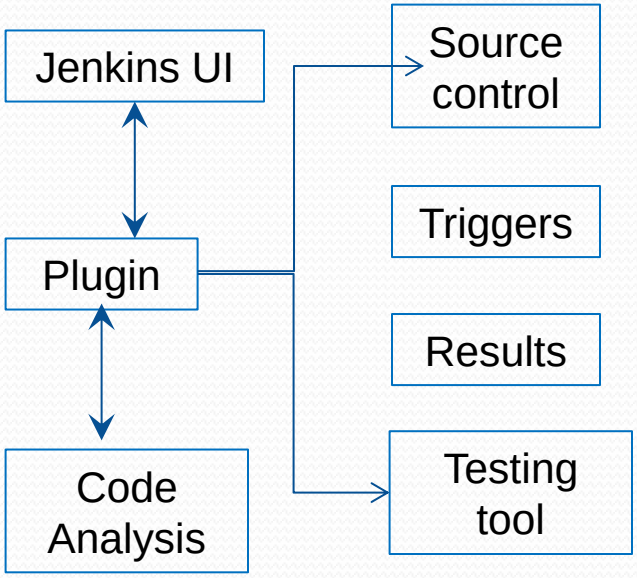
There is no separate database maintained for Jenkins configuration file. So for backing up the Jenkins setting, these folders should be backed up.

Jenkins can also be hosted inside a **Docker container**. Docker store provides Jenkins image that can be used for creating a container to host Jenkins.

What's a Plug-in

https://plugins.jenkins.io/

Plugins in Jenkins are the tools to integrate different tools with each other. Plugin makes it possible to elevate the power of Jenkins.



Plugins can be written to enable integrate an enterprise application that might be an in-house application.

https://wiki.jenkins-ci.org

This is the official URL for information on Jenkins Plugin

Using plugins to Integrate tools

Source code Plugins : GIT , Mercurial, TFS

Trigger Plugin: **GITHUB** pull request:

Join: (different jobs / projects can be configured to run together, one after other etc.),

Locks and **Latches**:

Build Tool Plugins: Copy artifacts, Fitness, MSBuild, Promoted build

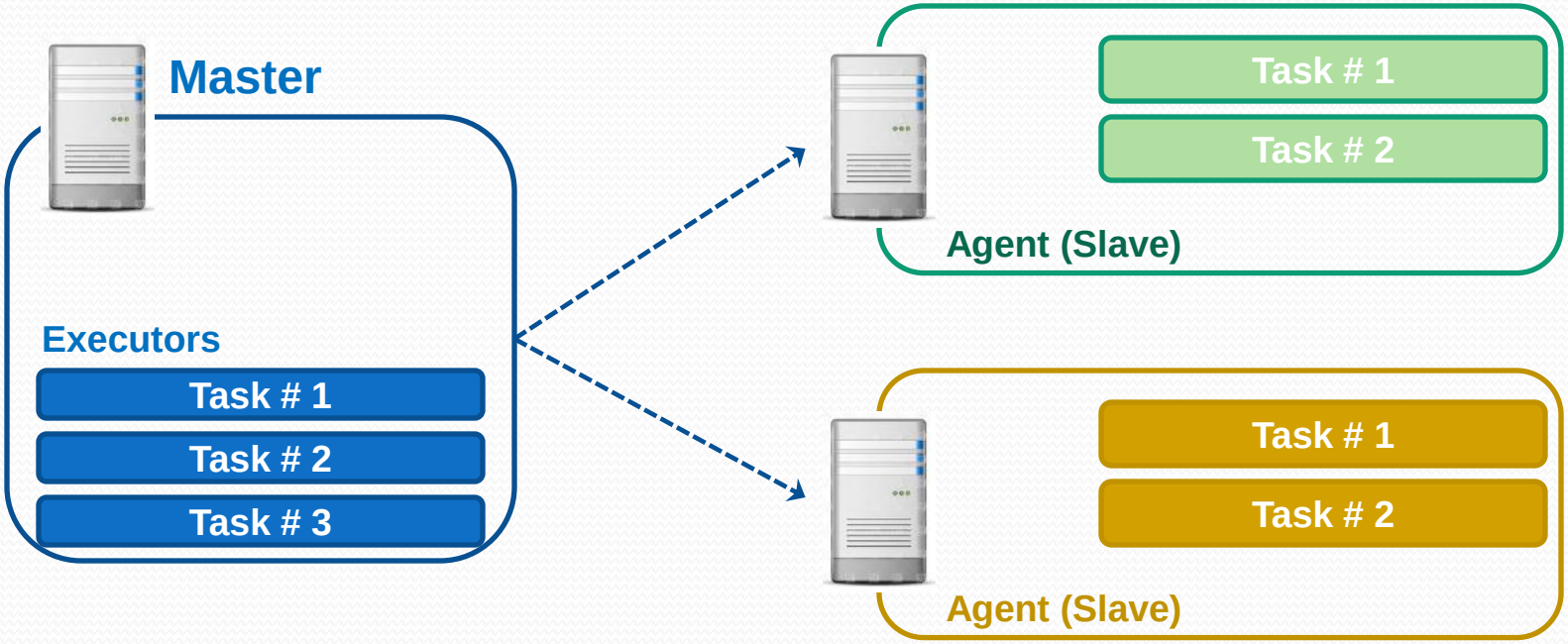
MSTest runner, NANT, MAVEN, PowerShell.

Wrapper Plugin: automated start and stop of a function upon certain action completion etc.

Notifier Plugins: HipChat, IRC, Twitter, Jabber… helps to send notifications.

Reporting plugin: Cobertura (code coverage), Findbugs, MSTest, PMD, etc.

Master <> Agent model : Jenkins Nodes



Jenkins Pipeline Job

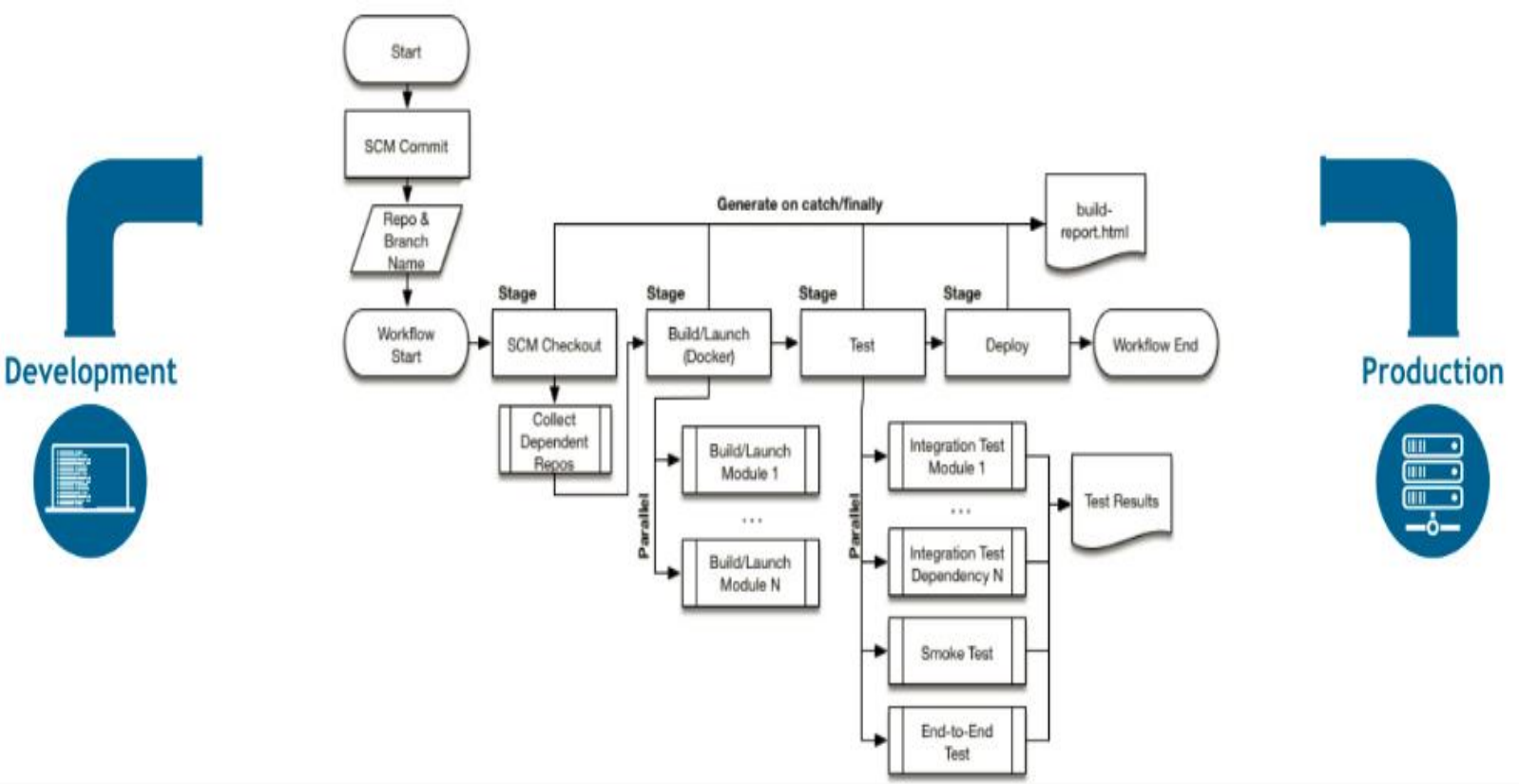
Pipeline job in Jenkins is automation of executing actions in a typical Build / delivery cycle.
Pipeline job is setup using the pipeline job type and then setting up a Groovy script to execute the actions one by one. Jenkins provides a wizard to create the Groovy script for the defined actions in the Build cycle, like,

- SCM checkout
- Build action, like compile, package, test etc.
- Archiving the artifacts.
- ……..

Pipeline Job also provides flexibility to assign multiple jobs at a time, this also needs a Node to be available which Jenkins can assign the Pipeline job to.

Jenkins – CI – Cd tool

Jenkins Pipeline View



Thanks You