Jenkins Administration

Hi There !!

Instructor : Vishal Saini

RedHat Certified Architect

AWS certified professional

Corporate Trainer and Consultant with 13+ years of rich experience in Linux, Cloud computing, Devops.

Objectives

* Understanding Jenkins
* Understanding Git
* Working With Jenkins Plugins
* Creating Jenkins Jobs
* Implementing Automating Testing
* Distributed Jenkins Configuration
* Maintaining Jenkins
* Working with Continuous Deployment/Delivery
* Jenkins Tips and Tricks

Topic 1: What is Jenkins

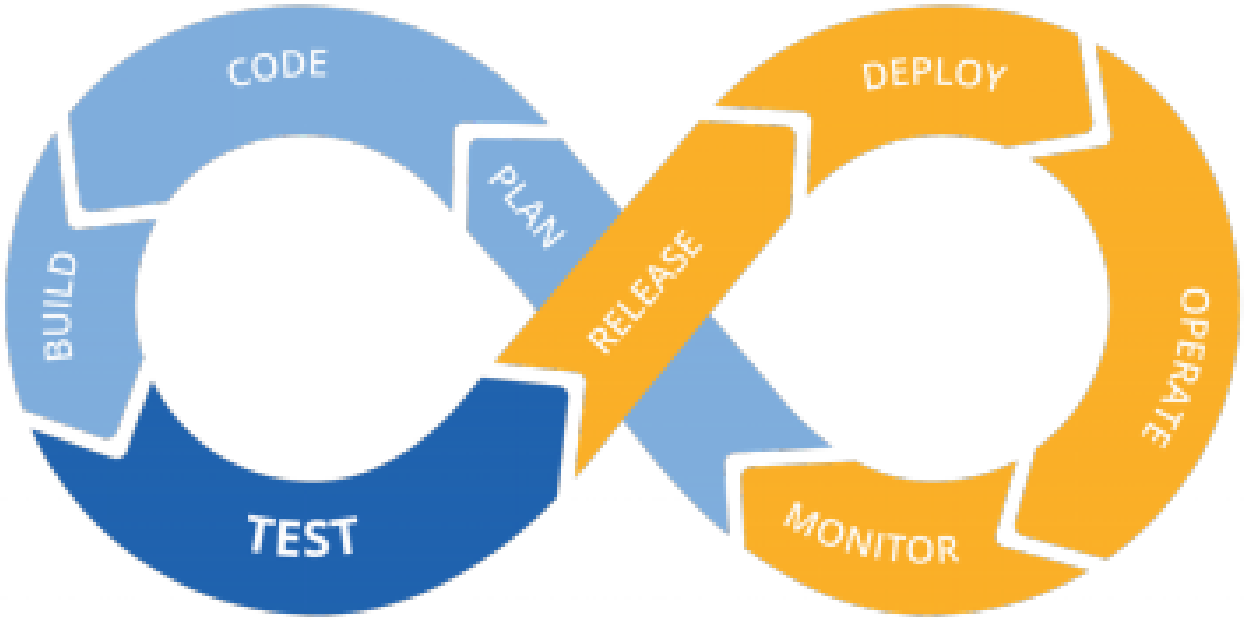
Before you begin

What is Devops ??

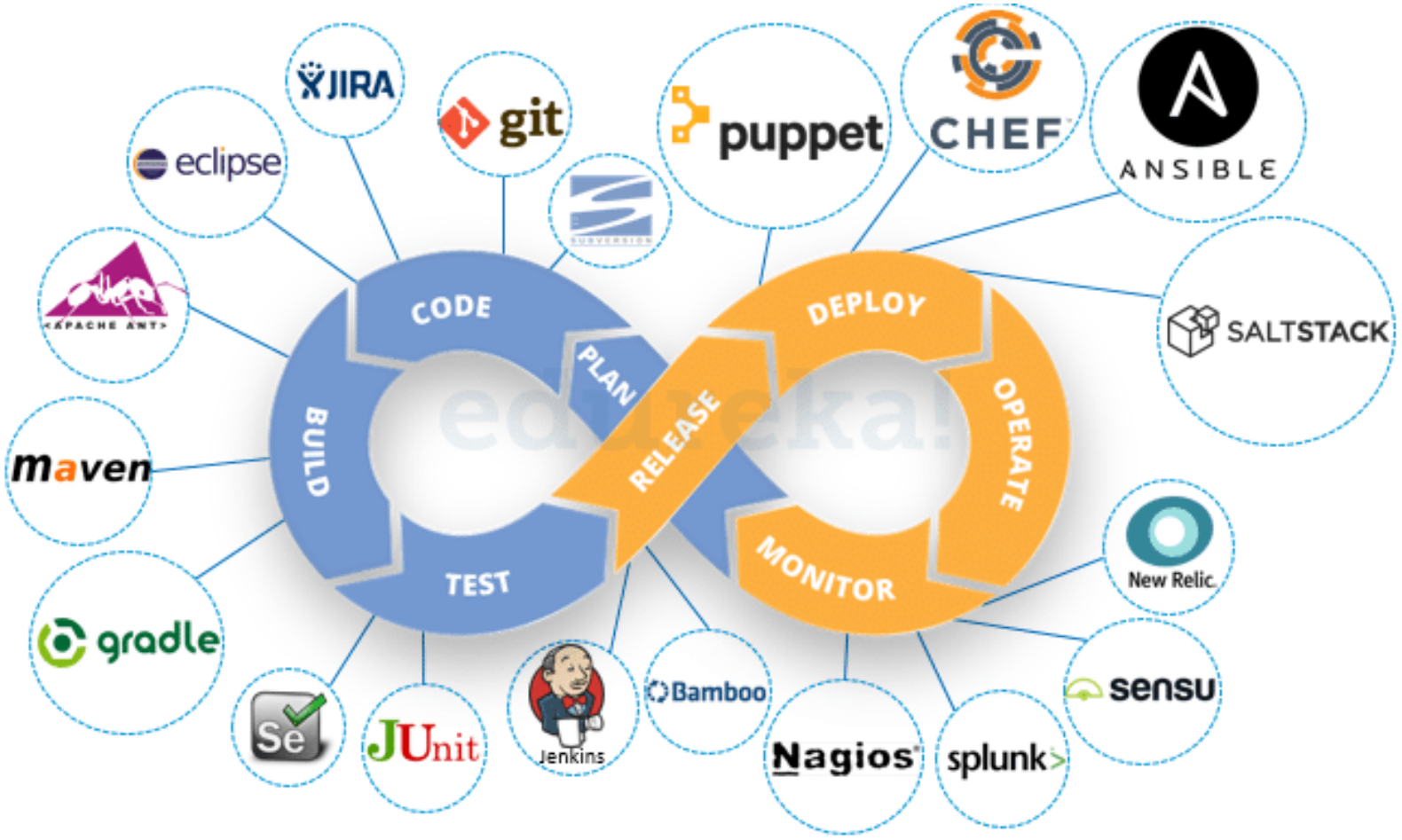
“DevOps” as a term was first coined in 2009 by Patrick Debois

DevOps is a set of practices that combines software development (Dev) and IT operations (Ops). It aims to shorten the systems development life cycle and provide continuous delivery with high software quality.

Before you begin .....



Before you begin .....



CI - Defined

Continuous Integration is a software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple integrations per day.

Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible" – Martin Fowler

CI – What Does it really Means ??

At a regular frequency (ideally at every commit), the system is:

*** Integrated : All changes up until that point are combined into the project**

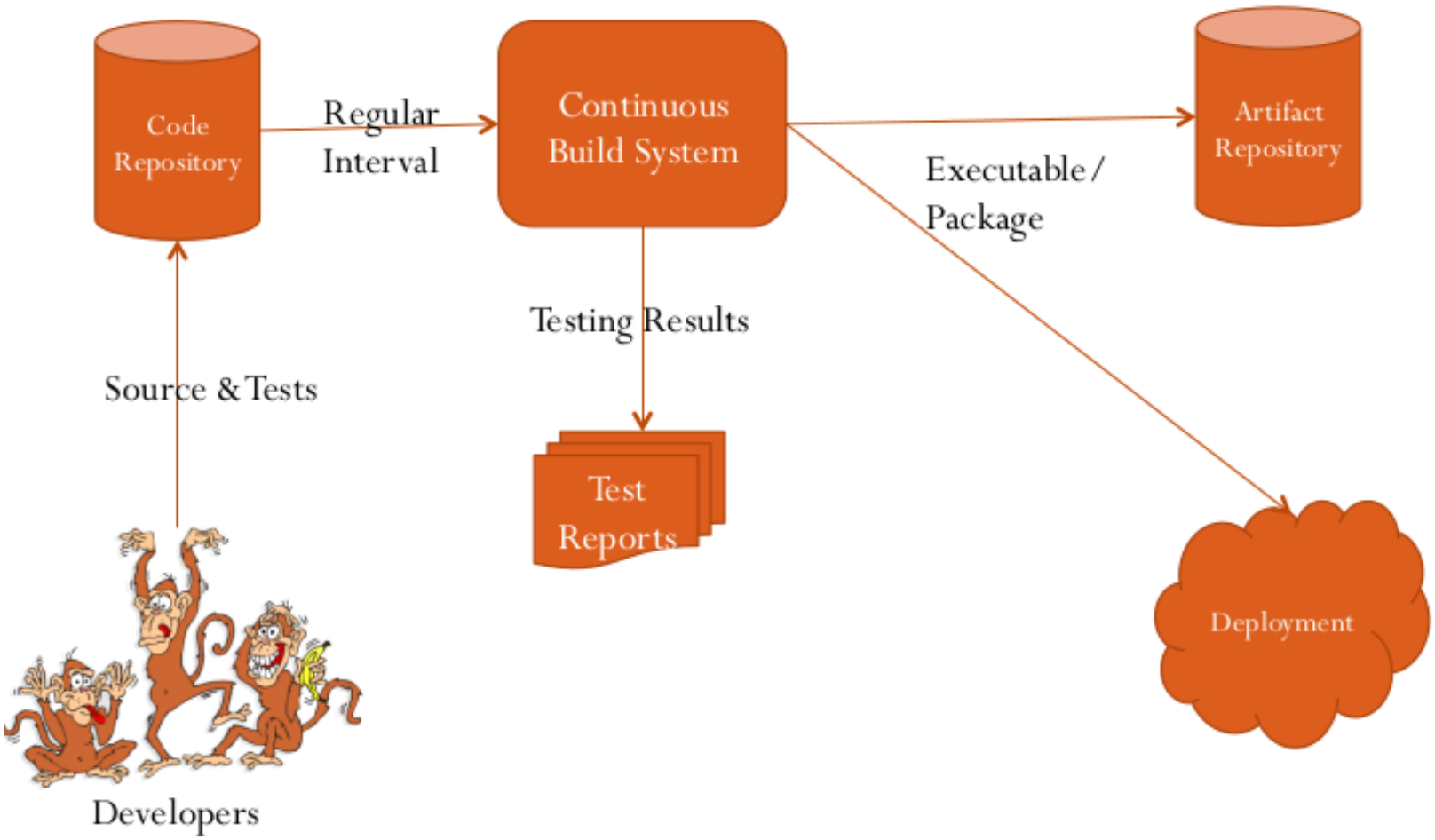
*** Built : The code is compiled into an executable or package**

*** Tested : Automated test suites are run**

*** Archived : Versioned and stored so it can be distributed as is, if desired**

*** Deployed : Loaded onto a system where the developers can interact with it**

CI - Workflow


Code Repository
Regular Interval
Continuous Build System
Artifact Repository
Executable / Package
Testing Results
Source & Tests
Test Reports
Deployment
Developers

CI – Benefits

* Immediate bug detection
* No integration step in the lifecycle
* A deployable system at any given point
* Record of evolution of the project

CI – Best Practices

- **Maintain a single-source repository**
- **Keep the build fast, Self-testing and Automate the build**
- **Test in a clone of the production environment**
- **Everyone commits to the mainline every day**
- **Automate deployment**
- **Fix broken builds immediately**

CI – Tools

* Code Repositories

  > SVN, Mercurial, Git, Bazaar

* Continuous Build Systems

  > Jenkins, Bamboo, Cruise Control, Travis CI, Gitlab

* Test Frameworks

  > JUnit,Cucumber, CppUnit, Selenium

* Artifact Repositories

  > Nexus, Artifactory, Archiva

Jenkins

Jenkins is a free and open source automation server.

It helps automate the parts of software development related to building, testing, and deploying, facilitating continuous integration and continuous delivery.

Jenkins ..

* **Branched from Hudson**
* **Java based Continuous Build System**
* **Runs in servlet container**

  Glassfish, Tomcat
* **Supported by over 400 plugins**

  SCM, Testing, Notifications, Reporting,

  Artifact Saving, Triggers, External Integration
* **Under development since 2005**
* **http://jenkins.io**

Jenkins History

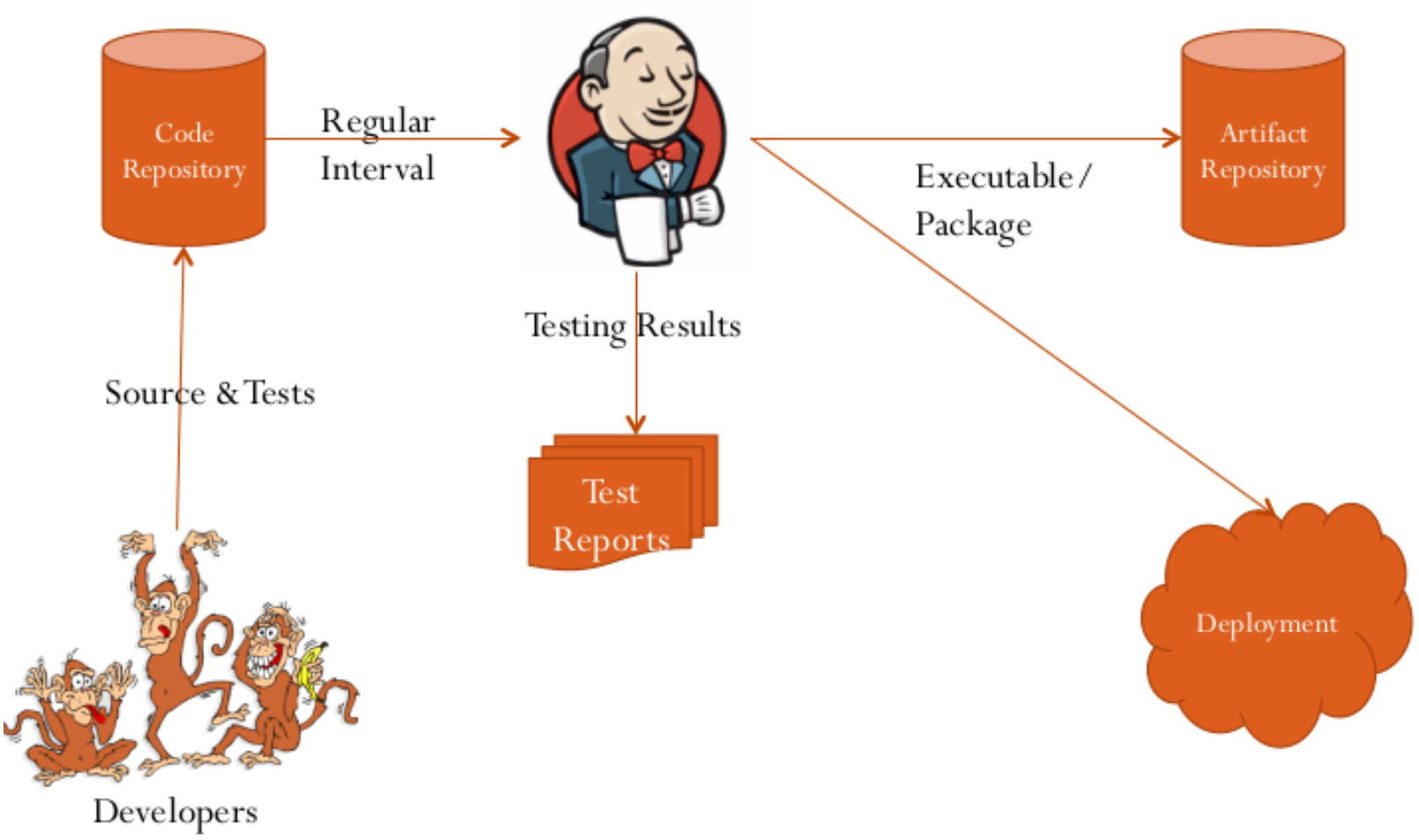
*** 2005 - Hudson was first release by Kohsuke Kawaguchi of Sun Microsystems**

*** 2010 – Oracle bought Sun Microsystems**

> Due to a naming dispute, Hudson was renamed to Jenkins

> Oracle continued development of Hudson (as a branch of the original)

Jenkins – Fitting In



Jenkins Advantages

* Continuous Integration and Continuous Delivery
* Easy installation
* Easy configuration
* Plugins and Extensible
* Distributed

Who uses Jenkins ?

Running Jenkins yourself...

* Jenkins is packaged as a WAR, so you can drop it into whichever servlet container you prefer to use.

* Jenkins comes pre-packaged with a servlet if you just want a light-weight implementation

* Native/Supported packages exist for

Windows

Linux flavor : Ubuntu/Debian/Redhat/Fedora/Centos

Mac OSX

OpenSUSE / FreeBSD/ OpenBSD

Solaris/OpenIndiana

Running Jenkins yourself... Updates

Jenkins has two release lines

* **Standard releases**
  - **Weekly bug fixes and features**
* **Long-Term Support releases**
  - **Updates about every 3 months**
  - **Uses a “Stable but older” version from the standard release line**
  - **Changes are limited to backported, well-tested modifications**

Lab Orientation

Classroom Setup

> 2 Ubuntu Boxes with internet access

> One will be having jenkins installed

> Another will be used as agent in further topics.

Install Jenkins

Step 1: Install java

Since Jenkins is a Java application, the first step is to install Java. Update the package index and install the Java 8 OpenJDK package with the following commands:

$ sudo apt update

$ sudo apt install openjdk-8-jdk

Install Jenkins ..

Step 2: Add Repository

Import the GPG keys of the Jenkins repository using the following wget command:

```
$ 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'
```

Install Jenkins ..

Step 3: Install Jenkins

Once the Jenkins repository is enabled, update the apt package list and install the latest version of Jenkins by typing:

$ sudo apt update

$ sudo apt install jenkins

$ systemctl status jenkins

Install Jenkins ..

Step 4: Verify Installation

$ systemctl status jenkins

To set up your new Jenkins installation, open your browser, type your domain or IP address followed by port 8080,

http://your_ip_or_domain:8080