

# Full Stack Development

## Containers, Microservices and UI

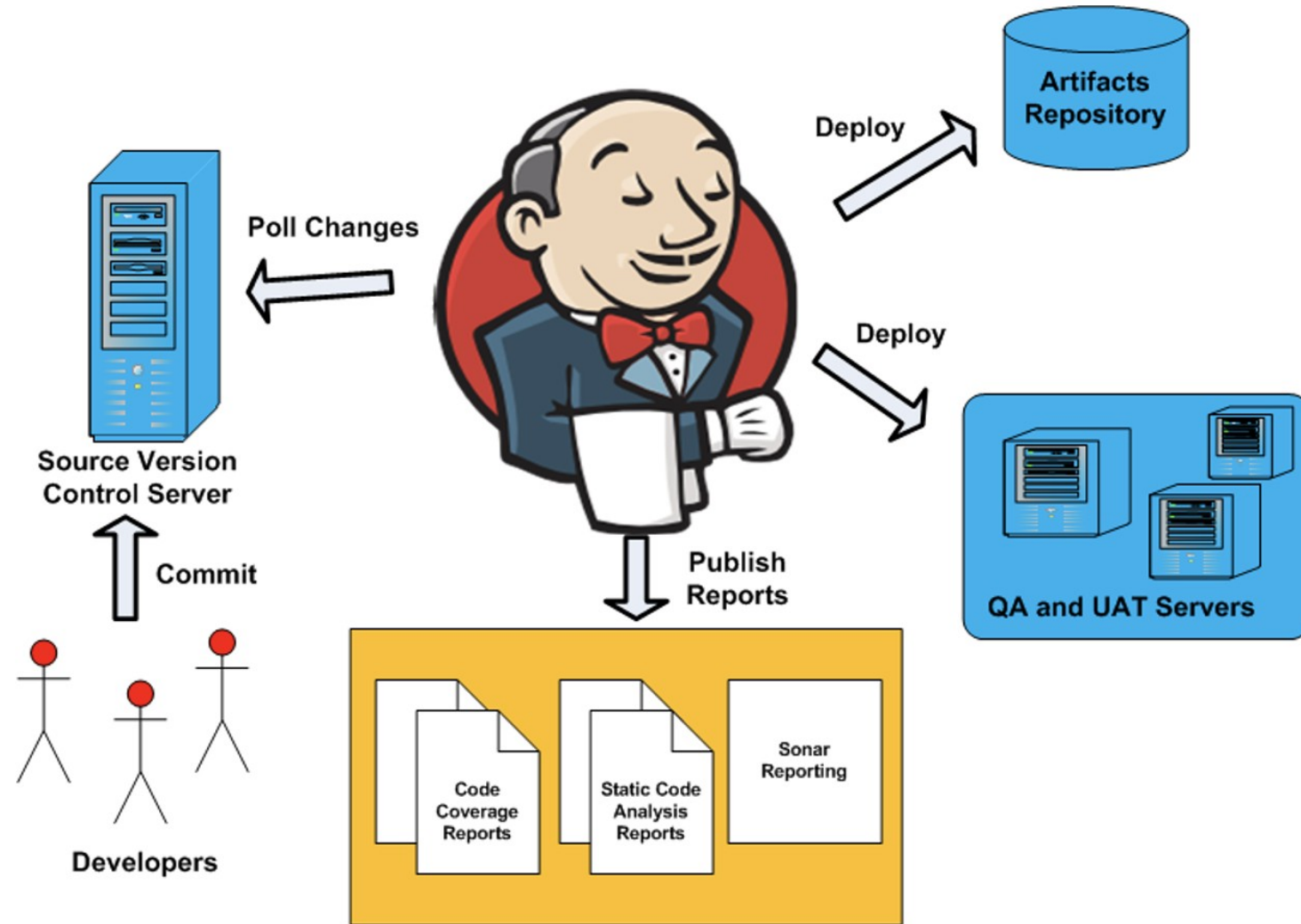
8. CICD with Jenkins



# Jenkins

- Continuous Integration (CI) Server
  - Open Source, free and written in Java
  - Large and dynamic community with massive adoption
  - Easy to install on many different platforms
  - Easy to use with a user-friendly interface
  - Significant amounts of resources and tutorials available
  - More than 1000 plugins
  - New plugins released every week
  - Extensible architecture – easy to extend and customize
  - Distributed builds
  - Many more features...

# How Jenkins Fits in CI and CD



# Jenkins - History

- Jenkins is derived from Hudson
- Hudson was first released by Kohsuke Kawaguchi of Sun Microsystems in 2005
- Initially it was only used within Sun but by 2010 Hudson captured 70% of CI market share
- Oracle bought Sun Microsystems in 2010
- Due to naming and an open-source dispute, original Hudson team forked Jenkins forked from Hudson as a new project
- Oracle continued the development of original Hudson
- Majority of Hudson users migrated to Jenkins within few months of the initial Jenkins release

# State of the Jenkins Community

- Largest installed base of any open-source continuous integration and delivery platform
- More than 100K active users in open-source Jenkins CI project
- Community contributed with more than 1000 plugins
- Over 1000+ public repositories on GitHub and strong commit activity
- Quick feedback with addressing bugs and issues
- Get answer on any questions from Jenkins user mailing list and StackOverflow
- Chances are other people have had your question and may have a solution
- Learn more about Jenkins at <http://jenkins-ci.org/>
-

# Managing Jenkins

- All in one configuration dashboard
- Configure JDKS, Ant, Maven, Security, Email and Version Controls
- Install new plugins and update any existing plugins
- Configure parallel and distributed builds
- View Jenkins logs and statistics in real time



# A Tour of Jenkins

## Demo





# Running Hello World

Demo





# Running Maven

## Demo





# Hello World

## Lab Jenkins 1





# Runing Maven

## Lab Jenkins 2



# Jenkins Distributed Architecture

- The main node is the default node
  - Runs the Jenkins instance
  - Manages access to Jenkins Interface
  - Should be the only node accessible via web interface
- The master/main node is the controller
  - Best practice is that CI/CD jobs are delegated to other nodes
- Agent/Slave nodes
  - Tasked with running parts of the CI/CD jobs
  - Agents are tagged so they can be used selectively
  - Jenkins does not have to be installed on an agent
  - Just Java to run the Jenkins agent jar file
  - Communication takes place via ssh



# Jenkins Pipelines

- A pipeline defines a CI/CD process for that project
- Stages
  - Sequential series of steps to be executed
  - Build > Test > Package > Deploy
- JenkinsFile (infrastructure as code)
  - A script that defines the stages of a pipeline
  - Older form is written in the Groovy scripting language
  - New form is a declarative form of scripting
- Three basic configurations
  - The JenkinsFile is kept in Jenkins
  - The JenkinsFile is kept in a SCM repository
  - The JenkinsFile is kept in the project itself
    - This allows for multi branch builds

# JenkinsFile HelloWorld

## Demo





# JenkinsFile Hello World

## Lab Jenkins 3



# Post Build Steps

- In addition to stages, Jenkins has a post build stage
- Contains any of a number of clause types
  - Always – always executes
  - Failure – executes only when the pipeline fails
  - Success – executes only when the build succeeds
  - Cleanup – always runs after all the other clauses run
  - Changed – runs if the pipeline or a stage completion status is different from a previous run
- There are more clause types at:
  - <https://jenkins.io/doc/book/pipeline/syntax/#post>



# Environment Variables

- There are a number of predefined environment variables
  - Accessed via the global variable “env”
  - Value is accessed using Groovy syntax
  - “Build ID is \${BUILD\_ID}”
- We can define environment variables in either the whole pipeline or a given stage using the environment block

# MultiStage Pipeline

## Demo





# MultiStage Pipelines

## Lab Jenkins 4





# Advanced MultiStage

## Demo





Questions?





# End Module

