

Building and Using a Multi-node Jenkins Farm

SETTING UP A MASTER NODE



George Smith

SOLUTION ARCHITECT & EDUCATOR

@GeorgeS11323298

Intro



CI & CD - Refresher

Jenkins deployment options

Installing Jenkins on Amazon Linux 2

Core Jenkins Plugins

Integration with SCM

Prerequisites



Working knowledge of Jenkins

Experience with command-line tools

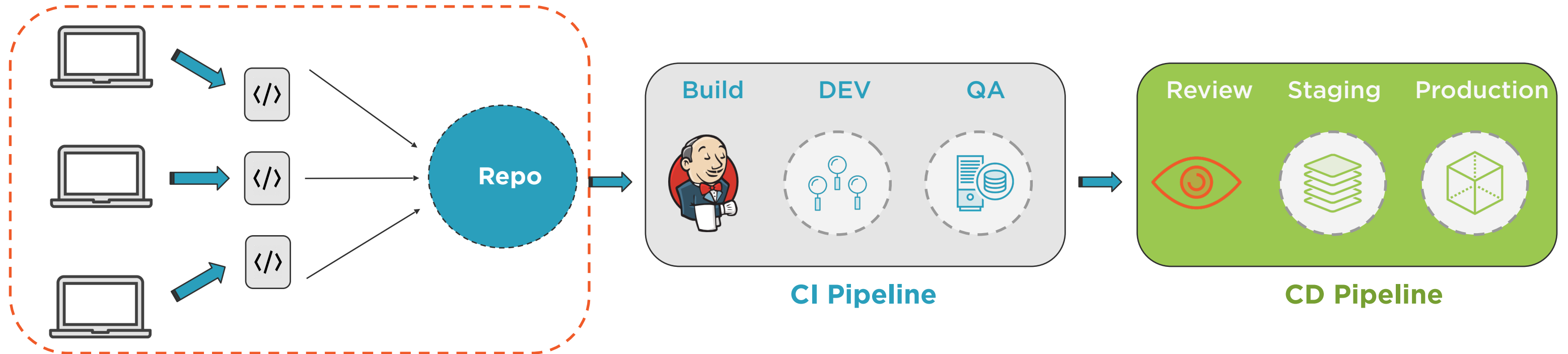
Familiarity with CI and CD

Knowledge of cloud technology

CI & CD Concept

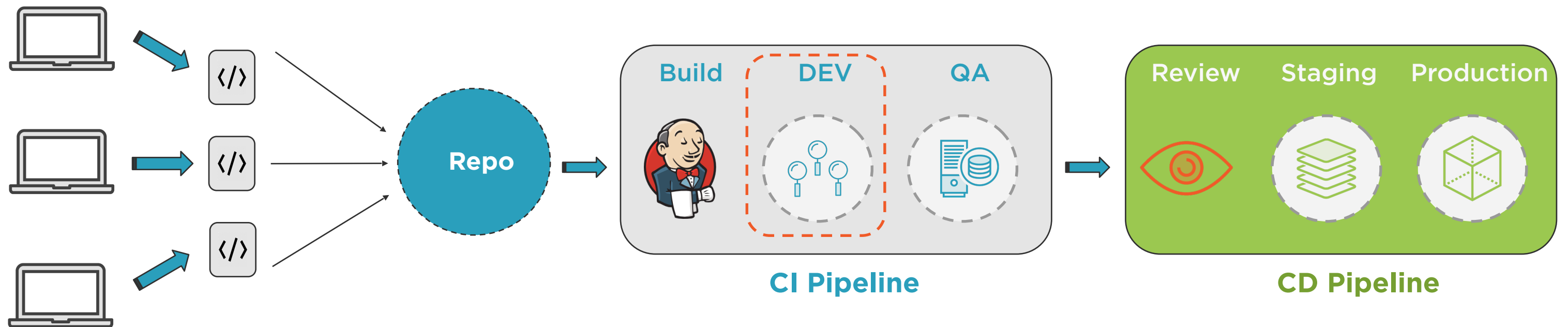
Phase 1

Developers integrate
code frequently

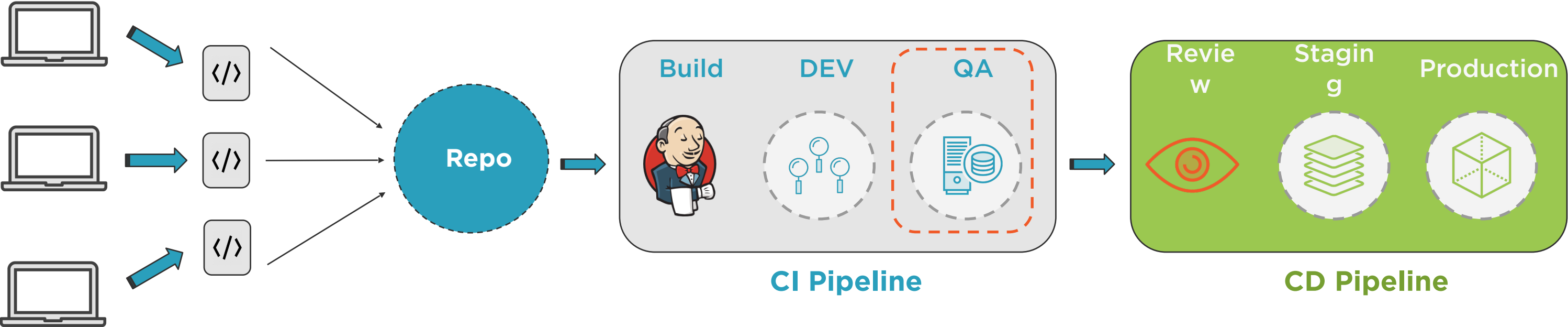


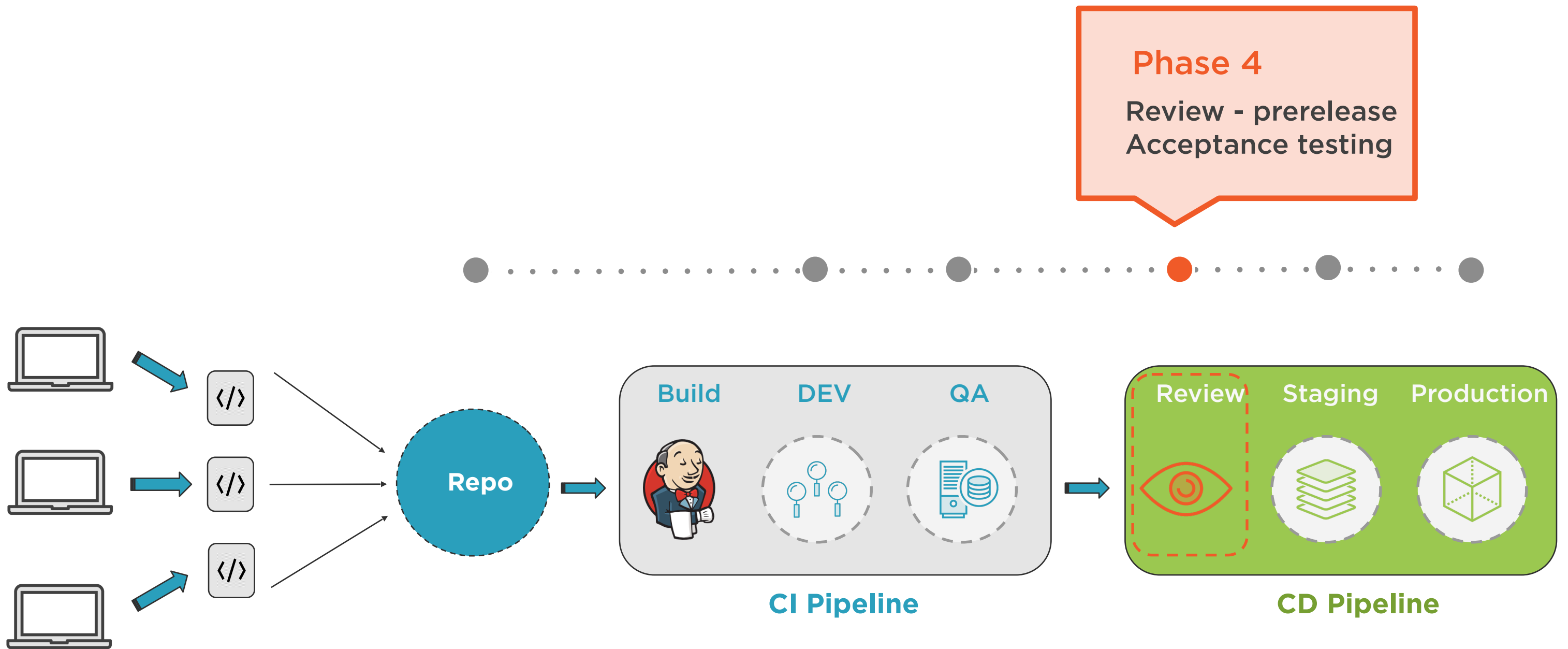
Phase 2

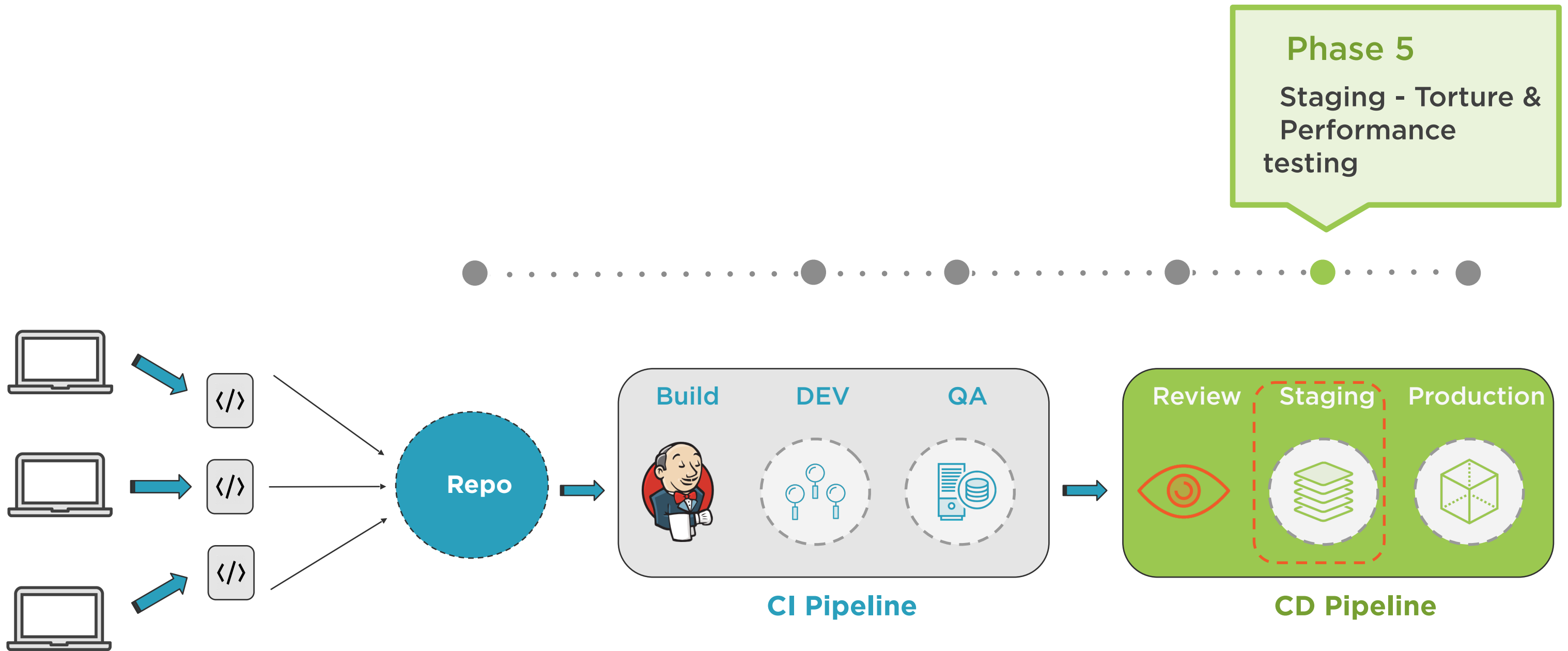
Code deployed to
DEV environment

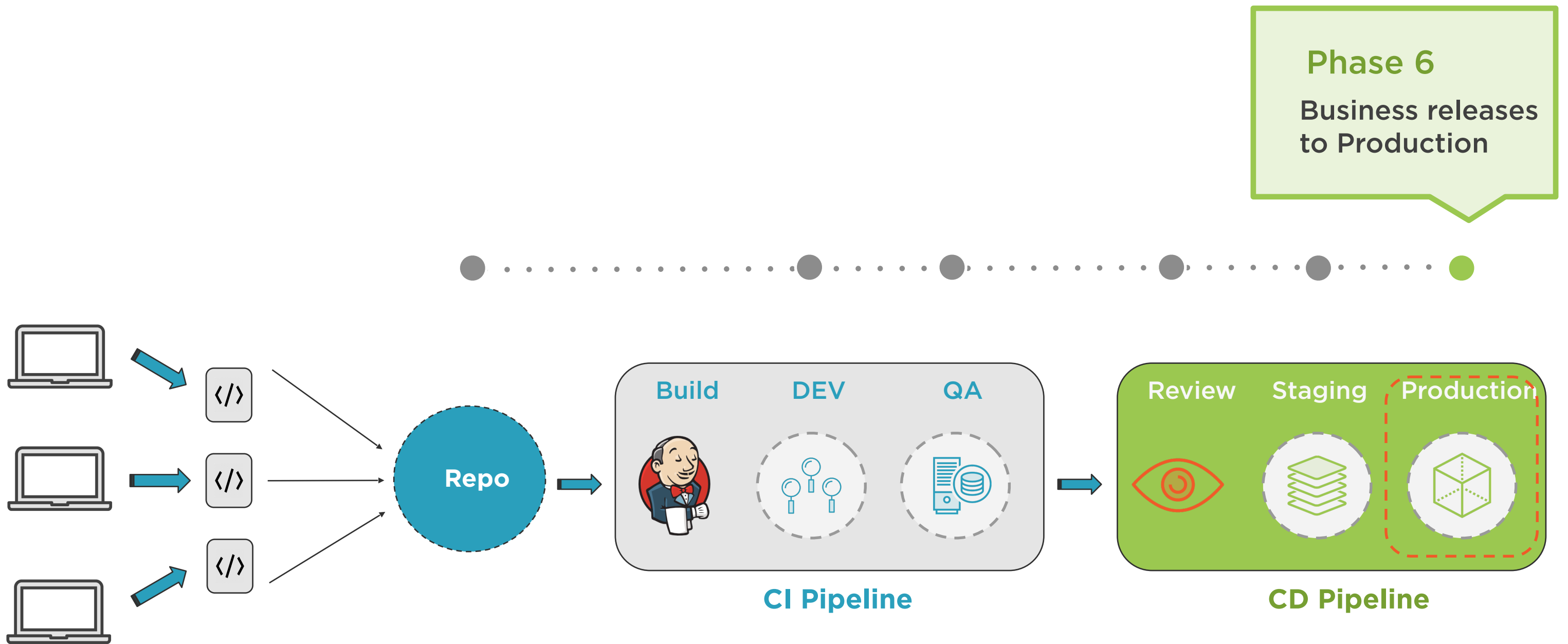


Phase 3
Code deployed to
QA environment









Plugin Architecture

Architecture



Jenkins



jetty://

◆ User Interface

◆ Command Line Interface

◆ Configuration

◆ Plug-in Provider

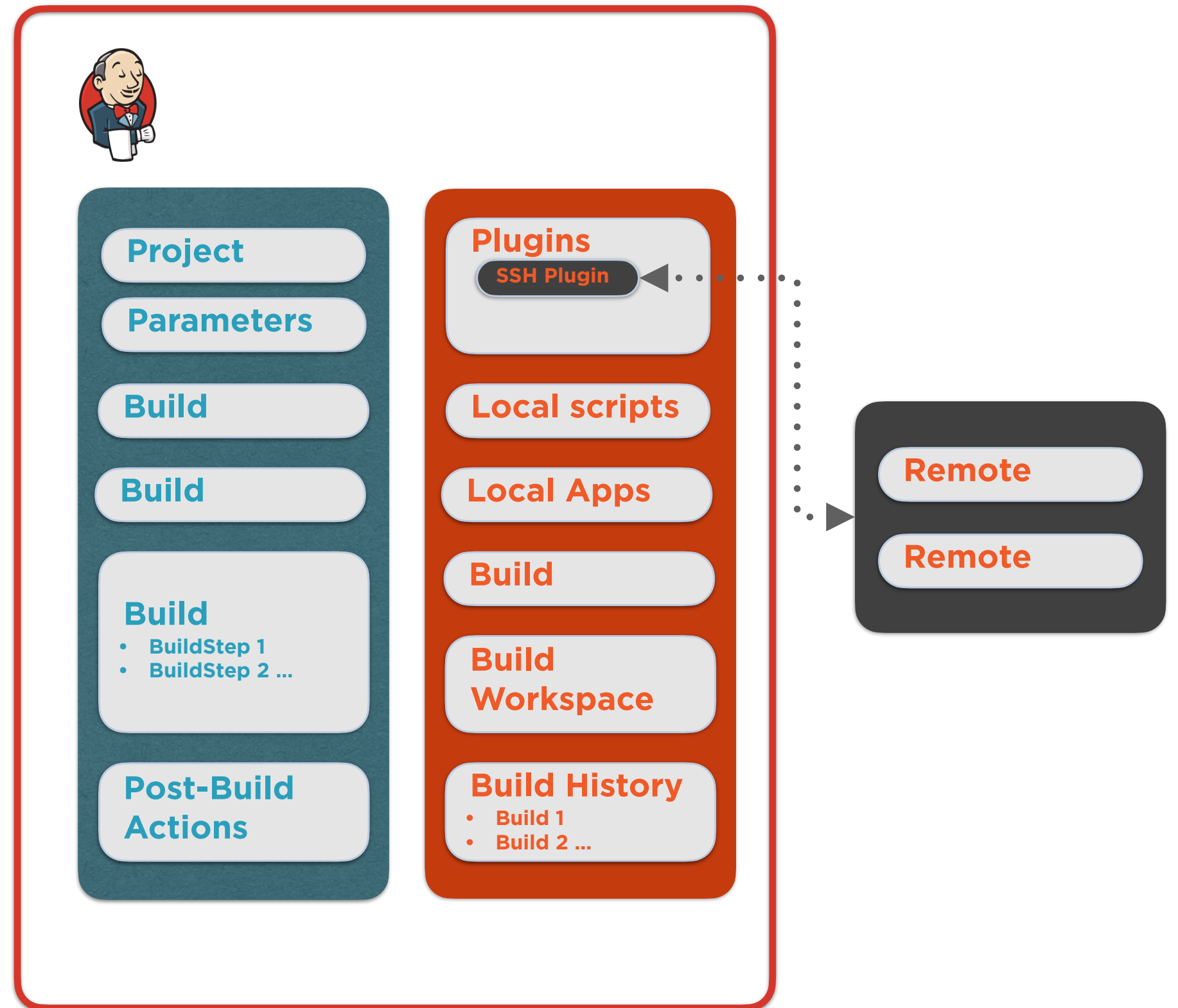
◆ Plug-in Provider

◆ Plug-in Provider

◆ Plug-in Provider

◆ Plug-in Provider

Architecture



Jenkins Extension Points

**Security Models
for Specific Tools**

**Console
Annotators**

**New Post Build
Steps**

**New View
Types**

Status widgets

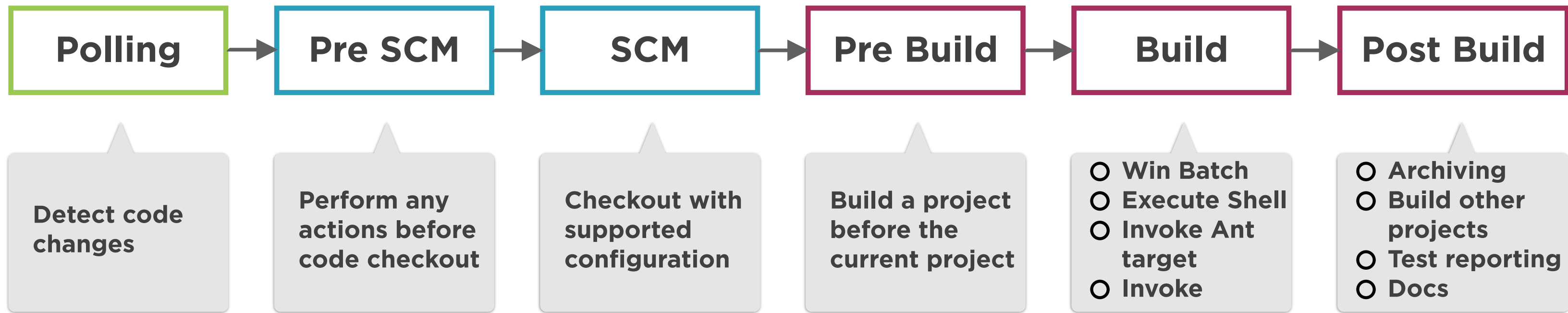
Listeners

**New View
Columns**

**New Build
Steps**

Actions

Default Project (Job) Lifecycle



Deployment Options

Deployment Options



Local Machine



On-premise



In the Cloud



Managed service

Deployment Options

Demo - Create an AMI

Demo



Use CloudFormation to:

Stand up a New EC2 instance

Install Java & Jenkins on the VM

Configure Logging

- Locally
- AWS Cloud

Unlock Jenkins

Install Suggested Core Plugins

Create a Jenkins AMI

Demo - Install Plugins via UI

Demo Notes

Demo - Install Plugins via CLI

Demo Notes

Integration with SCM - GitHub

Inventory

Jenkins on EC2

Java & Maven on EC2

Git on EC2 & Local Workstation

- GitHub Repo
- Sample Java App

Tasks

Brief Intro to Webhooks

Configuring Webhooks in GitHub to detect code push events

Configuring a Jenkins Job to trigger a build when code push changes occur in the GitHub repository

Make code changes and push them upstream to test the Jenkins build trigger

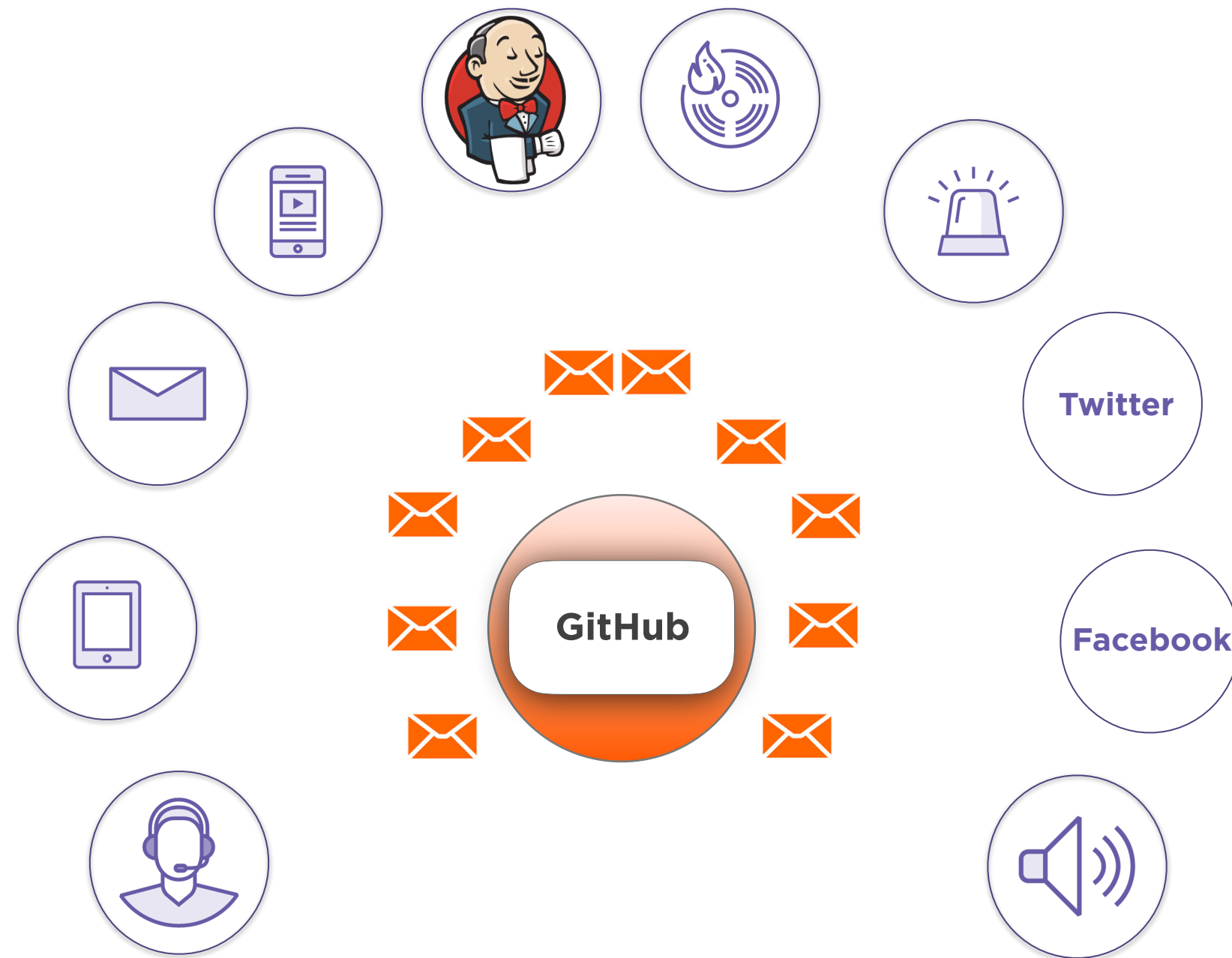
Webhooks

Web callbacks

HTTP API

Reverse API

Webhook Principle



Sequence Diagram



Demo - Integration with SCM - GitHub

Demo



Create Webhook in GitHub

Create a Jenkins Project (Job)

Configure Jenkins Project to:

- Access GitHub Source Repository
- Consume GitHub Webhook AWS

Demo



Be triggered by our new webhook when new code is pushed to the repository

Pull the latest code

Build the source code and produce a deployable artifact

Demo - Integration with SCM - CodeCommit

Demo



Automate plug-in management

Set Jenkins maintenance standards in your organization

Scaling Jenkins

Jenkins Security



Jenkins Security Areas

Authentication

Authorization

Agents

Enabling Jenkins Security

| Prior to 2.214 & LTS 2.222.1 | Post to 2.214 & LTS 2.222.1 |
|--|---|
| <div> Configure Global Security</div> <div><input checked="" type="checkbox"/> Enable security</div> <div>TCP port for JNLP agents <input type="radio"/> Fixed : <input type="text"/> <input type="radio"/> Random <input checked="" type="radio"/> Disable</div> <div>Disable remember me <input type="checkbox"/></div> <div>Access Control</div> <div>Security Realm</div> | <div> Configure Global Security</div> <div>Authentication</div> <div><input type="checkbox"/> Disable remember me</div> <div>Security Realm</div> <div>Security Realm</div> |

Common Security Realms

Web Servers

App Servers

...

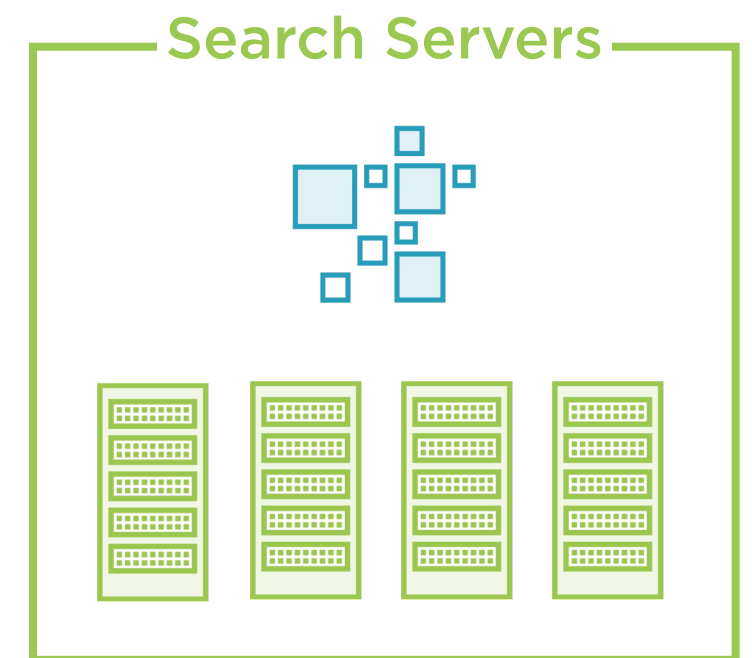
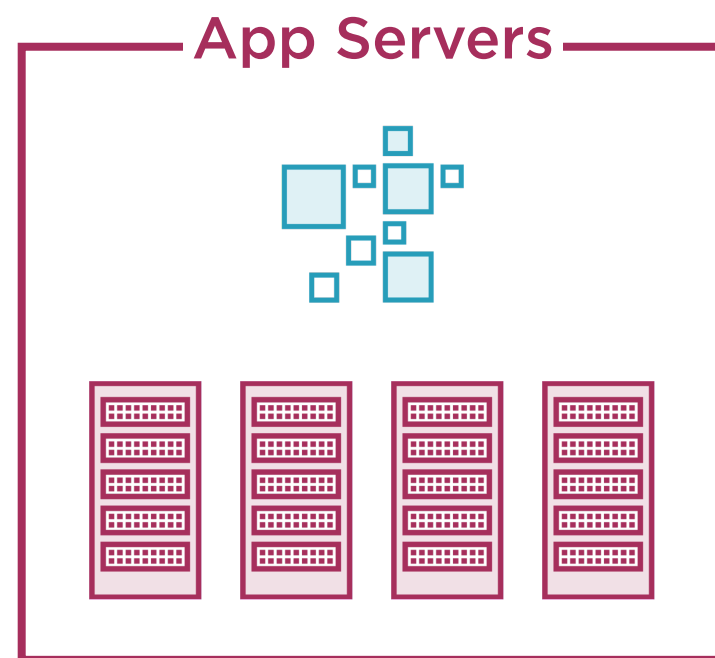
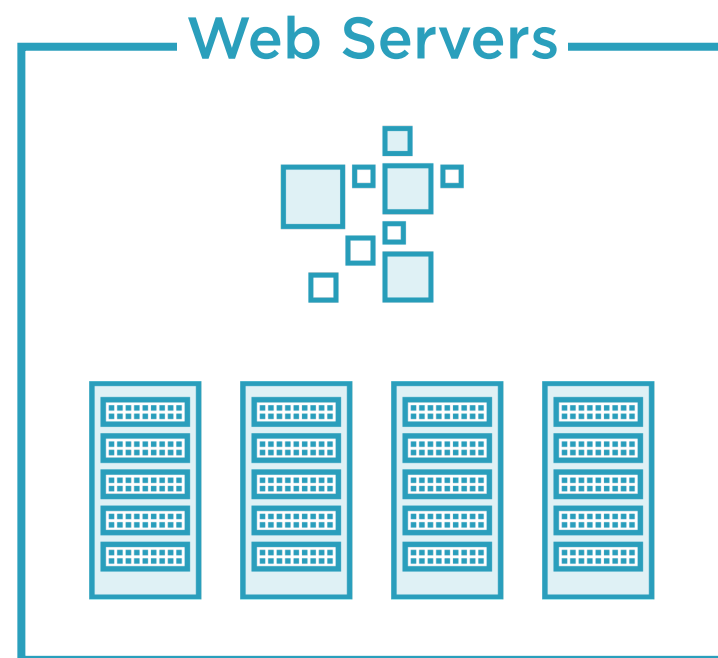
LDAP Solutions

OpenLDAP

**Apache
Directory Server**

**Microsoft Active
Directory**

Security Realms



Master Node Lifecycle Management

Lifecycle



Plugin Versions & External Plugins

- Rolling back a deployed plugin version
- Installing a plugin that is not available via the Update Center

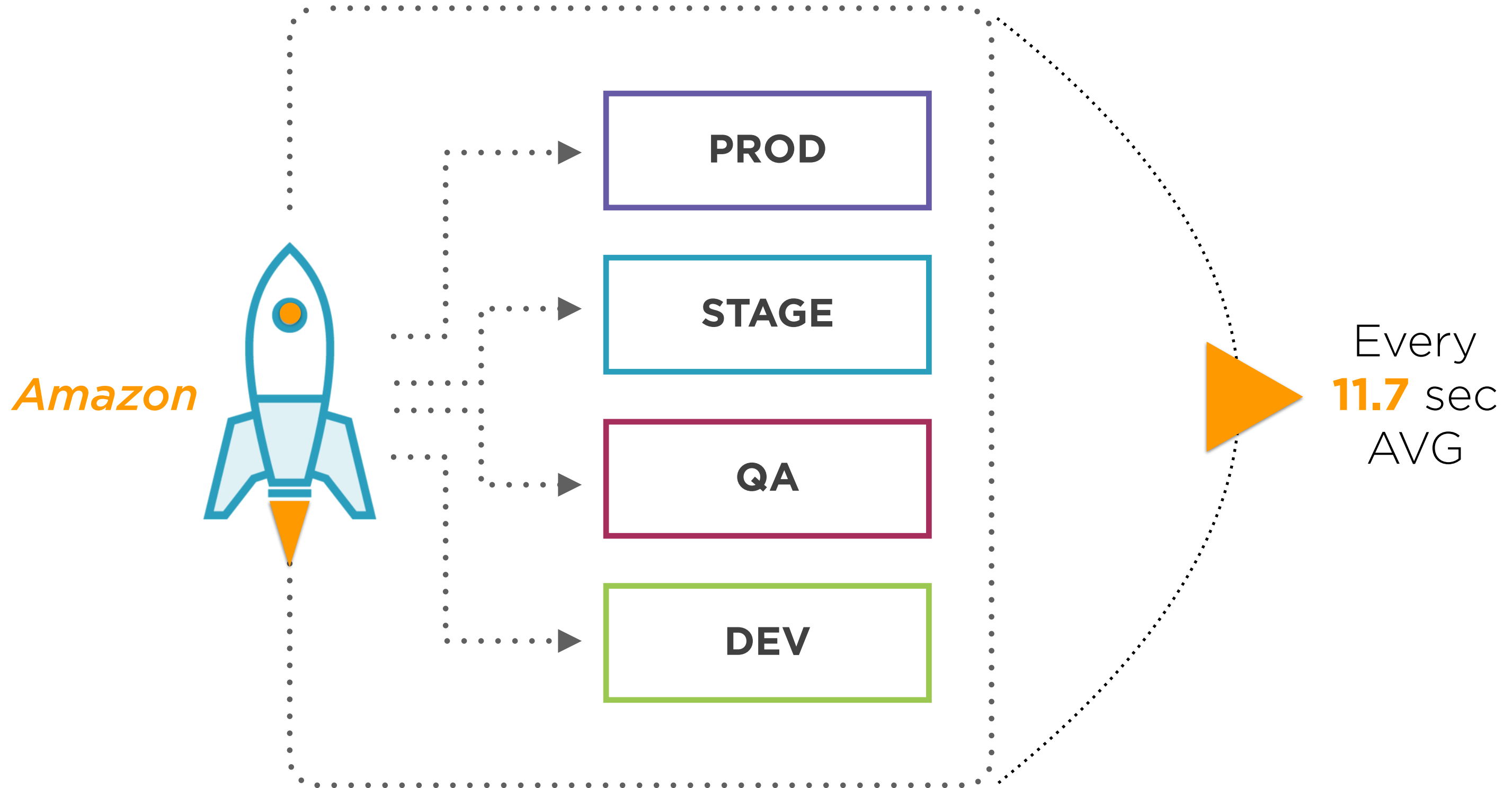
Plugin Updates

Removing Plugins

Removing Old Data

AMI Updates

Release Cycle



Version Control

