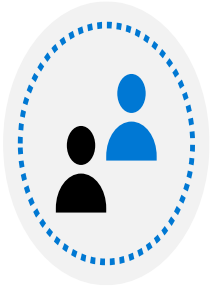
DevOps

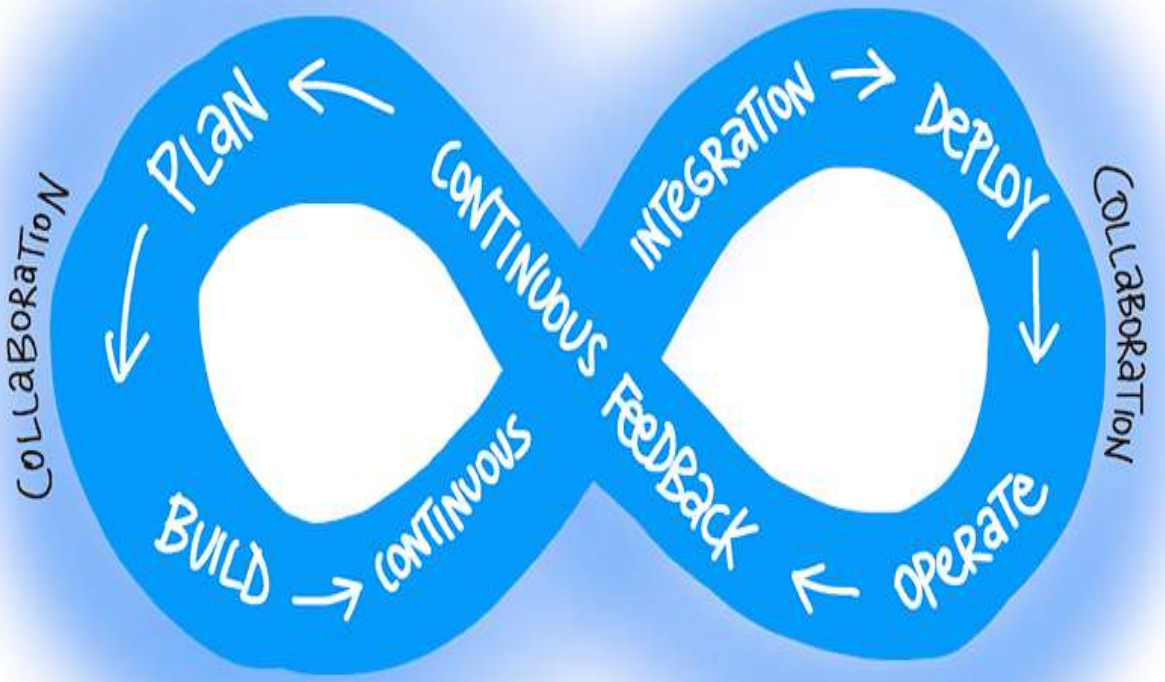
Ravindra Kudache

What is DevOps?

"DevOps is the union of people, process, and products to enable continuous delivery of value to end users."

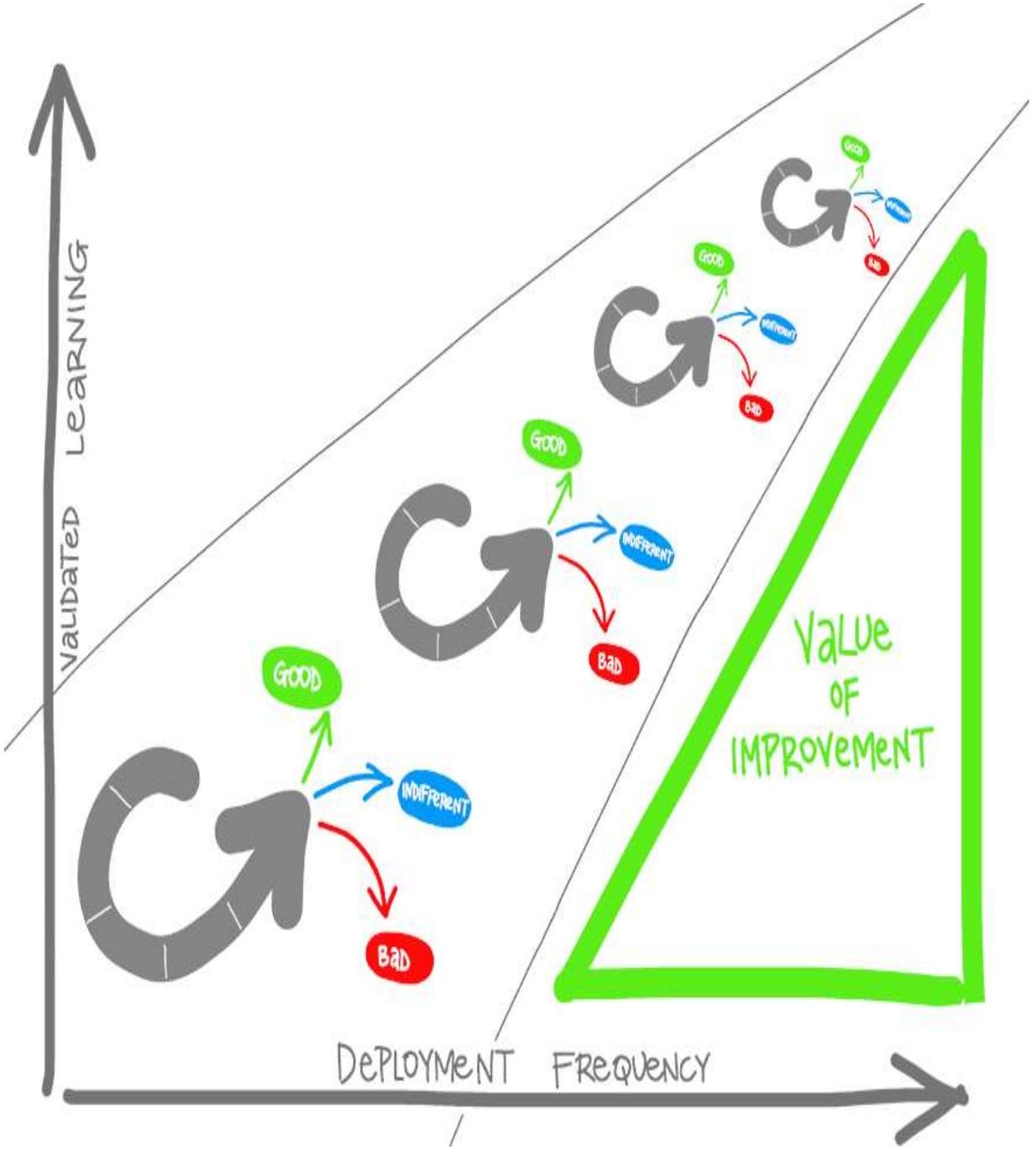
– Donovan Brown, [What is DevOps?]





What is DevOps? (contin

- Understand your cycle time
  - Observe, Orient, Decide, Act (OODA) loop
- Become data-informed
- Strive for validated learning
- Shorten your cycle time
- Optimize validated learning



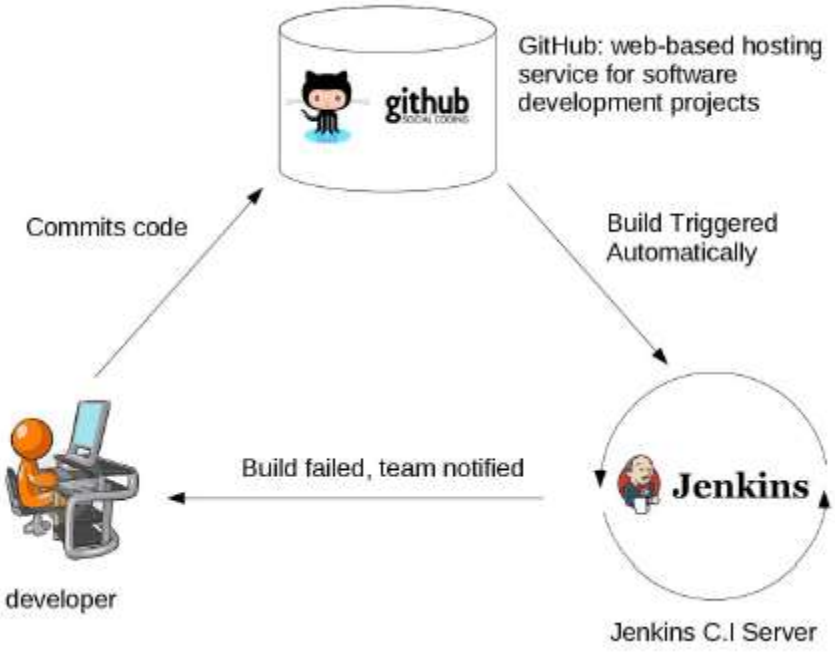
The DevOps journey

- Continuous Integration
- Continuous Delivery
- Version Control
- Agile/lean
- Monitoring and logging
- Cloud
- Infrastructure as Code (IaC)
- Microservices
- Containers
- DevOps may hurt at first

https://docs.microsoft.com/en-us/azure/devops/learn/what-is-devops

Continuous Integration

- What is Continuous Integration?
- Why do we need it?
- Different phases of adopting Continuous Integration



What is Continuous Integration?

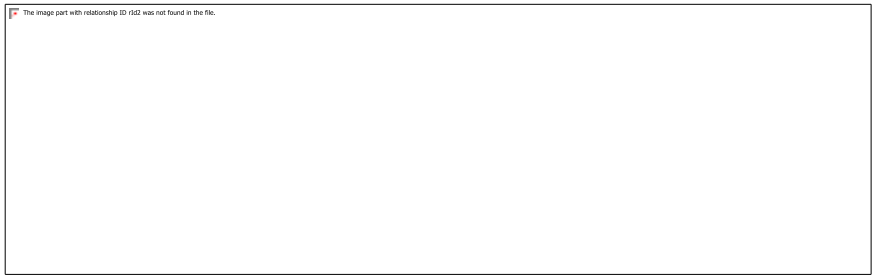
- Developers commit code to a shared repository on a regular basis.
- Version control system is being monitored. When a commit is detected, a build will be triggered automatically.
- If the build is not green, developers will be notified immediately.

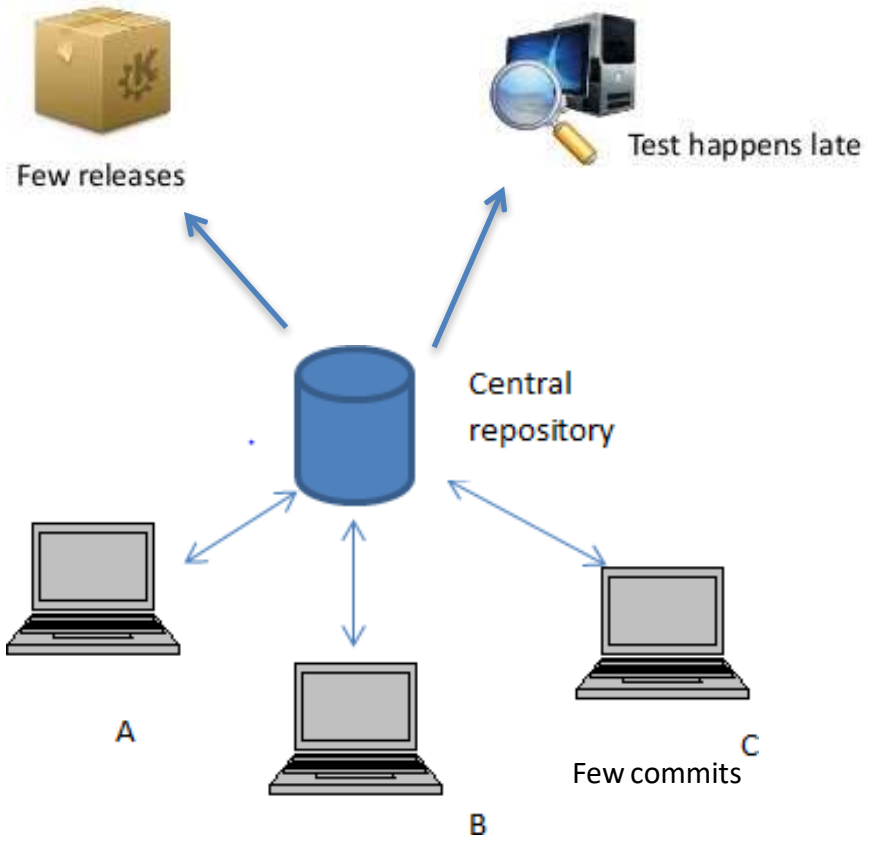
Why do we need Continuous Integration?

- Detect problems or bugs, as early as possible, in the development life cycle.

- Since the entire code base is integrated, built and tested constantly , the potential bugs and errors are caught earlier in the life cycle which results in better quality software.

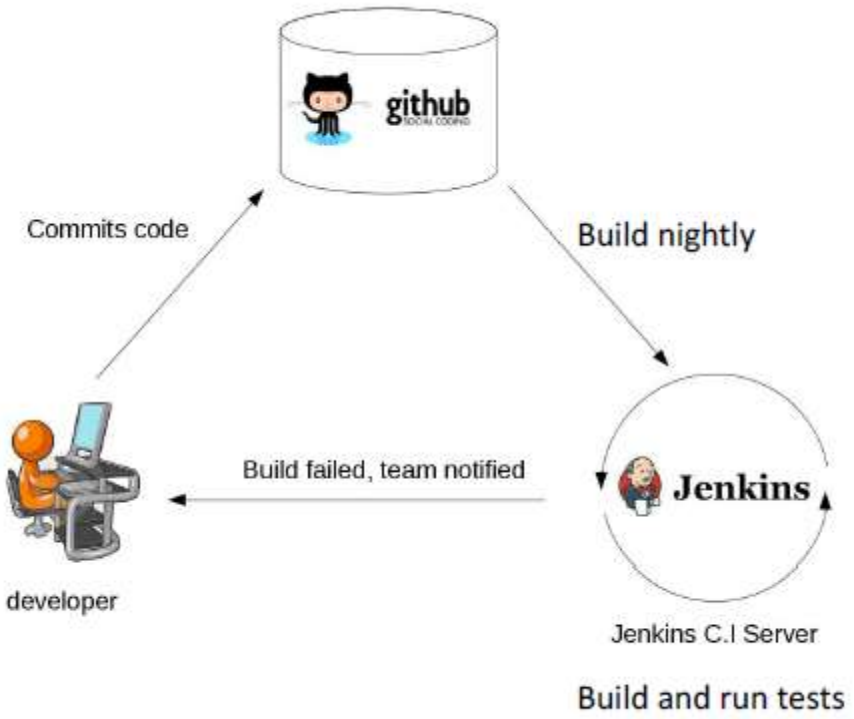
Different stages of adopting Continuous Integration





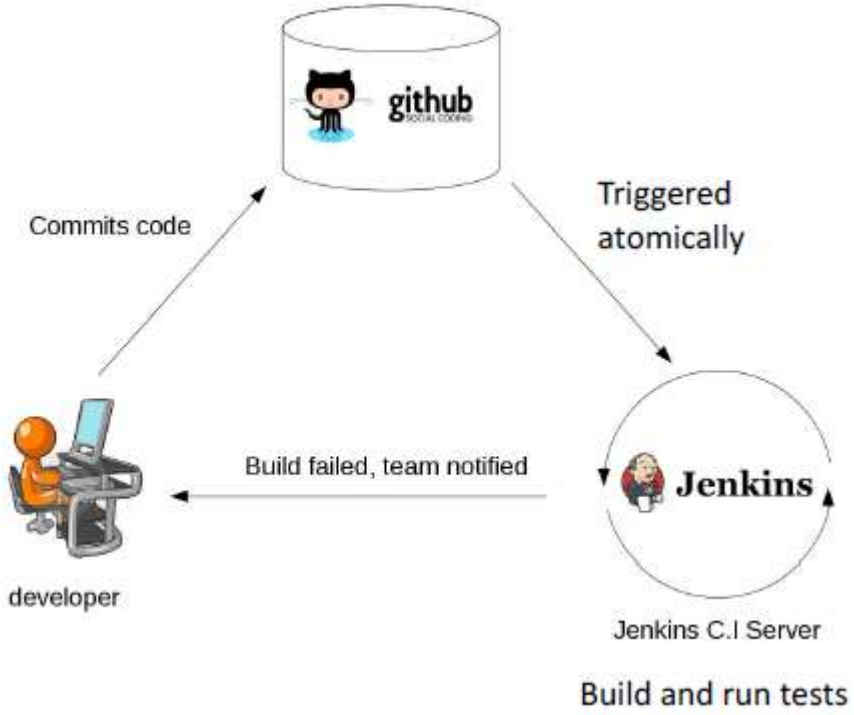
Stage 1:

- No build servers.
- Developers commit on a regular basis.
- Changes are integrated and tested manually.
- Fewer releases.



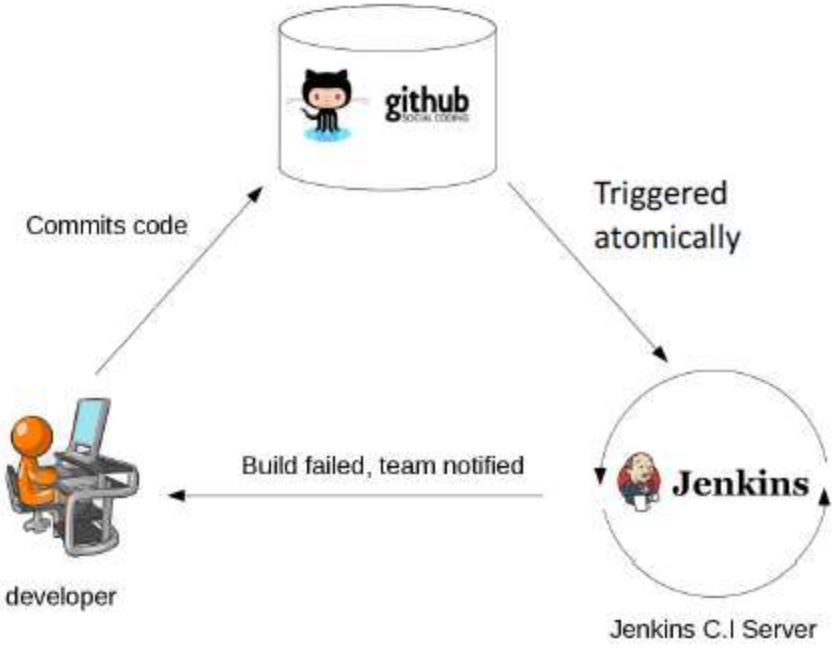
Stage 2:

- Automated builds are scheduled on a regular basis.
- Build script compiles the application and runs a set of automated tests.
- Developers now commit their changes regularly.
- Build servers would alert the team members in case of build failure.



Stage 3:

- A build is triggered whenever new code is committed to the central repository.
- Broken builds are usually treated as a high priority issue and are fixed quickly.



Build, run code quality and code coverage metrics along with tests

Stage 4:

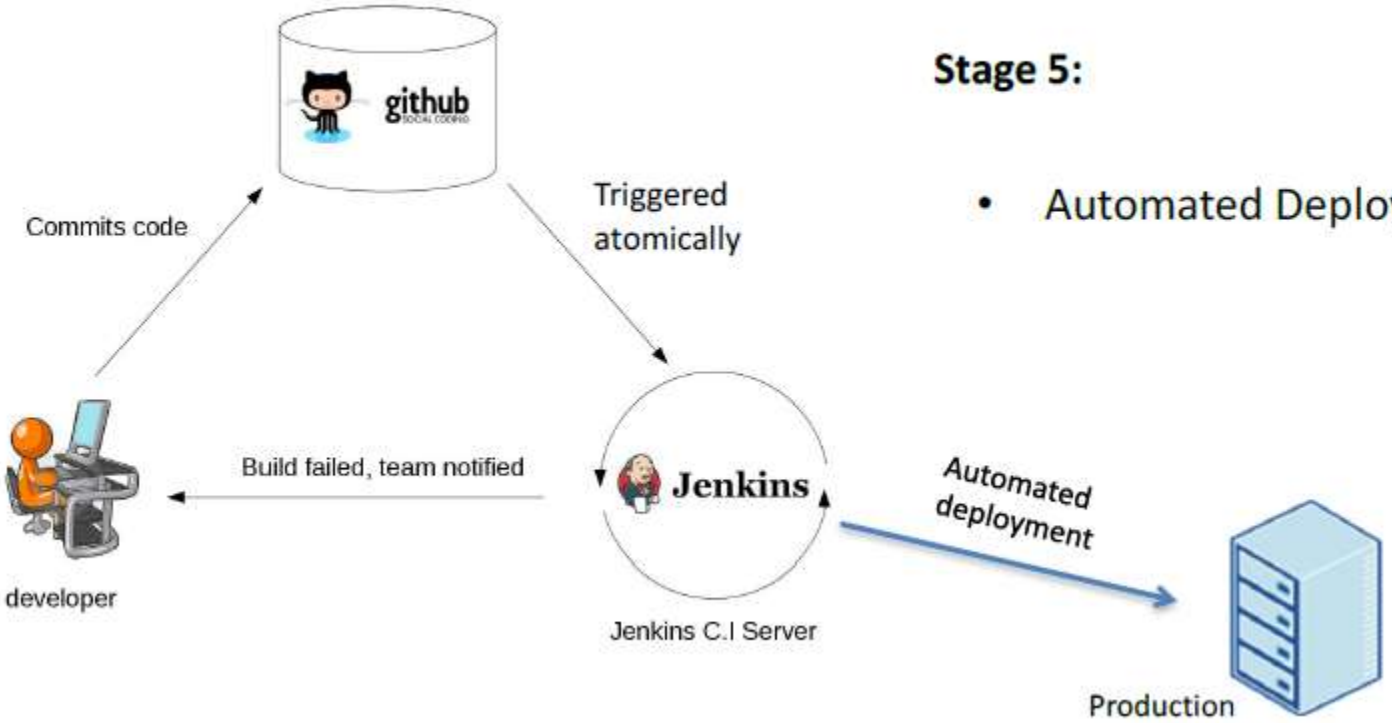
- Automated code quality and code coverage metrics are now run along with unit tests to continuously evaluate the code quality.

Is the code coverage increasing?

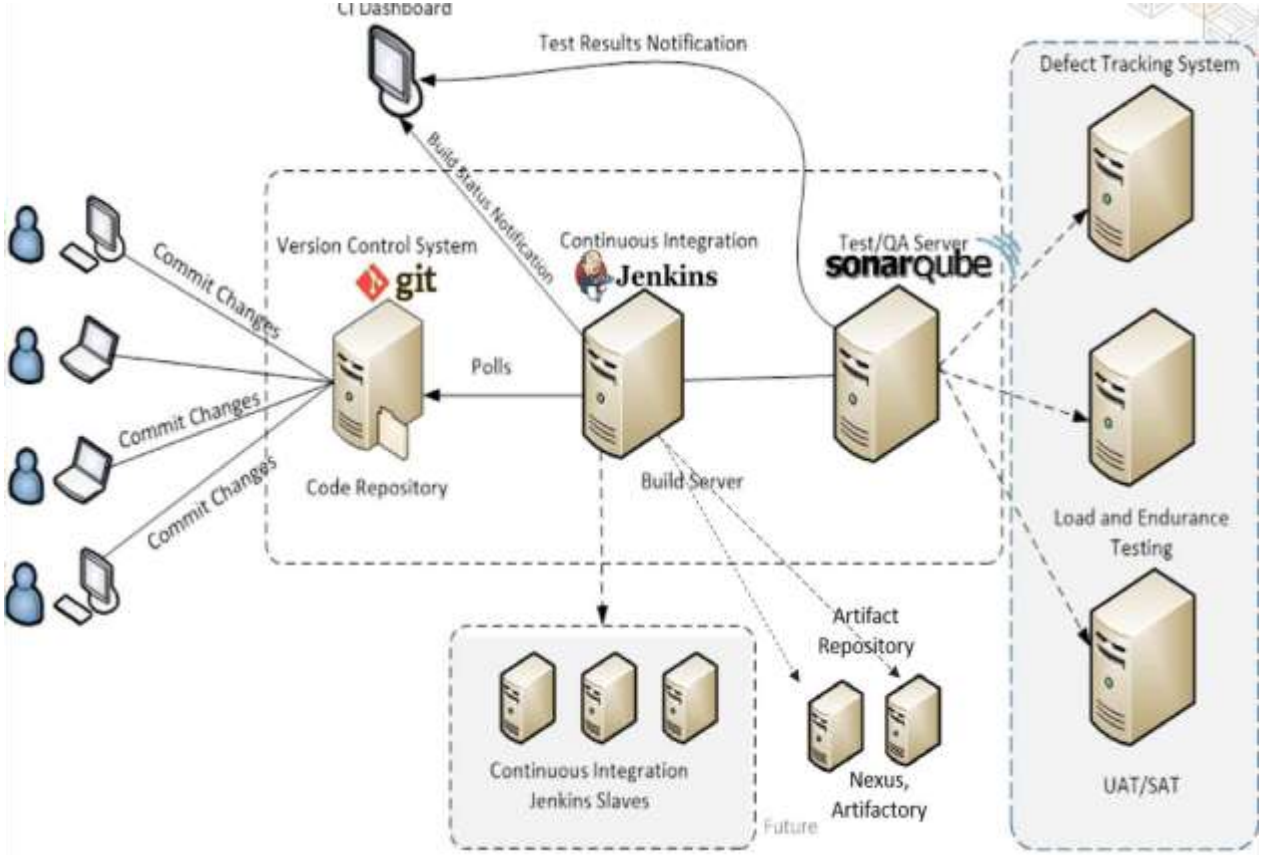
Do we have fewer and fewer build failures?

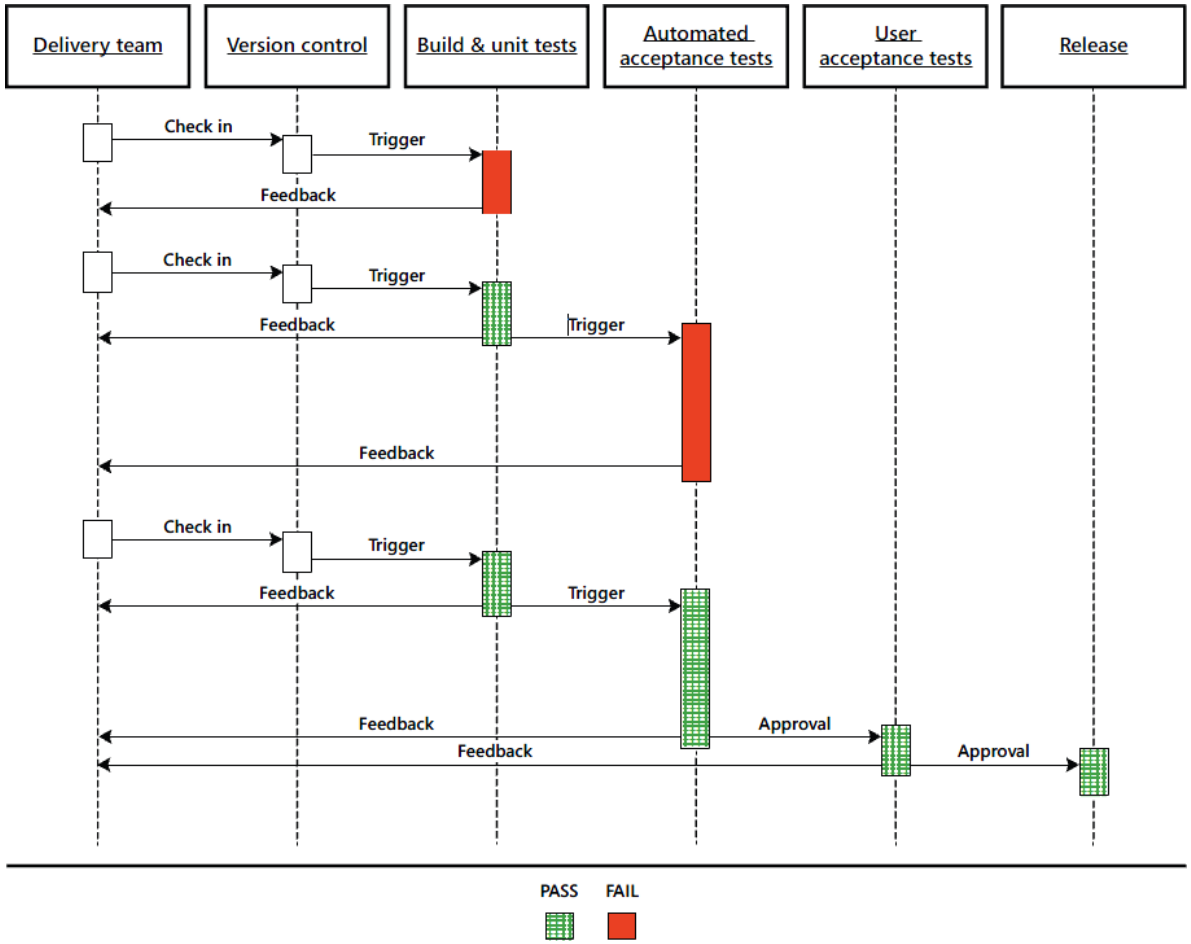
Stage 5:

- Automated Deployment



CI/CD Environment




Delivery team
Version control
Build & unit tests
Automated acceptance tests
User acceptance tests
Release
Check in
Trigger
Feedback
Check in
Trigger
Feedback
Trigger
Feedback
Check in
Trigger
Feedback
Trigger
Feedback
Approval
Feedback
Approval
PASS
FAIL

Continuous Integration

Continuous Delivery

Continuous Deployment

- **Continuous Integration**

  The practice of merging development work with the main branch constantly.

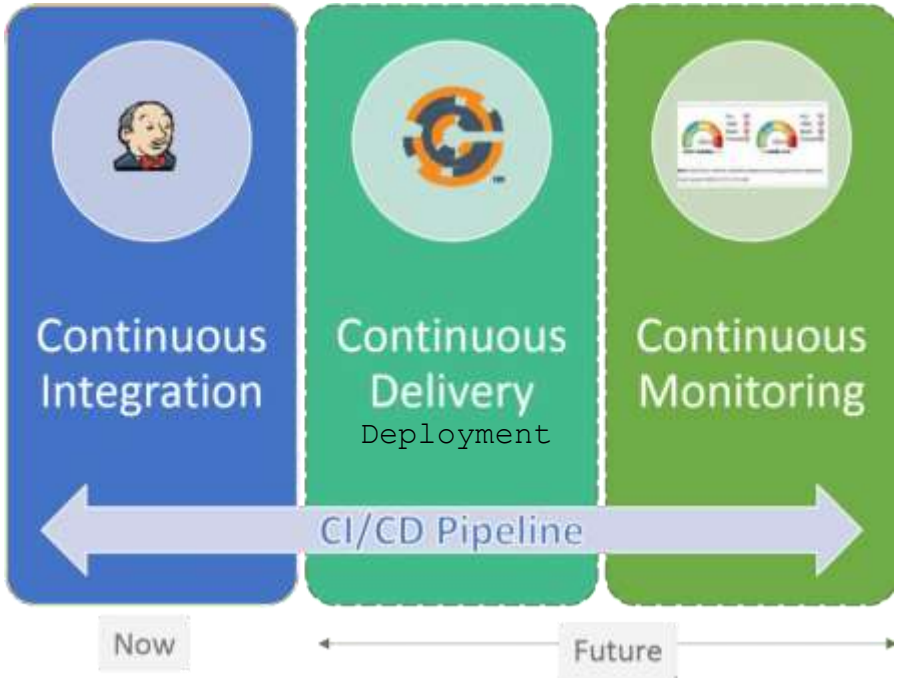
- **Continuous Delivery**

  Continual delivery of code to an environment once the code is ready to ship. This could be staging or production. The idea is the product is delivered to a user base, which can be QAs or customers for review and inspection.

- **Continuous Deployment**

  The deployment or release of code to production as soon as it is ready.

DevOps


Continuous Integration
Continuous Delivery
Deployment
Continuous Monitoring
CI/CD Pipeline
Now
Future

How to implement Continuous Integration?

Non-hosted solutions

Hosted solutions

Continuous Integration is also a mindset

- Fixing broken builds should be treated as a high priority issue for all team members.
- The deployment process should be automated, with no manual steps involved.
- All team members should focus on contributing to high-quality tests because the confidentiality of the CI process highly depends on the quality of the tests.

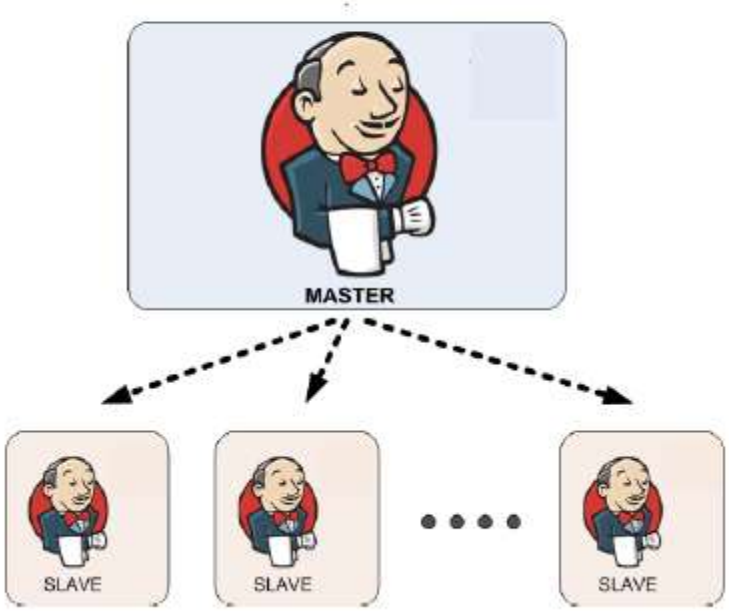
What is Jenkins

- Jenkins is a continuous integration and build server.
- It is used to manually, periodically, or automatically build software development projects.
- It is an open source Continuous Integration tool written in Java.
- Jenkins is used by teams of all different sizes, for projects with various languages.

Why Jenkins is popular

- Easy to use
- Great extensibility
  - Support different version control systems
  - Code quality metrics
  - Build notifiers
  - UI customization

Jenkins' Master and Slave Architecture



Master:

- Schedule build jobs.
- Dispatch builds to the slaves for the actual job execution.
- Monitor the slaves and record the build results.
- Can also execute build jobs directly.

Slave:

- Execute build jobs dispatched by the master.

Install Jenkins

Install GIT and GitHub plugin

Part 2

Install and Configure Maven

Apache Ant

- Widely-used and very well-known build scripting language for Java.
- Flexible, extensible, relatively low-level scripting language.
- An Ant build script is made up of a number of targets, each target performs a particular job in the build process.

Gradle

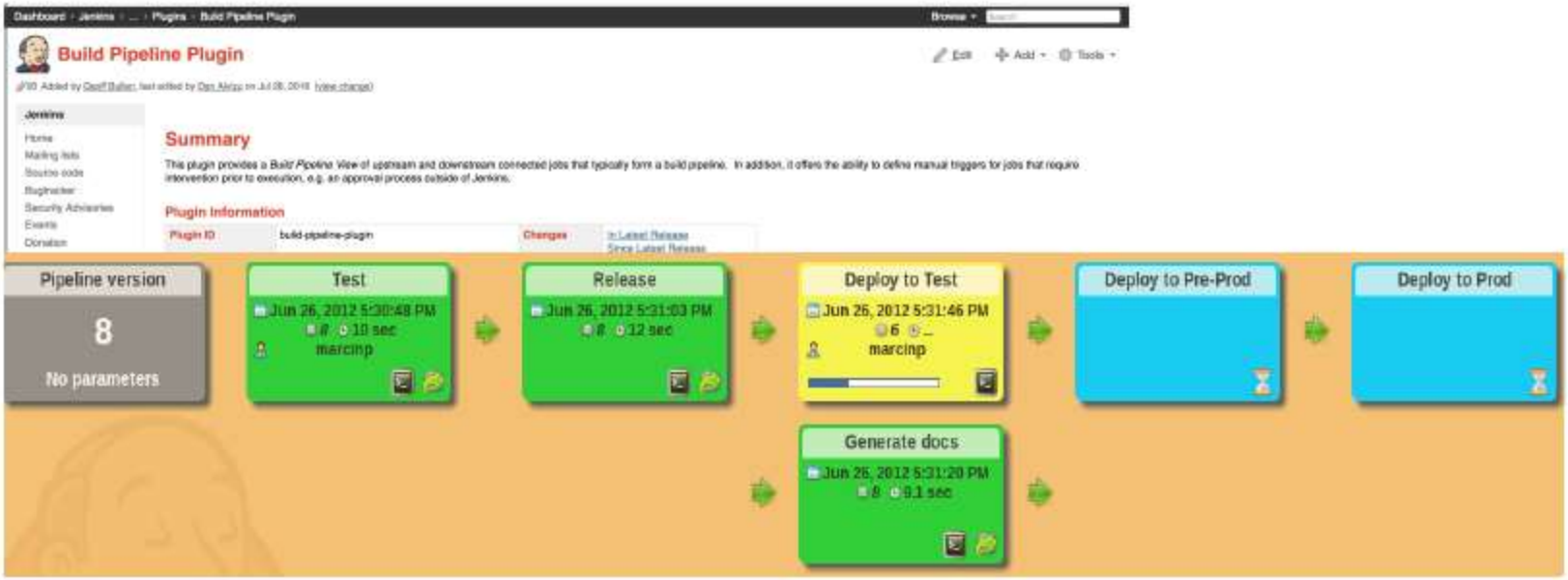
- Gradle is a relatively new open source build tool for the Java Virtual Machine.
- Build scripts for Gradle are written in a Domain Specific Language based on Groovy.
- The concise nature of Groovy scripting lets you write very expressive build scripts with very little code.

Tomcat

Tomcat is an open-source web server and provides a "pure Java” HTTP web server environment in which Java code can run.

Jenkins Build Pipeline

Build Pipeline Plugin



Benefits of a code-based pipeline

- Version control
- Best Practices
- Less error-prone execution of jobs
- Logic-based execution of steps

Sample Jenkinsfile

```
pipeline {
    agent any
    stages {
        stage ('Initialize') {
            steps {
                sh '''
                    echo "PATH = ${PATH}"
                    echo "M2_HOME = ${M2_HOME}"
                '''
            }
        }
        stage ('Build') {
            steps {
                echo 'Hello World!'
            }
        }
    }
}
```

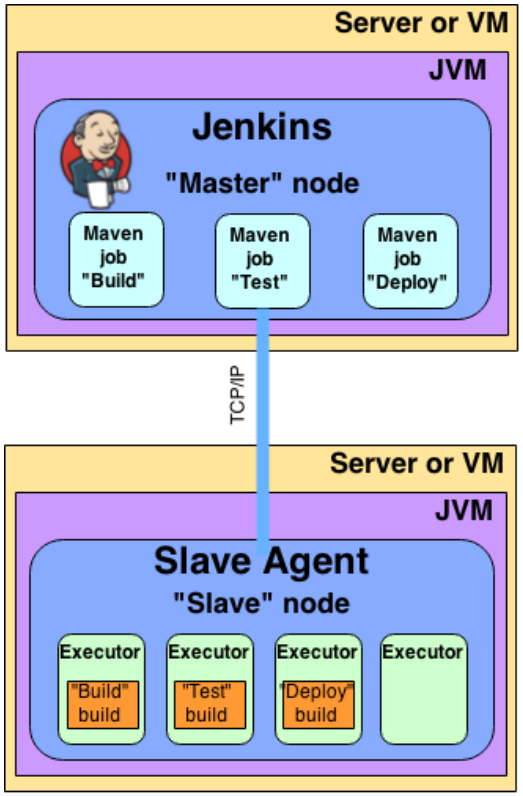
Additional automation

- Setup Git repository polling
- Deployment to our tomcat servers
- We will setup tasks to run in parallel

Steps

- **Step 1: Configure securit groups for Tomcat servers and create key pairs**.
- **Step 2:** Provision instances to staging and production environments.
- **Step 3:** Install and run Tomcat on created instances.
- **Step 4:** Fully automate our existing Jenkins pipeline.

Jenkins Slave Agent

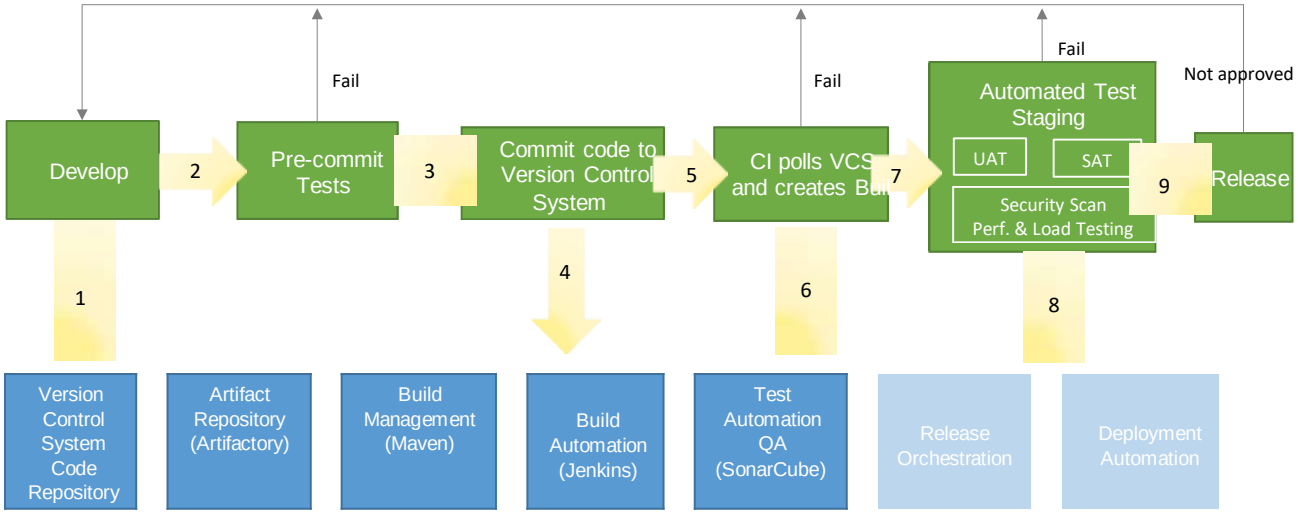


Build Orchestration: Jenkins

- Continuous integration system
- Enable automated build and test process
- Can monitoring executions of externally-run jobs, such as cron jobs and procmail jobs…
- Dependency tracking, allowing file finger printing and tracking for example which build is using which version of jars…
- Generates list of changes made to build from Subversion
- Distributed build/test

- Jenkins is a build orchestration, CI software
  - building/testing software projects continuously
  - monitoring executions of externally-run jobs
- FishEye allows you to extract information from your source code repository and display it in sophisticated reports.
- Crucible allows you to request, perform and manage code reviews.
- Subversion centralized version control system
- Sonar is a quality management platform for analyzing and measuring source code quality.

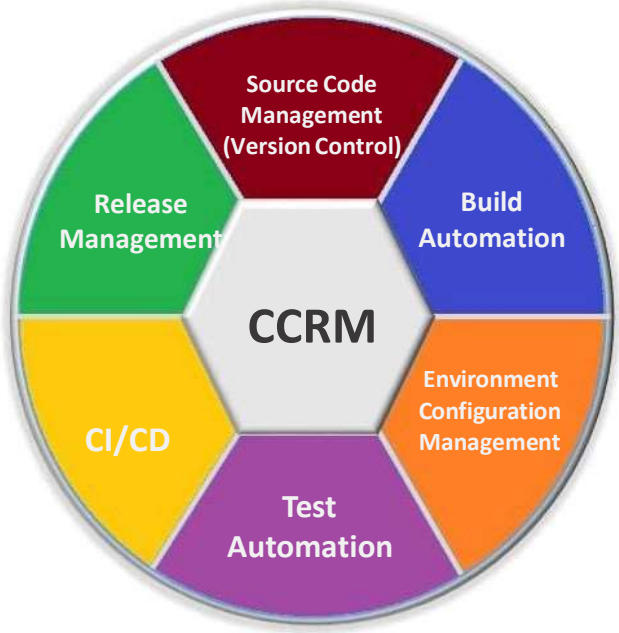
CI/CD Pipeline: Functional Architecture



An Automated, Integrated and End to Ent CCRM

Git, GitLab, GitHub Enteprise,

SVN…



Maven, Ant, Gradle

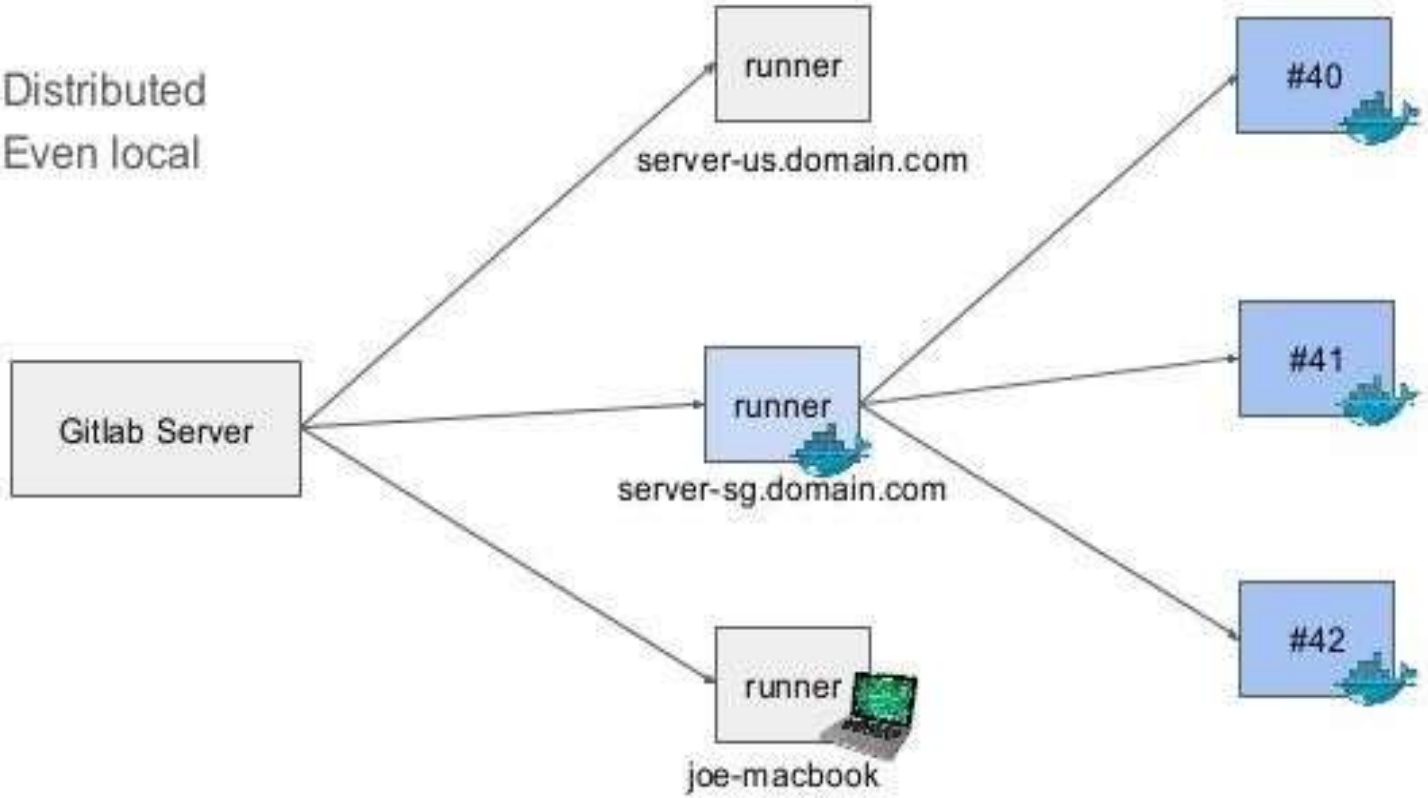
Nexsus, Artifactory

Jenkins…

Junit, Test NG, Cucumber, Selenium, JMeter, SoapUI, LoadRunner…

Runners

- Distributed
- Even local



END