



Zero to Continuous Delivery with Jenkins Pipeline and Blue Ocean

Liam Newman

Introduction

Introduction



- Liam Newman
 - CI/CD Practitioner
 - Jenkins Contributor
 - Word Engineer
- @bitwiseman
- bitwiseman@gmail.com

Continuous Delivery

Jenkins Pipeline

Jenkins Pipeline



- Pipeline as Code
 - Capture the entire continuous delivery process
 - Check a Jenkinsfile into your source repo
- jenkins.io/doc/book/pipeline

Blue Ocean

✓ dropspace / whimsy #1

PipelineChangesTestsArtifacts

Branch: master12m 48sNo changes
Commit: e41c4474 days ago

Build

Browser Tests

Static Analysis

Deploy

✓

✓

Chrome

✓

Firefox

✓

Internet Explorer

✓

Safari

✓

✓

Steps - Deploy

✓

> Shell Script

8s

✓

> Archive artifacts

<1s

✓

> General Build Step

<1s

✕ [dropspace / whimsy #1](#)

Pipeline

Changes

Tests

Artifacts

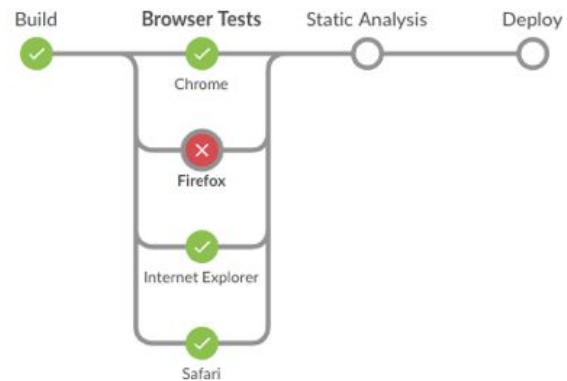


Branch: updated-with-more-whimsy 9m 55s

No changes

Commit: be2afe6

5 days ago



Steps - Firefox



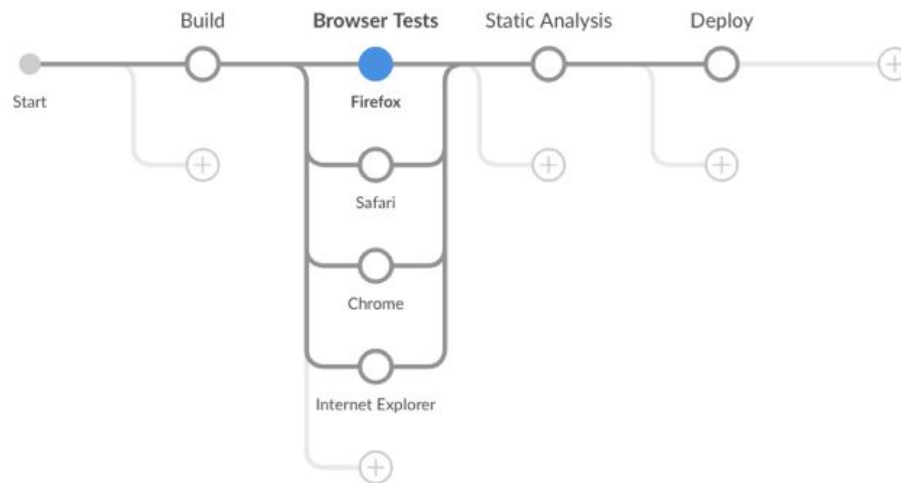
✓	> Shell Script	<1s
✓	> Shell Script	4s
✕	✓ Shell Script	2s
<pre>1 [sy_updated-with-more-whimsy-ELATQKS2L5HL6MMZ6Ww7LQXG5FBD27W40Y00LQXJ43F7HPIIVFNQ] Running shell script 2 + mvn clean bob 3 [INFO] Scanning for projects... 4 [INFO] 5 [INFO] ----- 6 [INFO] Building app-store 0.0.1-SNAPSHOT 7 [INFO] ----- 8 [INFO] 9 [INFO] BUILD FAILURE 10 [INFO]</pre>		



dropspace / whimsy / master

Cancel

Save



← Firefox



Shell Script

```
echo 'setting up selenium environment'
./xvfbsh
```

Shell Script

```
./run-firefox.sh
```

[+ Add step](#)

Blue Ocean



```
docker run -p 8080:8080 -u root \  
    -v /var/run/docker.sock:/var/run/docker.sock \  
    jenkinsci/blueocean
```


Zero

A basic Java application



- bitwise-jenkins/jhipster-sample-app
- Tools
 - Maven
 - Node / Gulp
 - Gatling
 - Docker

Planning the pipeline

Planning the pipeline



- Stages desired:
 - Build
 - Test
 - Unit
 - Performance
 - Front-end
 - Static Analysis
 - Deployment

Blue Ocean Pipeline Editor

Build

Build



- Don't reinvent your build system in Jenkins Pipeline
 - Think of Pipeline as glue
- Goal:
 - Perform a reproducible build
 - Create an artifact which can be used later
 - To run tests against
 - To deploy to an environment
- In this case, we're using Maven.

Build



```
pipeline {  
  agent any  
  stages {  
    stage('Build') {  
      steps {  
        sh 'mvn'  
      }  
    }  
  }  
}
```

Build



```
stage('Build') {  
    agent {  
        docker {  
            image 'maven:3-alpine'  
            args '-v /root/.m2:/root/.m2'  
        }  
    }  
    /* .. */  
}
```

Build



```
stage('Build') {  
    /* .. */  
    steps {  
        sh './mvnw -B clean package'  
        stash name: 'war', includes: 'target'  
    }  
}
```

Test

Test



```
pipeline {  
  agent any  
  stages {  
    stage('Backend Unit Test') {  
      steps {  
        sh './mvnw -B test'  
      }  
    }  
  }  
}
```


Test



```
stage('Backend Unit Test') {  
    agent {  
        docker {  
            image 'maven:3-alpine'  
            args '-v /root/.m2:/root/.m2'  
        }  
    }  
    /* .. */  
}
```

Test



```
stage('Backend Unit Test') {  
    /* .. */  
    steps {  
        unstash 'war'  
        sh './mvnw -B test'  
        junit '**/surefire-reports/**/*.xml'  
    }  
}
```

Test



```
stage('Backend Performance Test') {  
    /* .. */  
    steps {  
        unstash 'war'  
        sh './mvnw -B gatling:execute'  
    }  
}
```

Test



```
stage('Backend') {  
  steps {  
    parallel(  
      'Unit' : {  
        unstash 'war'  
        sh './mvnw -B test'  
        junit '**/surefire-reports/**/*.xml'  
      },  
      'Performance' : {  
        unstash 'war'  
        sh './mvnw -B gatling:execute'  
      })  
    }  
  }  
}
```

Test



```
stage('Frontend Test') {  
  agent { docker 'node:alpine' }  
  steps {  
    sh 'yarn install'  
    sh 'yarn global add gulp-cli'  
    sh 'gulp test'  
  }  
}
```

Analyze

Build



```
pipeline {  
  agent any  
  stages {  
    stage('Analyze') {  
    }  
  }  
}
```

Analyze



- Static analysis
- Code coverage checks
- Quality scanning
 - FindBugs
 - PMD
 - etc

jenkins.io/blog/2017/04/18/continuousdelivery-devops-sonarqube/

Deploy

Deploy



- Don't reinvent your deployment system in Jenkins Pipeline
 - Think of Pipeline as glue
- "Deployment" may have different meanings
 - Deploying to AWS/Azure
 - Deploying to a physical datacenter
 - Uploading to an app store
 - Uploading to an internal artifact server

Deploy



```
pipeline {  
  agent any  
  stages {  
    stage('Deploy') {  
    }  
  }  
}
```


Deploy



```
stage('Deploy') {  
  steps {  
    sh './deploy.sh'  
  }  
}
```

Build

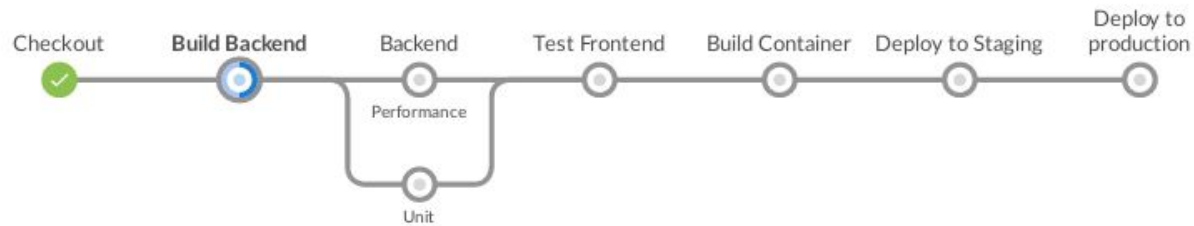


 **jhipster-sample-app #3**

PipelineChangesTestsArtifacts

Branch: master 15s No changes

Commit: — -



Steps - Build Backend



✓	> General SCM	1s
✓	> Shell Script	<1s
✓	> Restore files previously stashed	<1s
○	✓ Shell Script	6s
<pre>1 [jhipster-sample-app_master-CCGQKE2NXylimBF45LPMXCUIHIX7XAHK5QIZCAFIJLPYY7EKRN7NQ@2] Running shell script 2 + ./mvnw -B clean compile 3 [INFO] Scanning for projects... 4 [INFO] 5 [INFO] ----- 6 [INFO] Building Jhipster Sample Application 0.0.1-SNAPSHOT 7 [INFO] -----</pre>		

Test

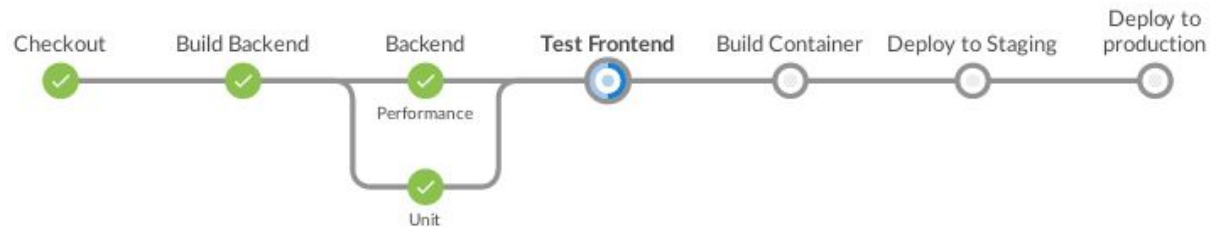


jhipster-sample-app #4

PipelineChangesTestsArtifacts

Branch: master 5m 31s No changes

Commit: — -



Steps - Test Frontend



✓	> General SCM	< 1s
✓	> Shell Script	< 1s
✓	> Restore files previously stashed	< 1s
○	▼ Shell Script	6s
<pre>1 [jhipster-sample-app_master-CCGQKE2NXylimBF45LPMXCUIHIX7XAHK5QIZCAFIJLPPYY7EKRN7NQ@2] Running shell script 2 + yarn install 3 yarn install v0.23.2 4 [1/4] Resolving packages... 5 warning Integrity check: Lock files don't match 6 [2/4] Fetching packages...</pre>		

Deploy



✓ jhipster-sample-app #2

Pipeline

Changes

Tests

Artifacts



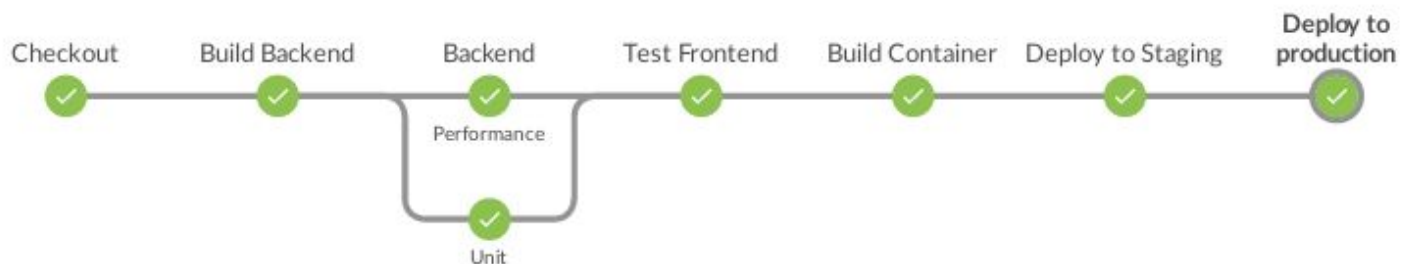
Branch: master

🕒 11m 42s

No changes

Commit: 1fd84c9

🕒 2 minutes ago



Steps - Deploy to production



✓	> General SCM	<1s
✓	> Wait for interactive input	5m 12s
✓	> Print Message	<1s

Other considerations

Notifications



- Feedback to the team is a key part of continuous delivery
- Jenkins Pipeline can be used for:
 - Sending email
 - Sending messages to:
 - HipChat
 - Slack
 - Updating JIRA tickets

jenkins.io/node/tags/notifications/

Manual Promotion



```
stage('Deploy') {  
  steps {  
    input message: 'Deploy to production?',  
           ok: 'Fire zee missiles!'  
  
    sh './deploy.sh'  
  }  
}
```

Manual Promotion

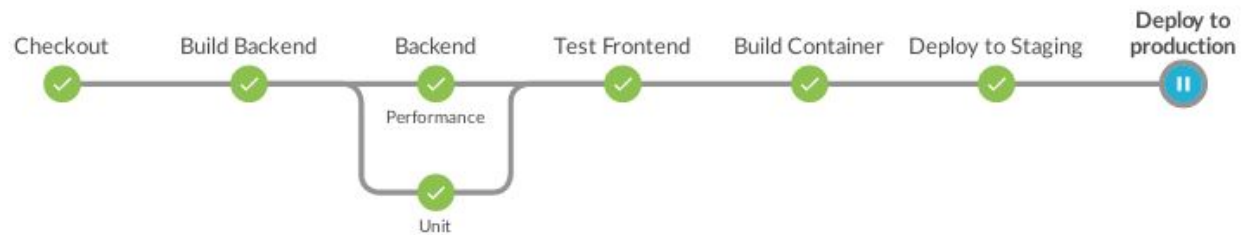


|| jhipster-sample-app #2

PipelineChangesTestsArtifacts⏵⚙️🔗✕

Branch: master ⌚ 17m 20s No changes

Commit: 1fd84c9 ⌚ -



Steps - Deploy to production



✓	> General SCM	< 1s
	▼ Wait for interactive input	4m 57s

Deploy to production?

Fire zee missiles!

Abort

Zero to Continuous Delivery with Jenkins Pipeline and Blue Ocean

Questions?

Resources



- jenkins.io/doc/book/pipeline
- jenkins.io/projects/blueocean
- Docker containers
 - `jenkinsci/jenkins:lts-alpine`
 - `jenkinsci/blueocean`
- jenkinsci-users@googlegroups.com