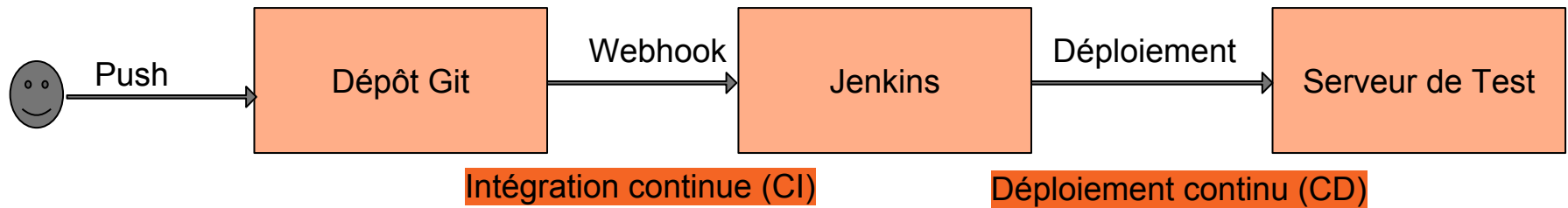

Intégration continue et déploiement continu avec Jenkins

par James Kokou GAGLO
DevOps @peopleinput

Avril 2017

Flow



Configuration de Jenkins : Plugins



Configuration globale des outils

Configuration Maven

- Maven integration

Configuration de Jenkins : Maven

Maven

Installations Maven

 Maven

Nom

M3



Install automatically



Install from Apache

Version

3.3.9



Supprimer un installateur

Jenkins: Créer un projet de type Multibranch Pipeline



Multibranch Pipeline

Creates a set of Pipeline projects according to detected branches in one SCM repository.

Si vous voulez créer un nouvel item à partir d'un autre, vous pouvez utiliser cette option:

Jenkins: Configuration du projet

Branch Sources



Git

Project Repository

`https://github.com/freemanpolys/spring-boot-labs.git`

Build Configuration

Mode

by Jenkinsfile

See Multibranch Pipeline Tutorial

Jenkins: Intégration continue

Objectif :

- Vérification et optimisation du code.
- Compilation du code
- Packaging de l'application pour l'envoyer dans un dépôt (nexus, image docker, ..)
- Exécution des tests unitaires , fonctionnels
-

Jenkins: Intégration continue

MavenTM

JUnit



Jenkinsfile

Jenkins: Déploiement continue

Objectif :

- Visibilité par rapport au statut du projet
- Réduire le time-to-market
-

Jenkins: Déploiement continue



Supervisor status

[REFRESH](#)[RESTART ALL](#)[STOP ALL](#)

State	Description	Name	Action
running	pid 1006, uptime 0:19:16	mybatis	Restart Stop Clear Log Tail -f

Jenkins: Jenkinsfile

```
#!/groovy
node {
    def mvnHome
    checkout scm

    stage('Preparation') {
        mvnHome = tool 'M3'
    }
    stage('Build') {
        job_name = env.JOB_NAME.toLowerCase()
        job_name = job_name.replaceAll(" ", "-")
        job_name_slug = job_name.replaceAll("/", "_")

        sh "${mvnHome}/bin/mvn clean install -f mybatis/pom.xml"
        sh "ansible-playbook mybatis/deploy/dev.yml --extra-vars 'job_name=${job_name} slug=${job_name_slug}'"
    }

    stage('Results') {
        archive '**/target/*.jar'
    }
}
```

Jenkins: Build

Tous +			
S	M	Nom du projet ↓	Dernier succès
		Mybatis	26 mn - log

Icône: [S](#) [M](#) [L](#)

[Légende](#) [RSS pour tout](#) |

Pipeline master

Full project name: Mybatis/master



[Last Successful Artifacts](#)



[mybatis-0.0.1-SNAPSHOT.jar](#)

27.05 MB [view](#)



[Recent Changes](#)

```
[[1;34mINFO[m] Installing /var/lib/jenkins/workspace/Mybatis_
OAJ50TJ5M3H2P2JITM6KCNKRN7ZA2IK2PLUMQIGB72LVWCAPM7HA/mybatis,
/var/lib/jenkins/.m2/repository/com/example/mybatis/0.0.1-SN
[[1;34mINFO[m] [1m-----
[[1;34mINFO[m] [1;32mBUILD SUCCESS[m
[[1;34mINFO[m] [1m-----
```

```
TASK [spring-boot : Copy config to server] *****
changed: [lab-server] => (item={'key': u'mybatis
u'artifactId': u'mybatis-0.0.1-SNAPSHOT.jar'})
```

```
TASK [spring-boot : Start services] *****
changed: [lab-server]
```

```
PLAY RECAP *****
lab-server                : ok=8    changed=6
```

Supervisor: Mybatis app

Supervisor status

[REFRESH](#)[RESTART ALL](#)[STOP ALL](#)

State	Description	Name	Action
running	pid 1006, uptime 0:19:16	mybatis	Restart Stop Clear Log Tail -f



localhost:4080



{=}



0m



Applications



Misericordia



Intelligence Artificielle

>>



Autres favoris

Populate

#

First Name

Last Name

Environnement de tests:

Dockerfile : <https://github.com/freemanpolys/spring-boot-labs/tree/master/jenkins-ci-cd>

- Jenkins : instance ubuntu docker avec Java8 et Ansible (Exposer le port 8080)
- Serveur de déploiement : instance ubuntu docker avec Java8 et Supervisor (Exposer les port 8080 et 9001)

Code source

<https://github.com/freemanpolys/spring-boot-labs>

Documentation

- https://docs.docker.com/engine/examples/running_ssh_service
- <https://www.digitalocean.com/community/tutorials/how-to-install-and-manage-supervisor-on-ubuntu-and-debian-vps>
- <http://www.geek-directeur-technique.com/2009/03/18/tests-unitaires-et-integration-continue>
- <https://jenkins.io/doc/book/pipeline/jenkinsfile/>
-