



Maven™



sonarqube

Présentation - Architecture applicative

Max Devulder

Présentation Frameworks et Outils

Le quatuor fondamental

GIT

Gestion des sources

Créé par Monsieur Linus Torvald



Maven

Outils d'automatisation du cycle de vie d'un projet.

Créé par Apache



Sonarqube

Mesure de la qualité de code



Jenkins

Outil d'intégration continue le plus populaire.





Début de l'industrie moderne

Automatisation
Travail à la chaîne
Répétition

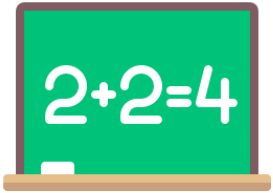


Qualité



Productivité





Deux catégories

Scripting tool

- Ant
- Rake (Ruby)
- Gradle (Groovy)
-

Artifact Tool

- Maven
- Debian package
- ...



Un peu d'histoire

(1977) Make

(2000) Apache ANT

(2002) Maven 1

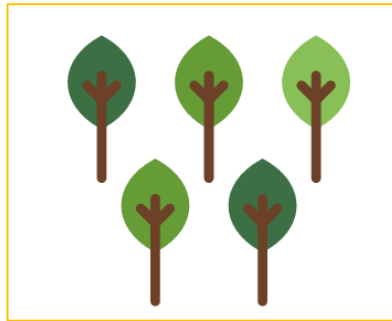
(2005) Maven 2

(2009) Maven 3

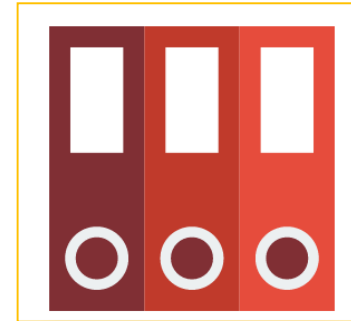
Cycle de vie d'un projet ?



Construction



Gestion
dépendances

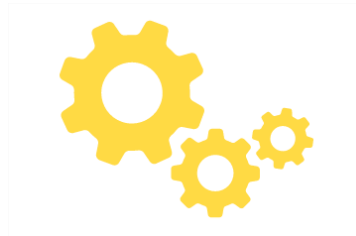


Documentation

1 fichier : Le POM



Project **O**bject **M**odel



Compilation

Génération artifacts
(jar, war, ear ...)

Documentation, statistics
(javadoc, junit ...)

1 fichier : Le POM



Information sur le projet

- GroupId, artifactId, version
- Héritage, aggrégation, sous modules.

« Build » section

- Project layout
- Plugin
- Configuration

« Dependance » section

- Liste des libs du projet
- scope

Le pom « Minimal »

Balise	Description
<project>	Racine XML
<modelVersion>	La version du modèle de POM (4.0.0 pour maven2)
<groupId>	Identifiant du groupe auquel appartient le module Commun à tous les projets. Souvent le nom de domaine de l'entreprise
<artifactId>	Identifiant du module
<version>	Version du module (SNAPSHOT lors de la phase de développement).
<packaging>	Type d'artefact à construire (jar, war ..).

1 fichier : Le POM



```

1 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.
2   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.ap
3   <modelVersion>4.0.0</modelVersion>
4   <parent>
5     <groupId>net.tgi</groupId>
6     <artifactId>oscar-vsope</artifactId>
7     <version>2.0.0-SNAPSHOT</version>
8   </parent>
9   <artifactId>oscar-vsope-back</artifactId>
10  <packaging>war</packaging>
11
12  <build>
13    <plugins>
14      <plugin>
15        <groupId>org.apache.maven.plugins</groupId>
16        <artifactId>maven-war-plugin</artifactId>
17        <version>2.6</version>
18        <configuration>
19          <!-- Ajoute le contenu du META-INF au war -->
20          <archive>
21            <manifest>
22              <addClasspath>true</addClasspath>
23              <classpathPrefix>lib</classpathPrefix>
24            </manifest>
25          </archive>
26          <webResources>

```

I

1 fichier : Le POM



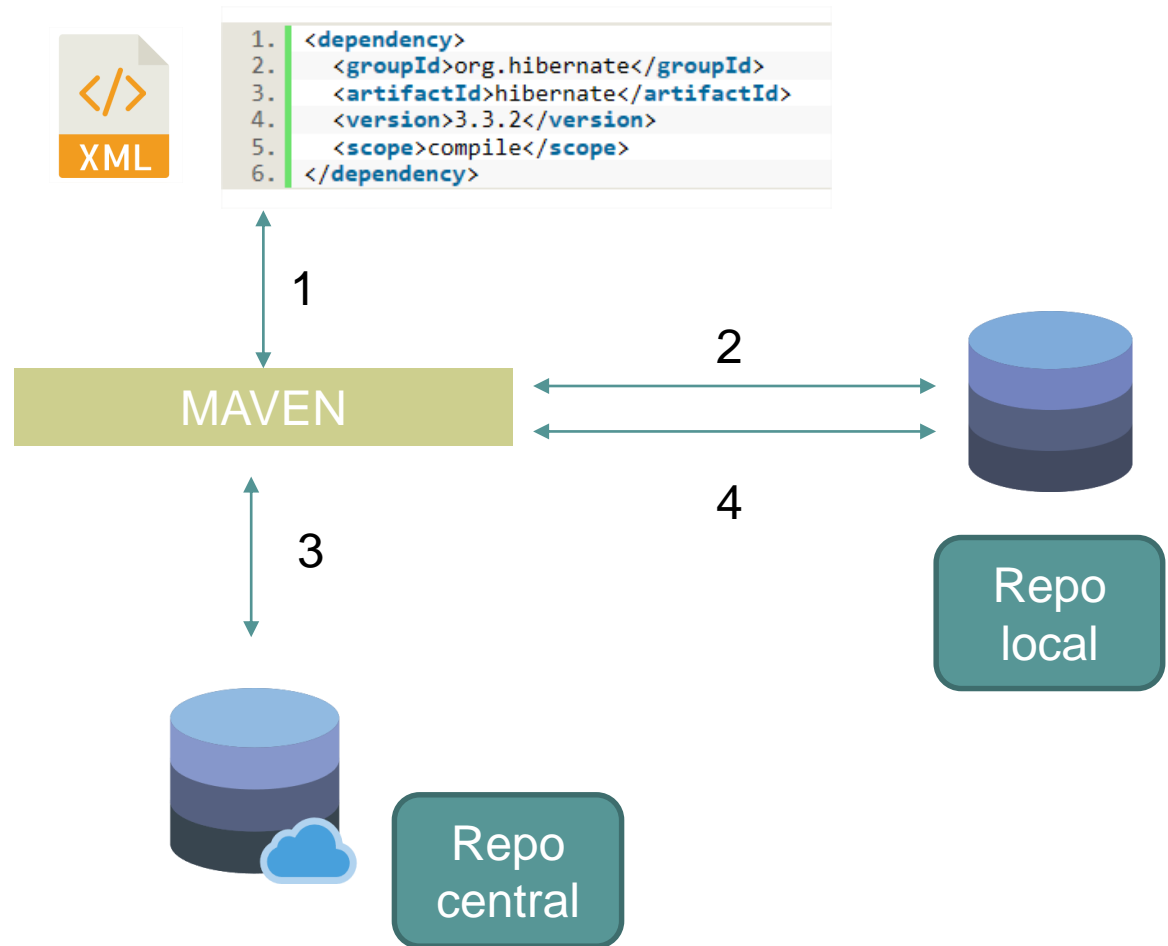
```

38
39  <dependencies>
40    <!-- Socle technique TGI -->
41    <dependency>
42      <groupId>net.tgi</groupId>
43      <artifactId>tgi-api</artifactId>
44      <version>0.0.1-SNAPSHOT</version>
45      <exclusions>
46        <exclusion>
47          <groupId>net.tgi</groupId>
48          <artifactId>tgi-framework</artifactId>
49        </exclusion>
50      </exclusions>
51    </dependency>
52    <dependency>
53      <groupId>net.tgi</groupId>
54      <artifactId>tgi-framework</artifactId>
55      <version>1.0.3-SNAPSHOT</version>
56    </dependency>
57
58    <!-- Spécifique projet -->
59    <dependency>
60      <groupId>net.tgi</groupId>
61      <artifactId>oscar-vsope-middleware</artifactId>
62      <version>${project.version}</version>
63    </dependency>
64    <dependency>
65      <groupId>javax.websocket</groupId>
66      <artifactId>javax.websocket-api</artifactId>
67      <version>1.0</version>

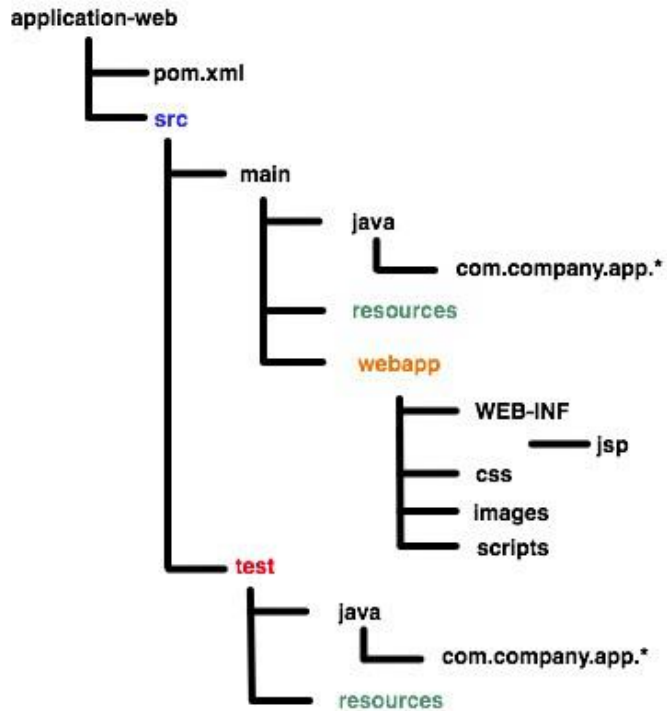
```

Notion de repository

1. Lancement de la tâche
2. Check si lib dans repo local
3. Check si lib dans repo central
4. Téléchargement



Structure d'un projet Maven



Répertoire	Contenu
/src/main/java	le code source (sera compilé dans /target/classes)
/src/main/resources	les fichiers de ressources (fichiers de configuration, images, ...). Le contenu de ce répertoire est copié dans target/classes pour être inclus dans l'artéfact généré
/src/main/webapp	les fichiers de la webapp
/src/test/java	le code source des tests (sera compilé dans /target/test-classes)
/src/test/resources	les fichiers de ressources pour les tests
/target	les fichiers générés pour les artéfacts et les tests (ce répertoire ne doit pas être inclus dans le gestionnaire de sources)
/pom.xml	le fichier POM de description du projet

Des commandes pour tout

```
> mvn --version
```

```
> mvn archetype:generate
```

```
> mvn package
```

En fait, c'est facile :



<https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html>



Jenkins



POURQUOI ?

Continuous Integration

Methodology

- ▶▶▶ Code Repository
- ▶▶▶ Daily Check-In
- ▶▶▶ Build Automation
- ▶▶▶ Test Automation
- ▶▶▶ Deploy Automation

Advantages

- ▶▶▶ Early bug detection
- ▶▶▶ Quality of delivery
- ▶▶▶ Increased release cadence

Disadvantages

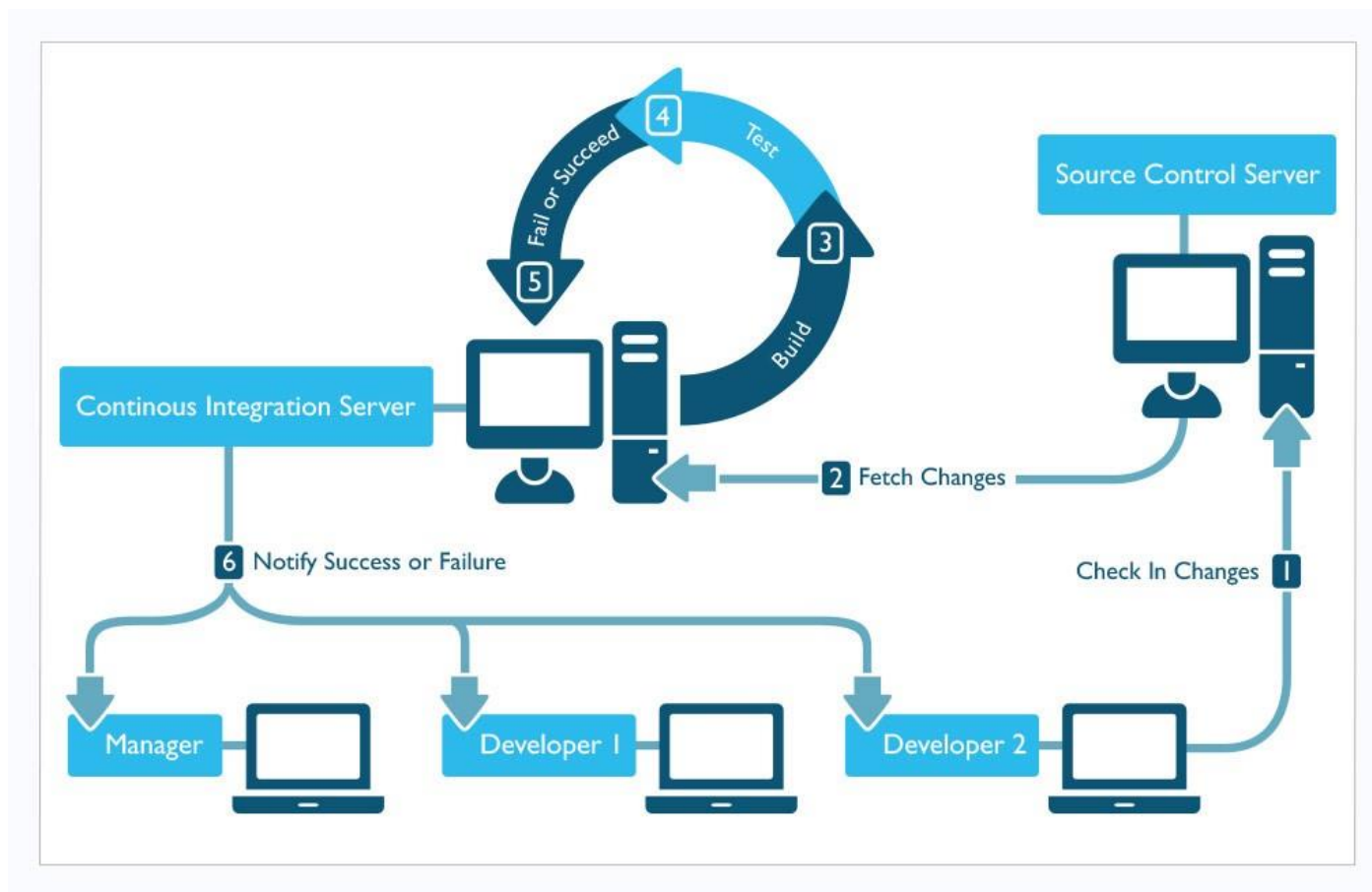
- ▶▶▶ Initial setup
- ▶▶▶ Extra cost: hard- & software
- ▶▶▶ Continuous maintenance

Return on Investment?





JENKINS : Chef d'orchestre





Ca ressemble à quoi ?

Jenkins

Pipelines

Administration



Déconnexion

Tableau de bord

Search pipelines...

Nouveau Pipeline

NOM	SANTÉ	BRANCHES	PULL REQUE
carto_ws		1 avec succ	-
Copie Files Yard To Deploy		-	-
Deploy Dashboard		-	-
Deploy libs		-	-
Deploy TGIBOX		-	-
Deploy TWEB		-	-
Deploy VSOPE		-	-
Deploy YARD		-	-
Katway - Deploy RCT - Back		-	-
Katway - Deploy RCT - Front		-	-



Ca ressemble à quoi ?

jenkins / Deploy VSOPE / #47

Non sécurisé | 10.157.222.224:8080/blue/organizations/jenkins/Deploy%20VSOPE/detail/Deploy%20VSOPE/47/pipeline

Applications TGI Katway Perso PIC BENIN Oscar FogBugz Vessel Operations -... JIRA - VSOPE MDE - Sprint Bollor... Site Helene pubnub/api-guide-...

✓ Deploy VSOPE < 47

Pipeline Modifications Tests Artefacts Déconnexion

Branche: — 5m 2s Aucune modification
Commit: — 4 days ago Démarré par l'utilisateur admin

Start Vérification Arrêt Tomcat Arrêt Tomcat : Check Préparation serveur Clean log : Check empty Upload livrable Check Upload livrable Start Tomcat Verif Tomcat started End

Access serveur distant
Check livrable
Clean livrable
Clean log

Verif Tomcat started - 1m 39s

Restart Verif Tomcat started

✓	> Print Message	<1s
✓	> Failed : java.net.ConnectException: Connexion refusée (Connection refused) — Print Message	<1s
✓	> Print Message	<1s
✓	> Failed : java.net.ConnectException: Connexion refusée (Connection refused) — Print Message	<1s
✓	> Print Message	<1s
✓	> Failed : java.net.ConnectException: Connexion refusée (Connection refused) — Print Message	<1s
✓	> Print Message	1m 35s
✓	> Succeed ! — Print Message	<1s



Depuis peu : Jenkins Pipeline

Jenkinsfile (Declarative Pipeline)

```
pipeline {
  agent any
  stages {
    stage('Build') {
      steps {
        sh 'echo "Hello World"'
        sh '''
            echo "Multiline shell steps works too"
            ls -lah
            ...
        '''
      }
    }
  }
}
```

<https://jenkins.io/doc/pipeline/tour/getting-started/>

Exemple complet de mise en production :



CD_PIPELINE_VSOPE.groovy



Projet

Non sécurisé | 10.157.222.224:9000/projects?sort=-analysis_date

Applications TGI Katway Perso PIC BENIN Oscar FogBugz Vessel Operations -... JIRA - VSOPE MDE - Sprint Bollor... Site Helene pubnub/api-guide-...

sonarqube Projets Défauts Règles Profils Qualité Barrières Qualité

Rechercher des projets Connexion

Perspective: Statut global Trier par: Date de dernière an... Rechercher par nom ou clé de projet 19 projets

Filtres

Barrière qualité

Ok 3

Avertissement 1

Erreur 15

Fiabilité (Bugs)

A 6

B 2

C 5

D 4

E 2

Sécurité (Vulnérabilité)

A 6

B 8

C 1

D 3

E 1

Maintenabilité (Mauvaises pratiques)

A 7

B 0

C 8

D 3

E 1

Couverture

≥ 80% 0

70% - 80% 0

50% - 70% 0

30% - 50% 0

< 30% 19

Aucune donnée 0

carto_ws Erreur

Dernière analyse : 6 janvier 2020 à 04:19

18 Bugs 8 Vulnérabilité 443 Mauvaises pratiques 0,0% Couverture (TU) 1,2% Duplications 17k M Java, CSS, ...

portail-back Erreur

Dernière analyse : 6 janvier 2020 à 03:59

0 Bugs 3 Vulnérabilité 167 Mauvaises pratiques 0,0% Couverture (TU) 0,7% Duplications 2,8k S Java

tgi-dashboard-back Erreur

Dernière analyse : 6 janvier 2020 à 03:49

3 Bugs 5 Vulnérabilité 208 Mauvaises pratiques 0,0% Couverture (TU) 11,9% Duplications 7,2k S Java, XML, ...

portail-front Erreur

Dernière analyse : 6 janvier 2020 à 03:42

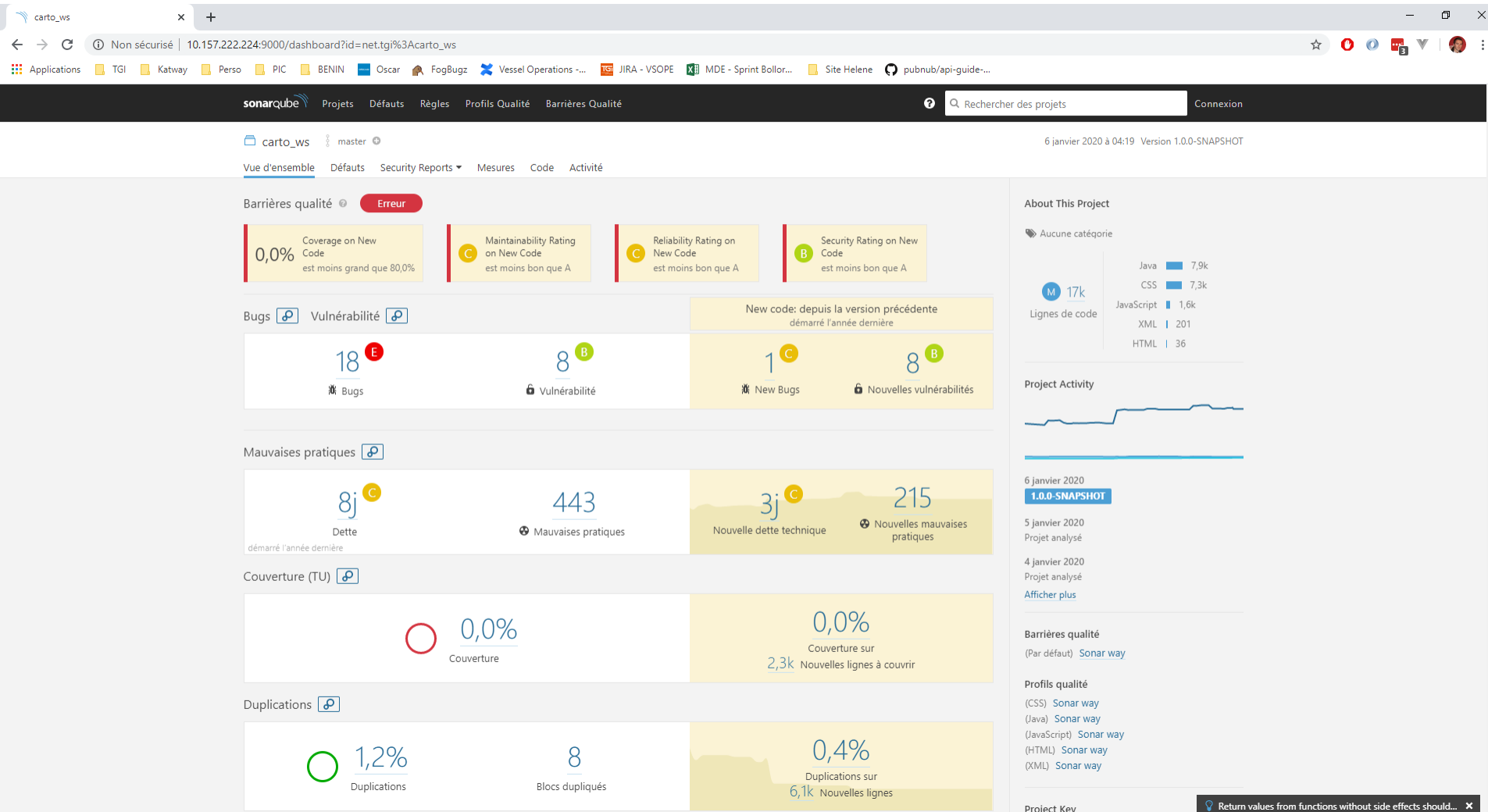
9 Bugs 0 Vulnérabilité 44 Mauvaises pratiques 0,0% Couverture (TU) 1,4% Duplications 14k M CSS, JavaScript

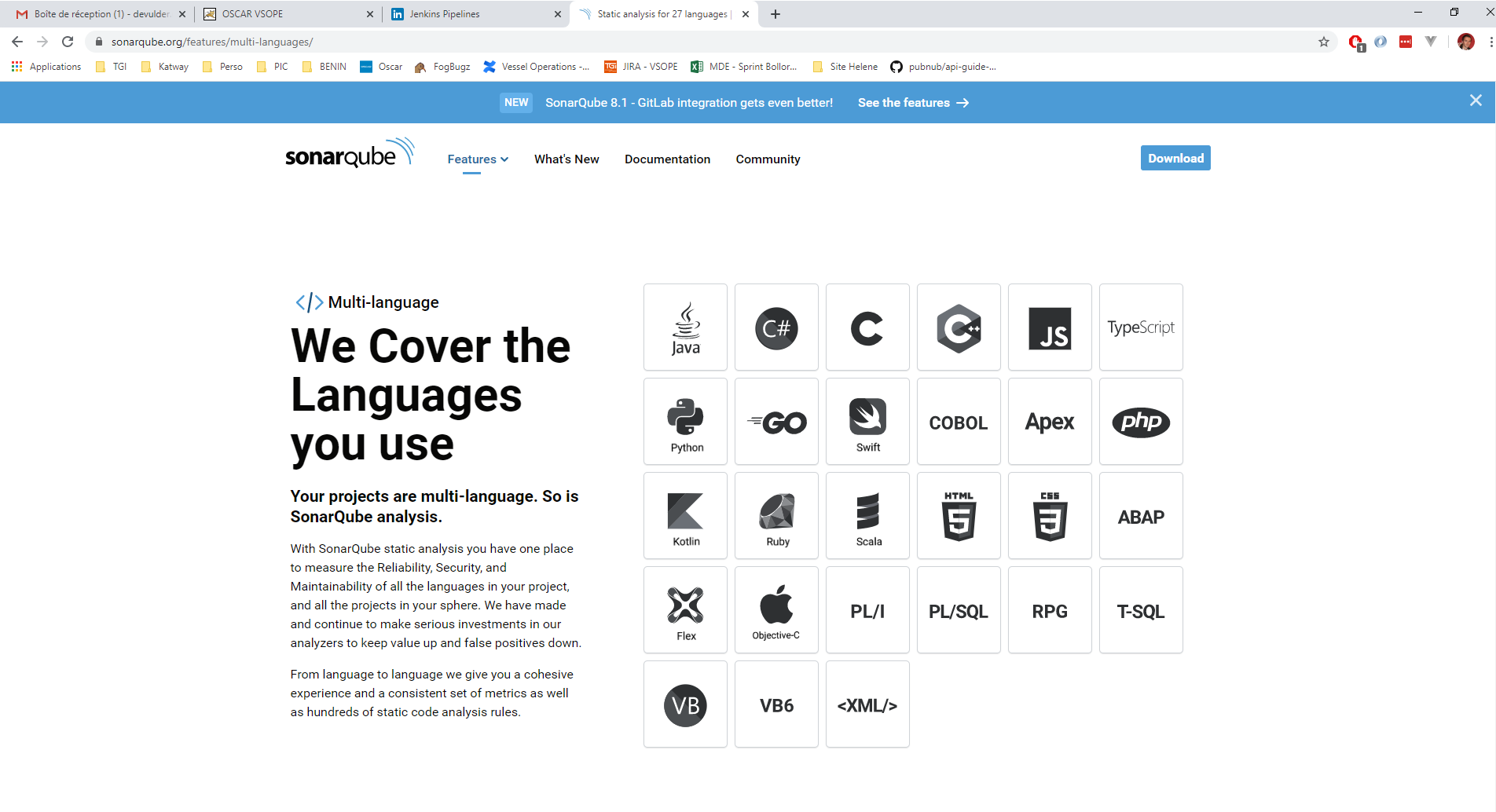
carto Erreur

Dernière analyse : 6 janvier 2020 à 02:57

71 Bugs 5 Vulnérabilité 974 Mauvaises pratiques 0,0% Couverture (TU) 7,8% Duplications 34k M JavaScript, CSS, ...

Return values from functions without side effects should...





Boîte de réception (1) - devulder x | OSCAR.VSOPE x | Jenkins Pipelines x | Static analysis for 27 languages | x +

sonarqube.org/features/multi-languages/

Applications TGI Katway Perso PIC BENIN Oscar FogBugz Vessel Operations ~... JIRA - VSOPE MDE - Sprint Bollor... Site Helene pubnub/api-guide...

NEW SonarQube 8.1 - GitLab integration gets even better! See the features →

sonarqube Features ▾ What's New Documentation Community Download

</> Multi-language

We Cover the Languages you use

Your projects are multi-language. So is SonarQube analysis.

With SonarQube static analysis you have one place to measure the Reliability, Security, and Maintainability of all the languages in your project, and all the projects in your sphere. We have made and continue to make serious investments in our analyzers to keep value up and false positives down.

From language to language we give you a cohesive experience and a consistent set of metrics as well as hundreds of static code analysis rules.

Java	C#	C	C++	JS	TypeScript
Python	GO	Swift	COBOL	Apex	php
Kotlin	Ruby	Scala	HTML	CSS	ABAP
Flex	Objective-C	PL/I	PL/SQL	RPG	T-SQL
VB	VB6	<XML/>			

<https://www.sonarqube.org/features/multi-languages/>