

INTÉGRATION CONTINUE

Intégration continue S. Fauvel – Univ. Nantes 2025

PLAN

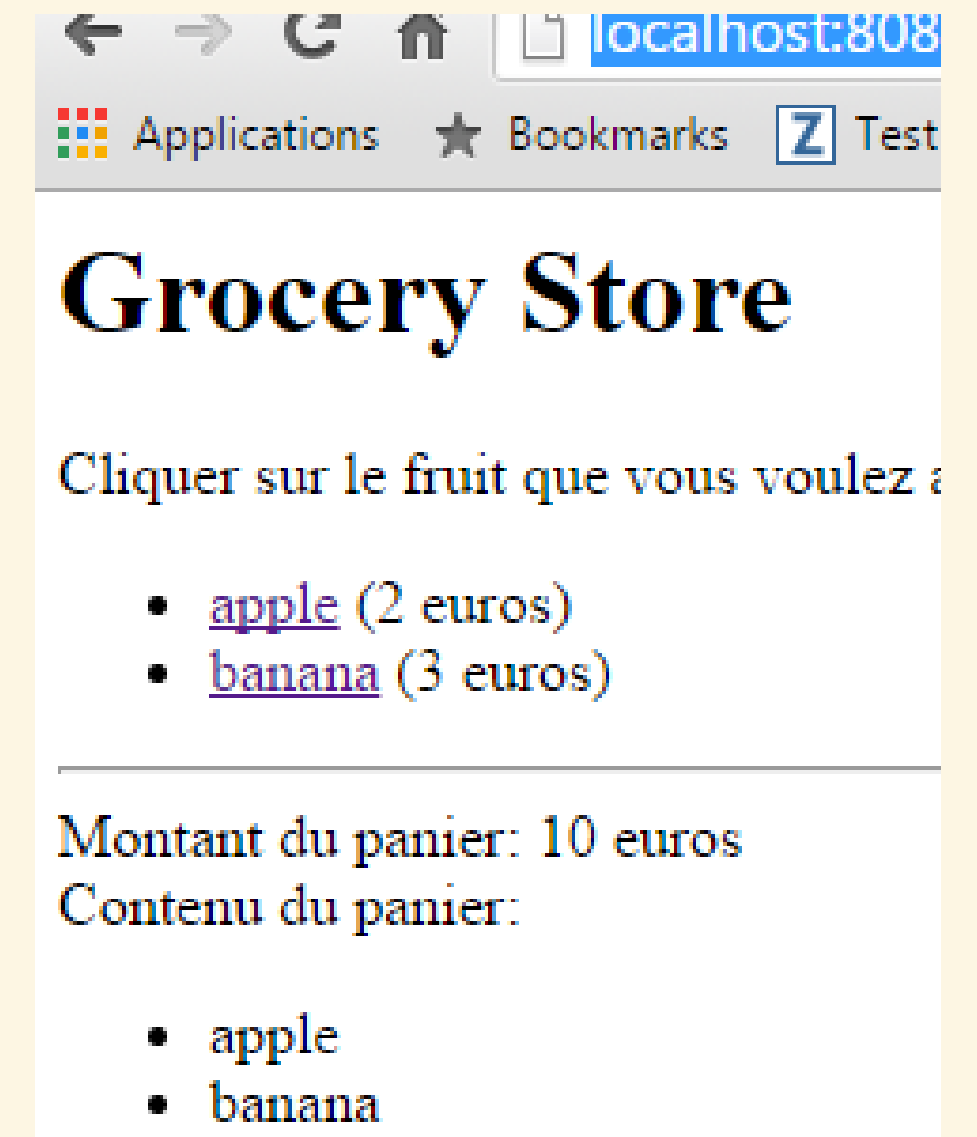
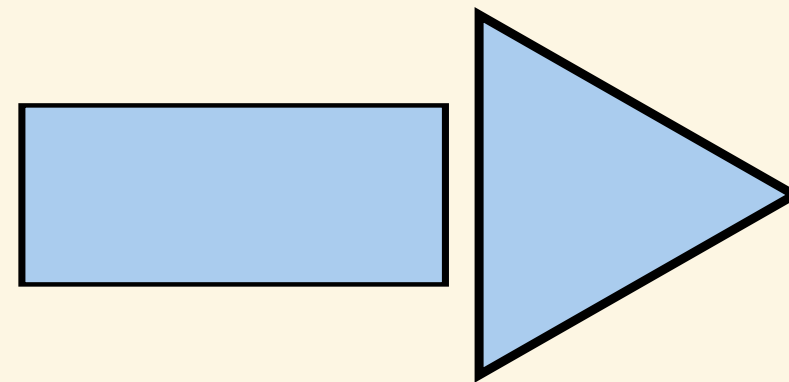
- Intégration
- Intégration continue
- Jenkins

INTÉGRATION

ENTRE LE DÉVELOPPEMENT ET LA PRODUCTION ?

```
public StoreServer(PriceStorage priceStorage) {
    this.priceStorage = priceStorage;
    Basket basket = new Basket();
    this.shopping = new Shopping();
}

public void start(int port) {
    HttpServer server = HttpServer.create(
        port, 1);
    server.createContext("/");
}
```



PLANNING

Test unitaires

Compilation

Analyse de code

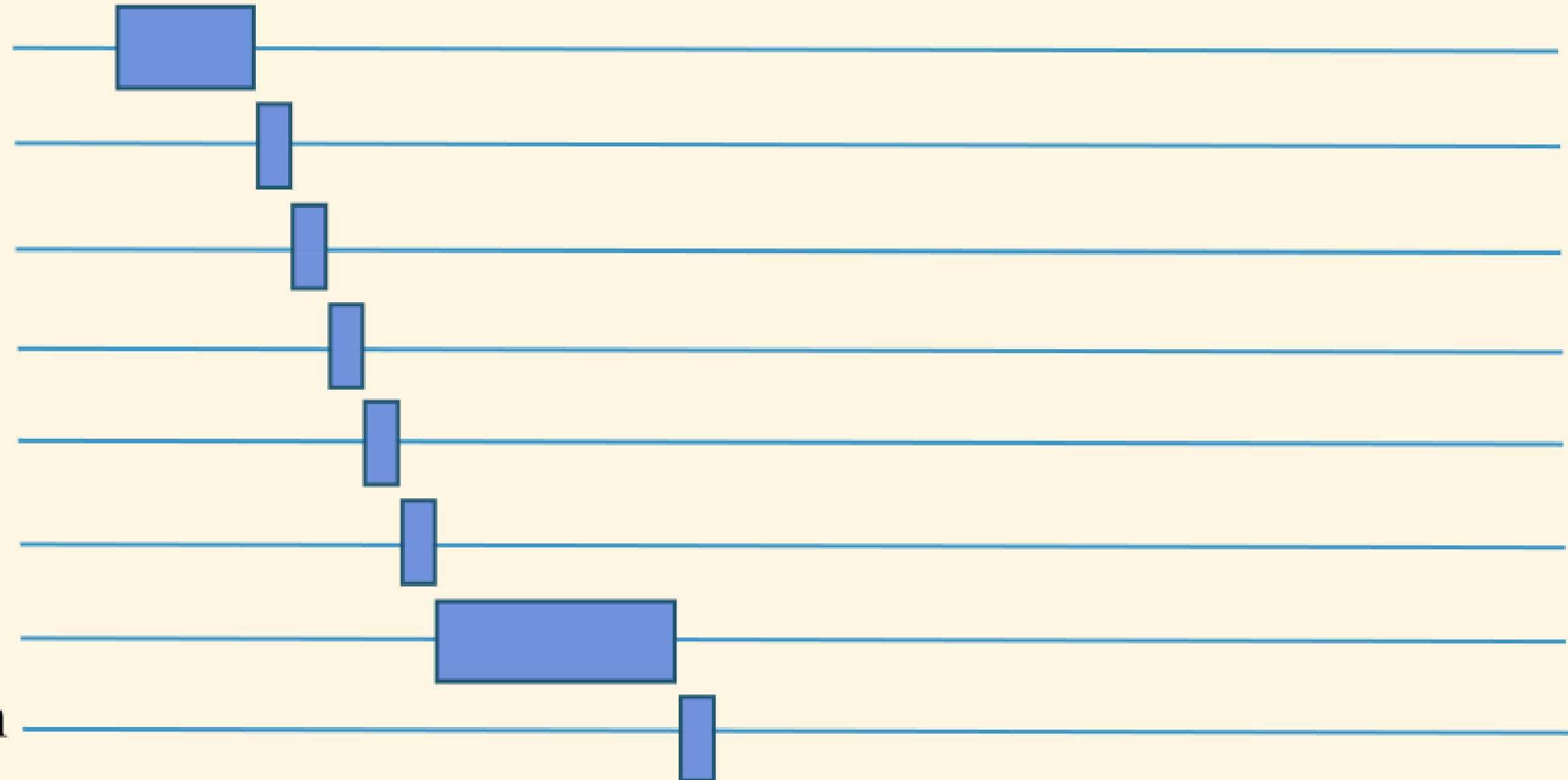
Packaging

Documentation

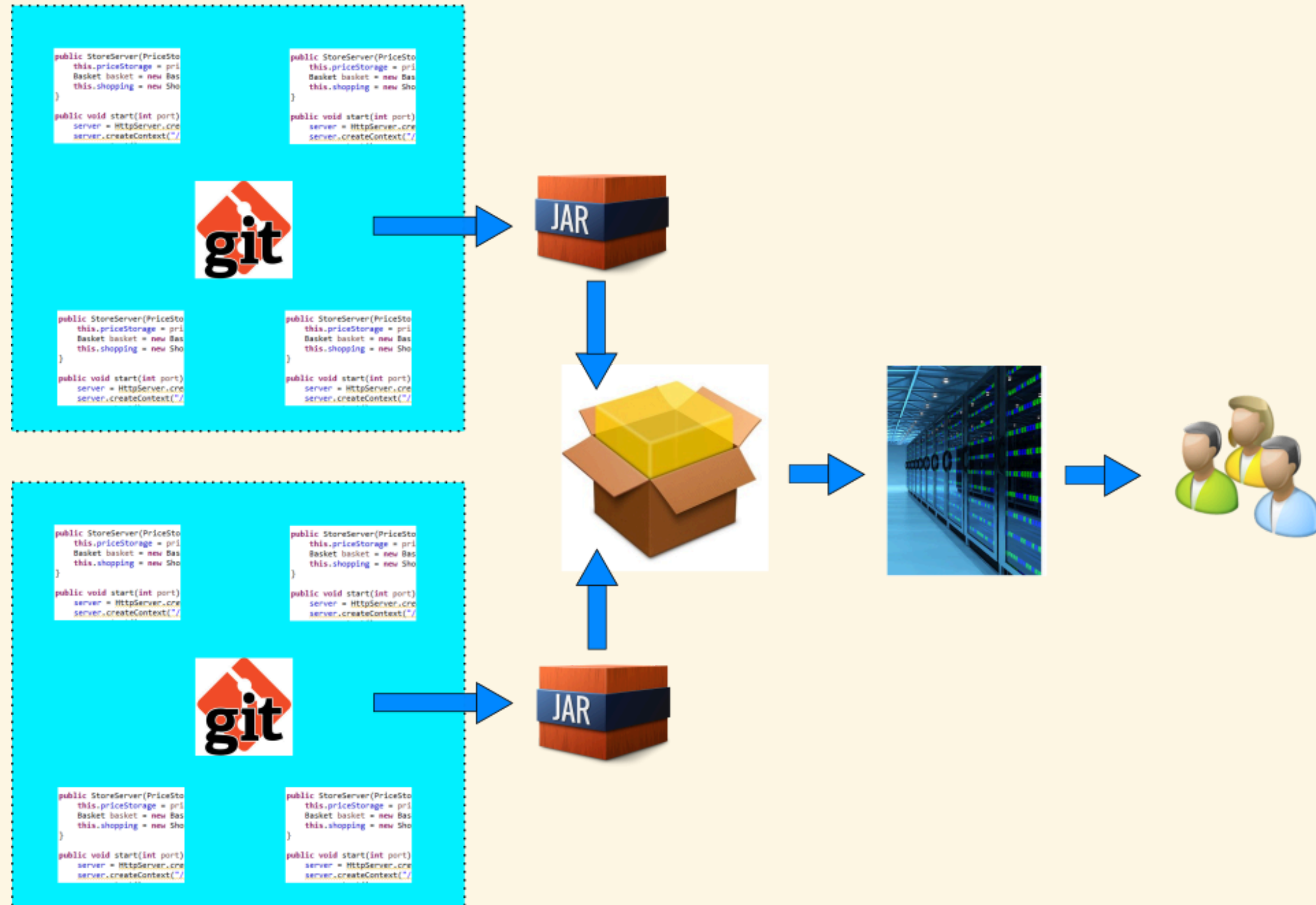
Déploiement

Recette

Mise en production



INTÉGRATION



PLANNING



**J'ADORE QUAND UN PLAN SE DEROULE
SANS ACCROC..**

PLANNING

Test unitaires

Compilation

Analyse de code

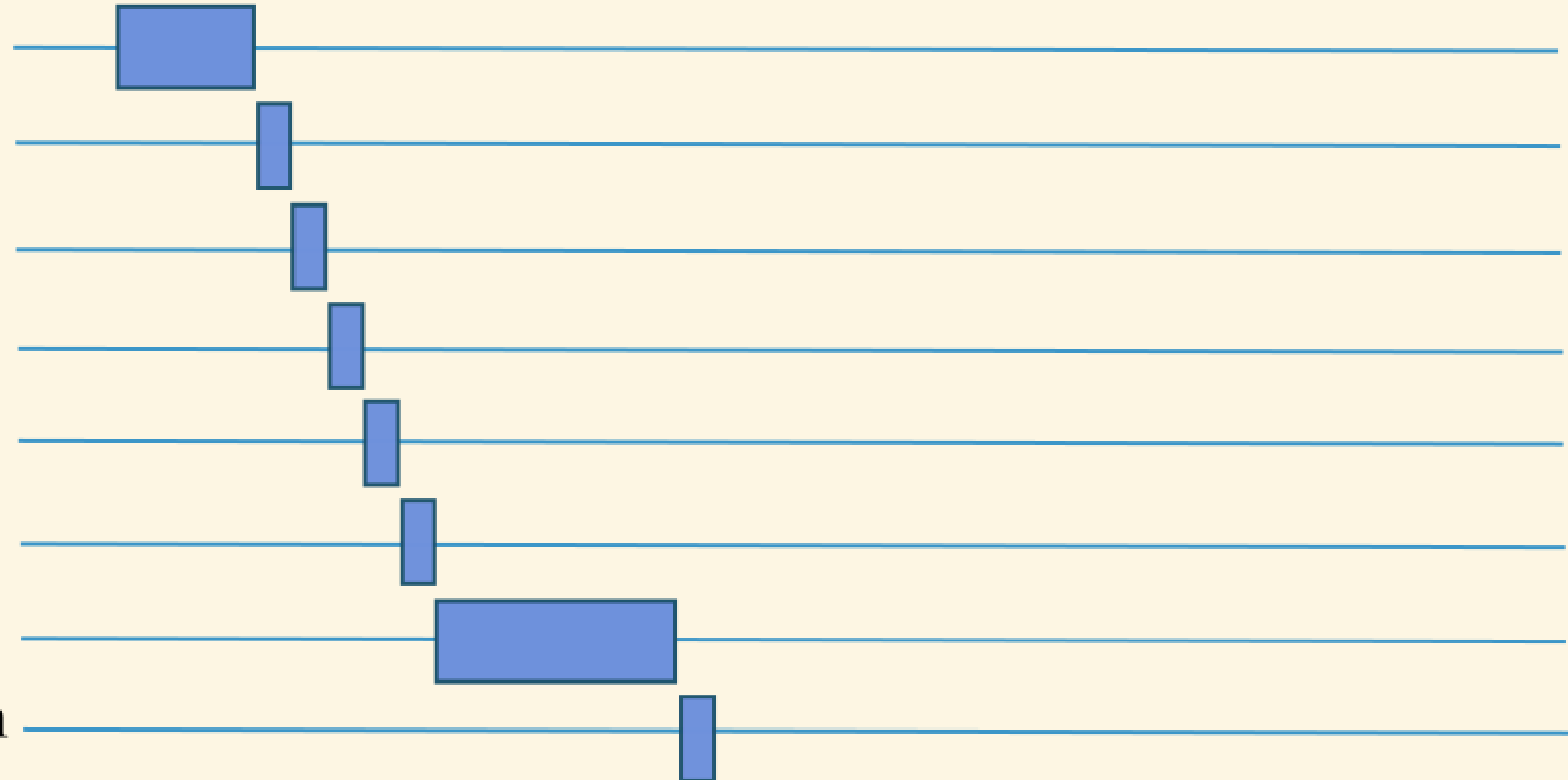
Packaging

Documentation

Déploiement

Recette

Mise en production



EN PRATIQUE

Test unitaires

Compilation

Analyse de code

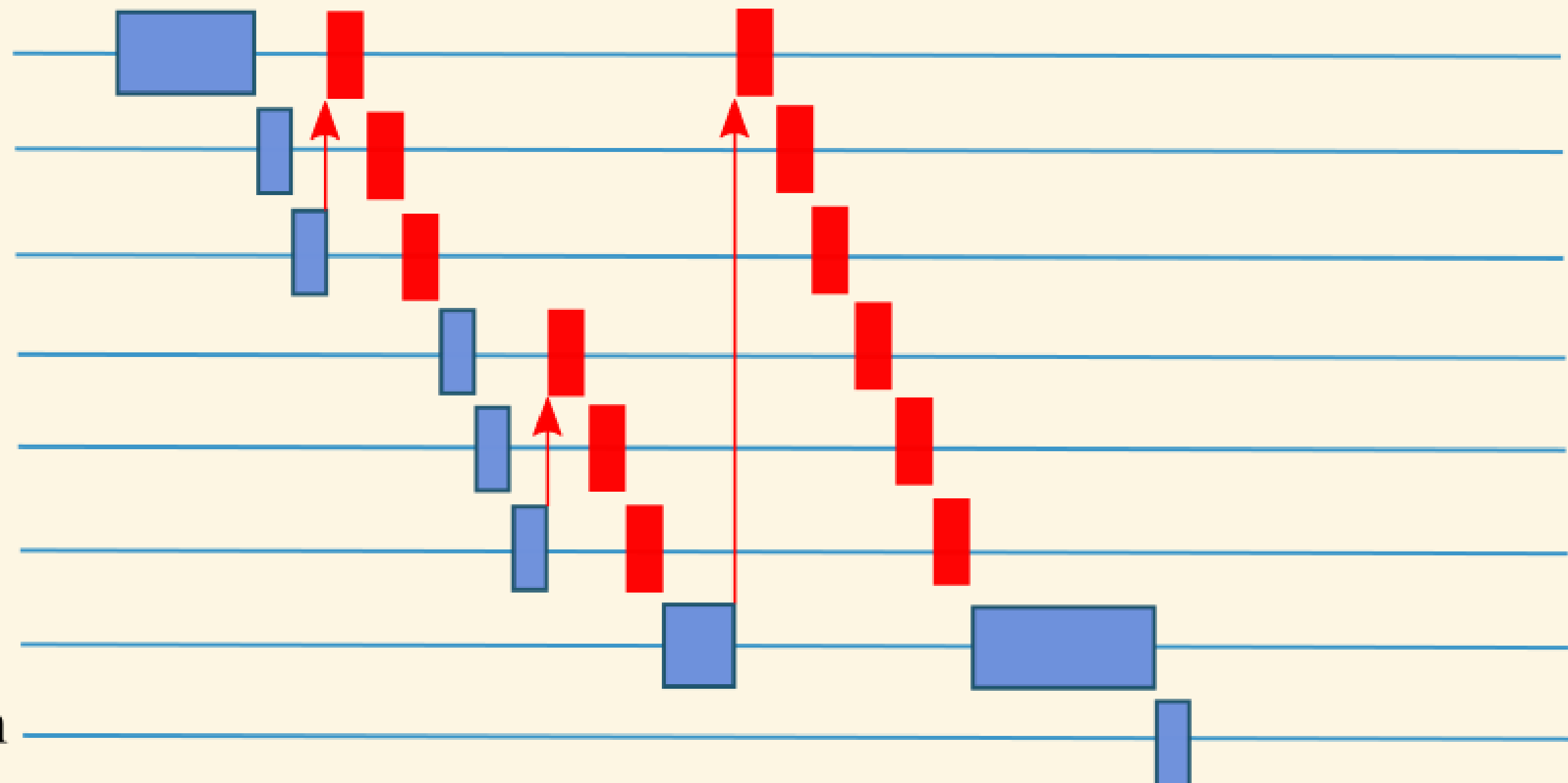
Packaging

Documentation

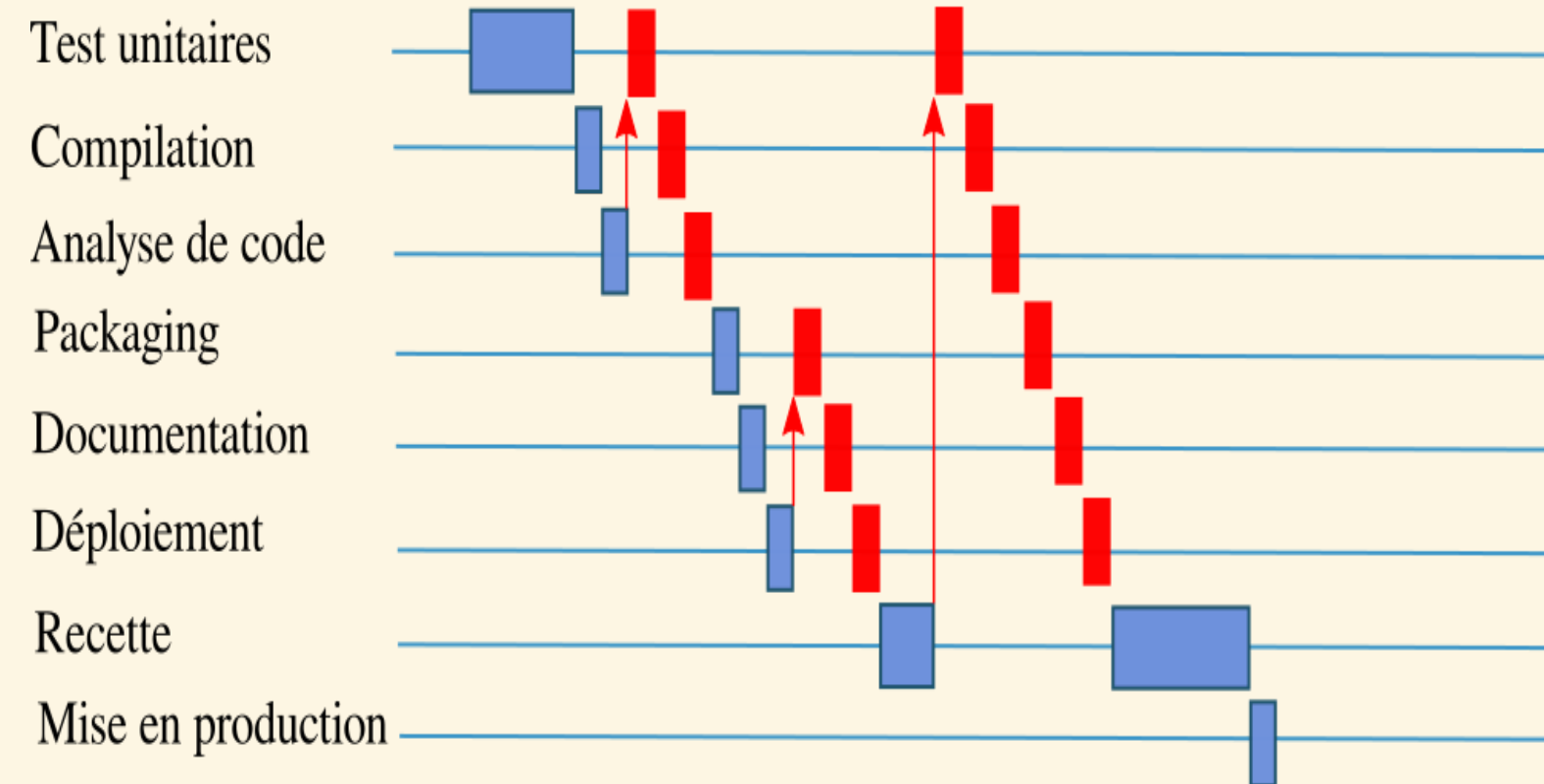
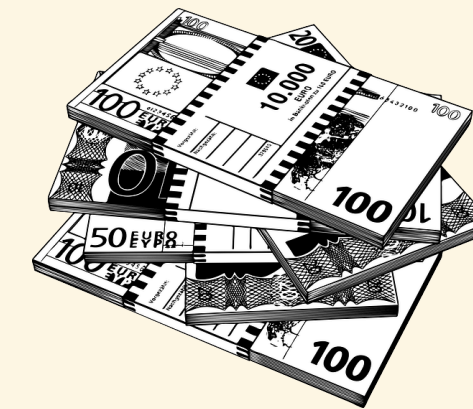
Déploiement

Recette

Mise en production

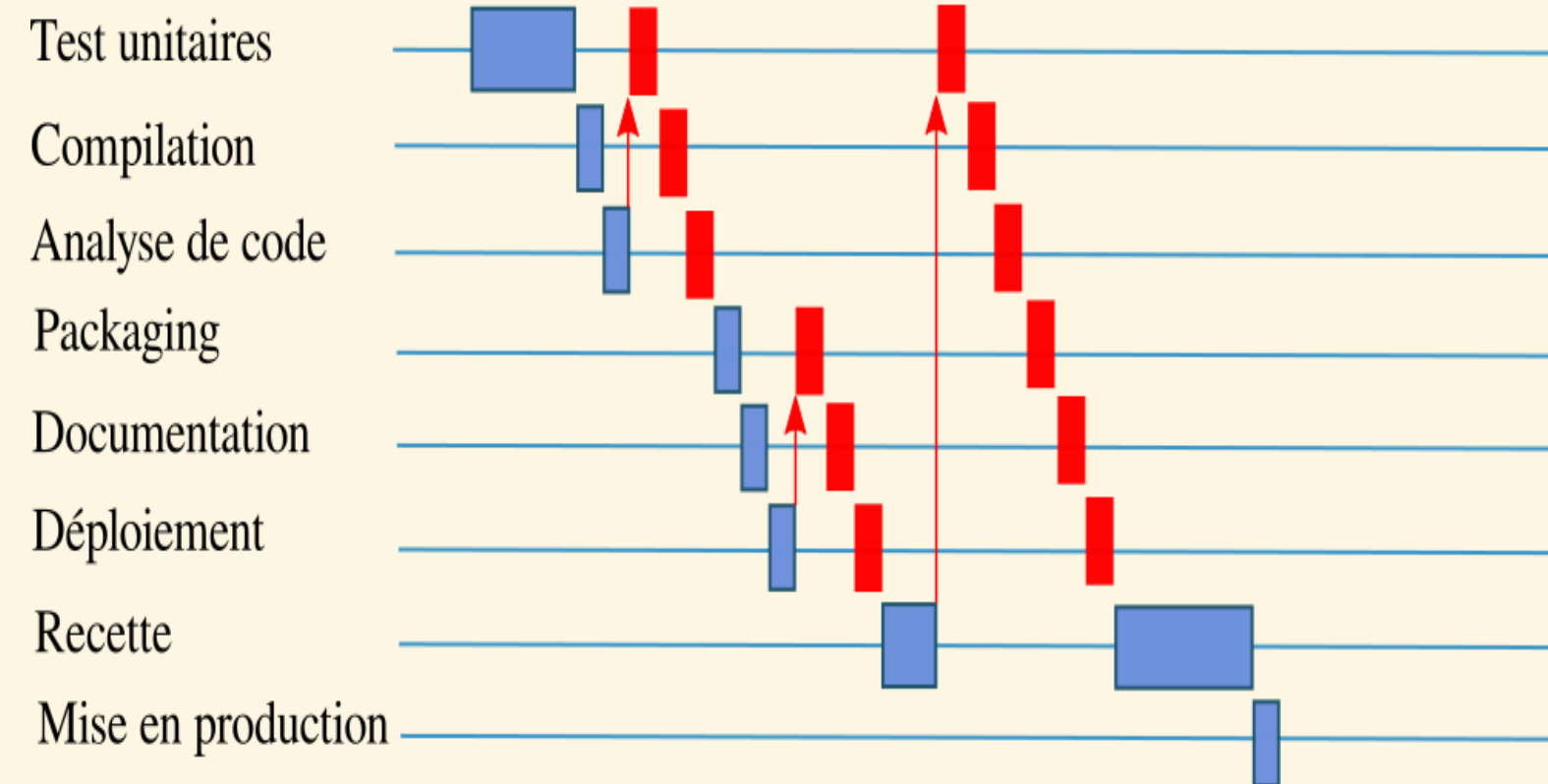
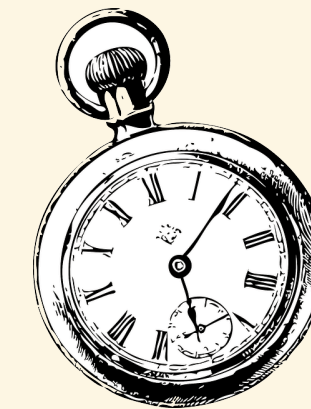


CHARGE



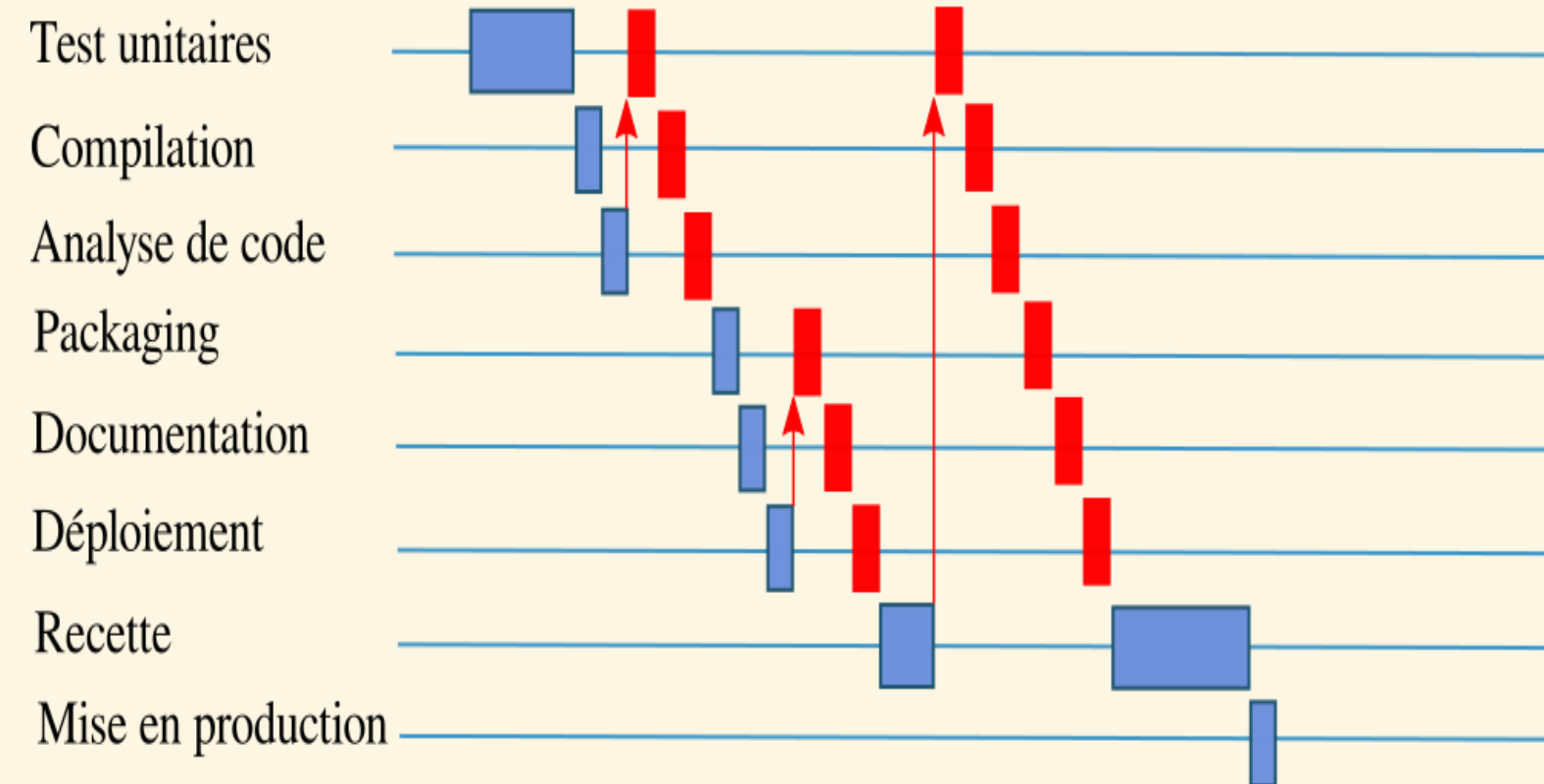
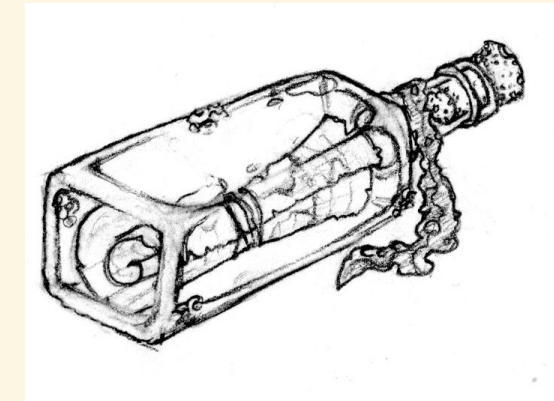
Les étapes à repasser sont une **charge de travail** supplémentaire

TEMPS



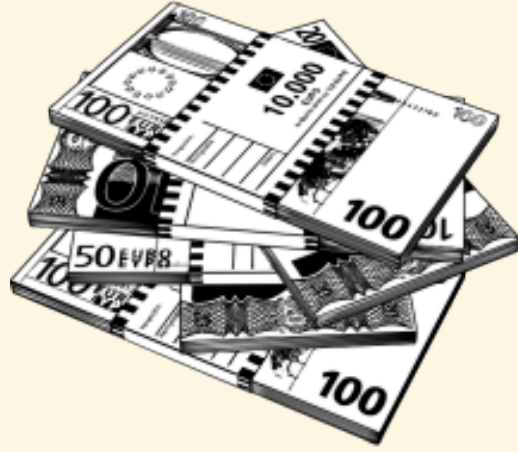
Chaque étape à refaire allonge le **délai** de livraison

FEEDBACK



Le temps d'intégration éloigne d'autant le **délai entre le développement et sa validation**

INCERTITUDE



Charges



Temps



Feedback



1 CONSTAT

- L'intégration est une phase compliquée
- C'est une étape incertaine
- Il faut prévoir une charge de travail

INTÉGRATION DANS UN CYCLE EN V

- Le **moins** souvent possible
- 1 seule grande intégration
- Estimer et planifier la phase d'intégration
- Prévoir pour limiter les risques

INTÉGRATION DANS UNE APPROCHE AGILE

- Le **plus** souvent possible
- De nombreuses petites intégrations
- Incorporer cette phase au développement
- Avoir une feedback au plus tôt

INTÉGRATION CONTINUE

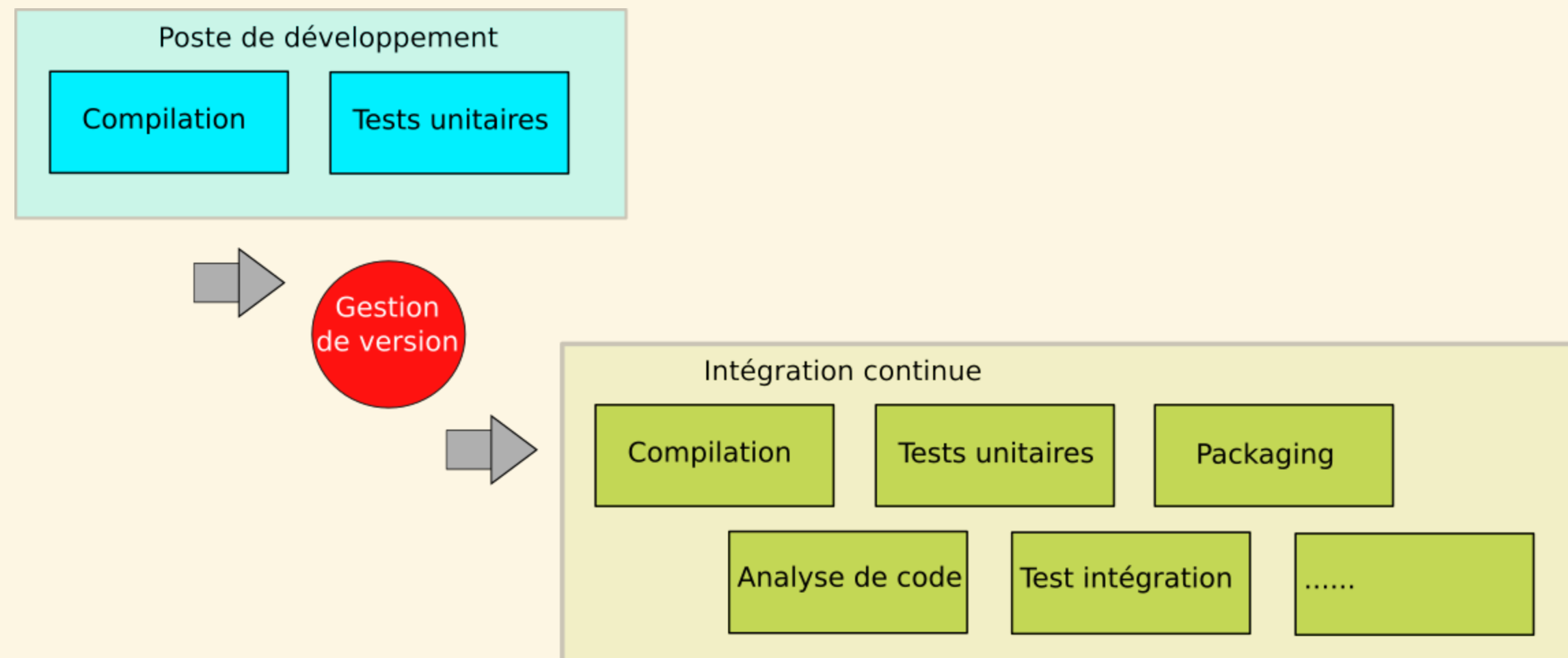
ORIGINE

- Grady Booch - **Object Oriented Design: With Applications**, 1991
- Kent Beck - **Extreme Programming Explained**, 1999
 - Diminuer les risques de conflit
 - Détecter les défauts au plus tôt
 - Avoir un logiciel opérationnel en permanence
- Martin Fowler - **Continuous Integration**, 2000

<http://www.martinfowler.com/articles/originalContinuousIntegration.html>

INTÉGRATION CONTINUE

Automatiser la chaîne de build pour faire l'intégration de manière transparente



BONNES PRATIQUES

- Commiter régulièrement du code fonctionnel
 - L'intégration continue s'appuie sur le code commité
 - L'application doit compiler et les tests passer
- Avoir des tests automatiser pour détecter les anomalies
- Corriger les builds en échec
- Optimiser les temps de builds pour améliorer le feedback

BÉNÉFICES

- Réduction des risques lié à l'incertitude
 - Etat du projet connu en permanence
 - Dernière version stable identifiée
- Réduction des tâches répétitives
 - Gain de temps pour les équipes
 - Réduction des délais pour l'intégration
- Impose des bonnes pratiques
 - Oblige une rigueur de travail
 - Assure la reproductibilité

DÉFINITIONS

- **Continuous Integration:** Vérifie l'intégration
- **Continuous Delivery:** Prêt à être mis en production
- **Continuous Deployment:** Mise en production

<https://medium.com/jorgeacetozi/continuous-integration-vs-continuous-delivery-vs-continuous-deployment-d5839a85a959>

OUTILS

- Jenkins
- Bitbucket Pipelines
- GitHub Actions
- Circle CI
- TeamCity
- Gitlab CI
- ...

JENKINS



JENKINS

Serveur d'intégration continue open source

Gère et contrôle les tâches du cycle de vie d'un logiciel:

- Compilation
- Documentation
- Tests
- Packaging
- analyse de code
- ...

Extensible en ajoutant des plugins

JENKINS - INSTALLATION

Serveur Web, un simple war

Lancement :

```
java -jar jenkins.war
```

Accès:

```
http://localhost:8080
```

JENKINS - JOBS

- Un **job** est une **configuration** pour l'intégration d'un projet
- Un projet peut avoir plusieurs jobs
- Le lancement d'un job exécute les instructions décrites dans sa configuration

JENKINS – TABLEAU DE BORD

Création d'un job

The screenshot shows the Jenkins dashboard interface. At the top is a black header with the Jenkins logo, a search bar, and user information. Below this is a light gray sidebar with navigation links. The main content area displays a table of jobs. Annotations with arrows point to specific elements: 'Création d'un job' points to the '+ Nouveau Item' button; 'Status du job' points to the status icon of the 'MavenScript' job; 'Nom du job' points to the 'MavenScript' job name; and 'Lancement manuel du job' points to the green play button icon at the end of the 'MavenScript' row.

Tableau de bord >

+ Nouveau Item [Ajouter une description](#)

Utilisateurs

Historique des constructions

Administrer Jenkins

Mes vues

File d'attente des constructions

File d'attente des constructions vide

État du lanceur de compilations

S	M	Nom du projet ↓	Dernier succès	Dernier échec	Dernière durée	
✓	⚙️	FirstTp	1 an. 8 mo. #15	s. o.	88 ms	▶
⌚	⚙️	HelloWorld	38 mn #3	s. o.	20 ms	
✗	☁️	MavenScript	s. o.	1 an. 8 mo. #2	73 ms	▶

Icône: S M L Légende Atom feed pour tout Atom feed de tous les échecs Atom feed juste pour les dernières compilations

Status du job

Nom du job

Lancement manuel du job

JENKINS – STATUT DU JOB

✓	Succès	Tout fonctionne
✗	En échec	La construction n'a pas pu se faire (compilation, ...)
!	Instable	La build a abouti mais des tests échouent
⌚	Désactivé	Le job est désactivé

JENKINS - JOBS

Plusieurs types de jobs

- **Free-style** : accès à toutes les options disponibles
- **Maven** : basé sur une configuration Maven
- **Multi-configuration** : projets complexes avec plusieurs builds
- **Job externe** : monitoring de processus externe
- **Pipeline** : définition du processus d'intégration/déploiement sous forme de script

JENKINS - JOBS





 **Jenkins**


Tableau de bord > HelloWorld >


Configuration


 **General**

 Gestion de code source

 Ce qui déclenche le build

 Environnements de Build

 Build Steps

 Actions à la suite du build

General

Description

[Plain text] [Prévisualiser](#)

☐ Supprimer les anciens builds

☐ Ce build a des paramètres avancés







☐ Throttle builds (en attente)

☐ Exécuter des builds en parallèle


Avancé...

GÉNÉRAL

Configuration

-  **General**
-  Gestion de code source
-  Ce qui déclenche le build
-  Environnements de Build
-  Build Steps
-  Actions à la suite du build

General

Enabled 

Description

[Plain text] [Prévisualisation](#)


- ☐ Ce build a des paramètres ?
- ☐ Supprimer les anciens builds ?
- ☐ Throttle builds ?
- ☐ Exécuter des builds simultanément si nécessaire ?

Avancé...


- Paramètres du build
- Supprimer les anciens build
- ...


GESTION DE CODE SOURCE


Configuration


 General

 Gestion de code source

 Ce qui déclenche le build

 Environnements de Build

 Build Steps

 Actions à la suite du build

Gestion de code source

☐ Aucune

☒ Git ?

Repositories ?

Repository URL ?

/home/project/cours_ic

Credentials ?

- aucun -

+ Ajouter

Avancé...

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

*/master

Add Branch

GESTION DE CODE SOURCE

- Récupération dans un espace dédié
- Construction de l'application dans cet espace

Tableau de bord > FirstMaven >

↑ Retour au tableau de bord

📄 État

</> Modifications

📁 Répertoire de travail

🗑 Effacer l'espace de travail

▶ Lancer un build

⚙ Configurer

🗑 Supprimer Maven project

📄 Modules

✎ Renommer

Workspace of FirstMaven on maître

FirstMaven / →

📁 .git

📁 src

📁 target

📄 .gitignore

📄 IntegrationContinue_Cours.pdf

📄 IntegrationContinue_TP.pdf

📄 Jenkinsfile


📄 pom.xml

27 nov. 2022 à 21:30:50	43 B	🔒	👁
27 nov. 2022 à 21:30:50	2.21 MB	🔒	👁
27 nov. 2022 à 21:30:50	144.91 KB	🔒	👁
27 nov. 2022 à 21:30:50	653 B	🔒	👁
27 nov. 2022 à 21:30:50	1.92 KB	🔒	👁


📄 (Tous les fichiers dans un zip)


CE QUI DÉCLENCHÉ LE BUILD


Configuration


 General

 Gestion de code source


 **Ce qui déclenche le build**


 Environnements de Build

 Pre Steps

 Build

 Post Steps

 Configuration du build

 Actions à la suite du build

Ce qui déclenche le build

- ☐ Lance un build à chaque fois qu'une dépendance SNAPSHOT est construite ?
- ☐ Déclencher les builds à distance (Par exemple, à partir de scripts) ?
- ☐ Construire après le build sur d'autres projets ?
- ☐ Construire périodiquement ?
- ☒ Scrutation de l'outil de gestion de version ?

Planning ?

H/15 * * * *

Aurait été lancé à Sunday, November 27, 2022 at 9:30:25 PM Coordinated Universal Time; prochaine exécution à Sunday, November 27, 2022 at 9:45:25 PM Coordinated Universal Time.

- ☐ Ignore post-commit hooks ?

Environnements de Build





- ☐ Use secret text(s) or file(s) ?

JENKINS - DÉCLENCHEMENT




- Planification:
 - Construire périodiquement
 - Scruter la gestion de configuration
- La syntaxe est celle du Cron
 - [MINUTES] [HEURES] [JOURMOIS] [MOIS] [JOURSEMAINE]
 - '*' représente l'ensemble des valeurs possibles
 - 'N' déclenche lorsque la valeur vaut N
 - '*/X' déclenche tous les X
- Exemple:
 - toutes les 5 minutes: */5 * * * *
 - A 23h30 le vendredi : 30 23 * * 5

JENKINS – GESTIONNAIRE DE CONFIGURATION

Déclenchement toutes les 2 minutes

 #12	<u>22 mars 2021 11:07</u>
 #11	<u>22 mars 2021 11:06</u>
 #10	<u>22 mars 2021 10:59</u>
 #9	<u>22 mars 2021 10:56</u>

Dernier accès au gestionnaire de configuration

-  Modules
-  Log du dernier accès à Git
-  Renommer

Dernier Log du dernier accès à Git

```
Started on Nov 27, 2022, 9:42:00 PM
Using strategy: Default
[poll] Last Built Revision: Revision b2c696c6df97f54c1e2ce80e9ede62d4d4c36493 (refs/remotes/origin/master)
The recommended git tool is: NONE
No credentials specified
> git --version # timeout=10
> git --version # 'git version 2.30.2'
> git ls-remote -h -- /home/project/cours_ic # timeout=10
Found 1 remote heads on /home/project/cours_ic
[poll] Latest remote head revision on refs/heads/master is: b2c696c6df97f54c1e2ce80e9ede62d4d4c36493 - already built by 16
Done. Took 55 ms
No changes
```

ENVIRONNEMENT DE BUILD

Environnements de Build

☐ Use secret text(s) or file(s) [?](#)

BUILD

Build Steps

Ajouter une étape au build ▴

Exécuter un script shell

Exécuter une ligne de commande batch Windows

Invoquer les cibles Maven de haut niveau

BUILD

Script shell

Build Steps

≡

Exécuter un script shell ?

×

Commande

Voir [la liste des variables d'environnement disponibles](#)

echo "Version 1.0" > version.txt

Avancé...

Ajouter une étape au build ▾

Build Maven

Build Steps

≡

Invoquer les cibles Maven de haut niveau ?

×

Version de Maven

Maven

▾

Cibles Maven

clean install

▾

Avancé...

Ajouter une étape au build ▾

JENKINS – INFORMATIONS SUR L’EXÉCUTION

Sur le tableau de bord

TP

Tous

+

S	M	Nom du projet ↓	Dernier succès	Dernier échec	Dernière durée	
✓	⚙️	FirstMaven	36 mn #16	1 an. 8 mo. #10	1 mn 9 s	▶
✗	🔗	FirstPipeline	1 an. 8 mo. #5	1 an. 8 mo. #6	9.5 s	▶

Sur le job

📄 Log du dernier accès à Git

✎ Renommer

⚙️ Historique des builds

tendance ▼

🔍 Filter builds...

✓ [#16](#) 27 nov. 2022 21:30

✓ [#15](#) 17 févr. 2022 07:40

✓ [#14](#) 22 mars 2021 11:09

⚠️ [#13](#) 22 mars 2021 11:08

✓ [#12](#) 22 mars 2021 11:07

📄 Derniers résultats des tests (aucune erreur)

Liens permanents

- Dernier build ([#16](#)), il y a 44 mn
- Dernier build stable ([#16](#)), il y a 44 mn
- Dernier build avec succès ([#16](#)), il y a 44 mn
- Dernier build en échec ([#10](#)), il y a 1 an. 8 mo.
- Dernier build instable ([#13](#)), il y a 1 an. 8 mo.
- Dernier build non réussi ([#13](#)), il y a 1 an. 8 mo.
- Last completed build ([#16](#)), il y a 44 mn

Console

↑ Retour au projet

📄 État

</> Modifications

📄 Sortie de la console

📄 Voir en texte brut

✓ Informations de la construction

🗑 Supprimer le build "#16"

✓ Sortie de la console

Started by user [admin](#)

Running as SYSTEM

Building in workspace /root/.jenkins/workspace/FirstMaven

Unpacking <https://repo.maven.apache.org/maven2/org/apache/maven/apache-maven/3.8.6/apache-maven-3.8.6-bin.zip> to /root/.jenkins/tools/hudson.tasks.Maven_MavenInstallation/Maven on Jenkins

The recommended git tool is: NONE

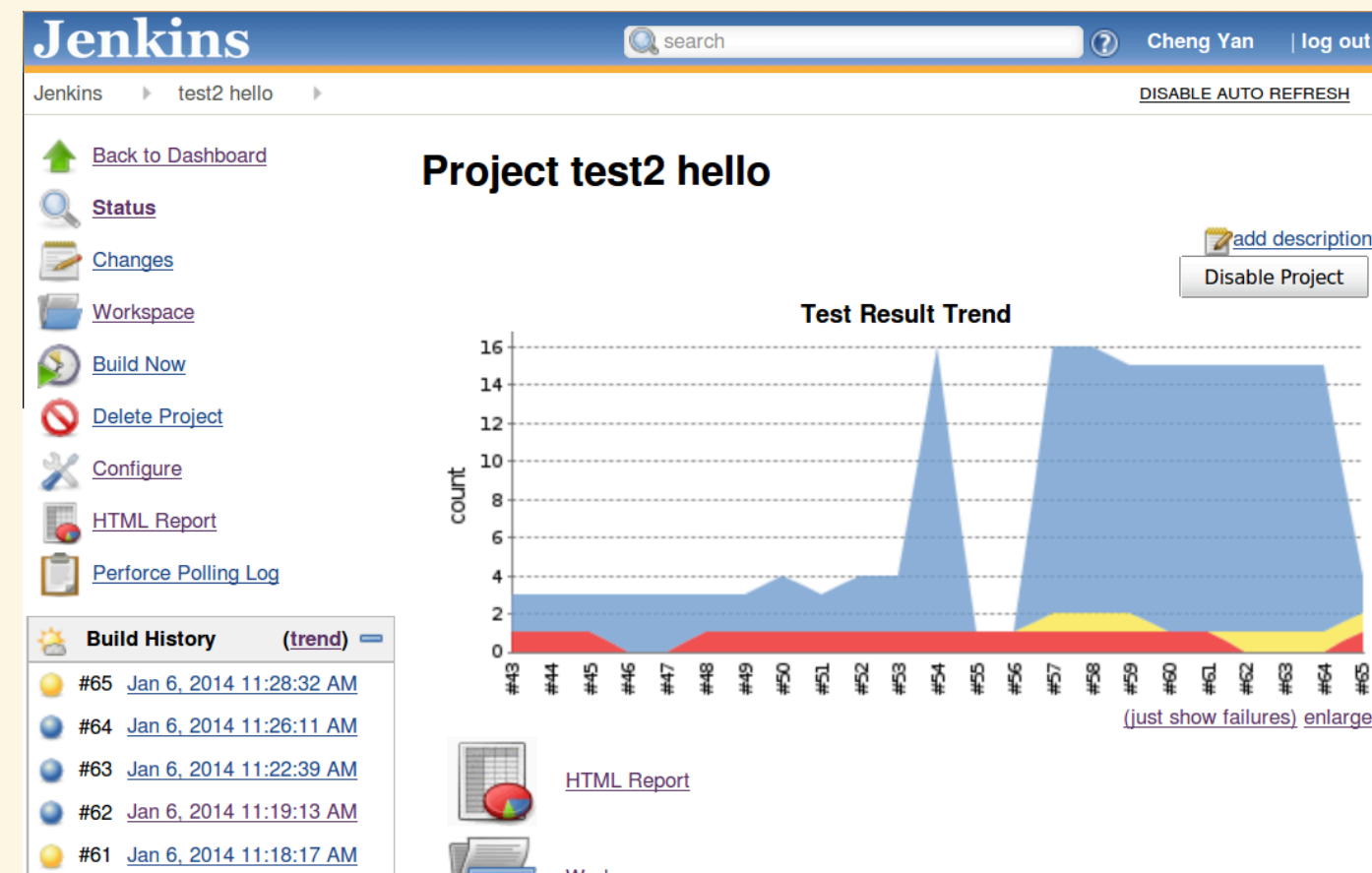
No credentials specified

Cloning the remote Git repository

Cloning repository /home/project/cours ic

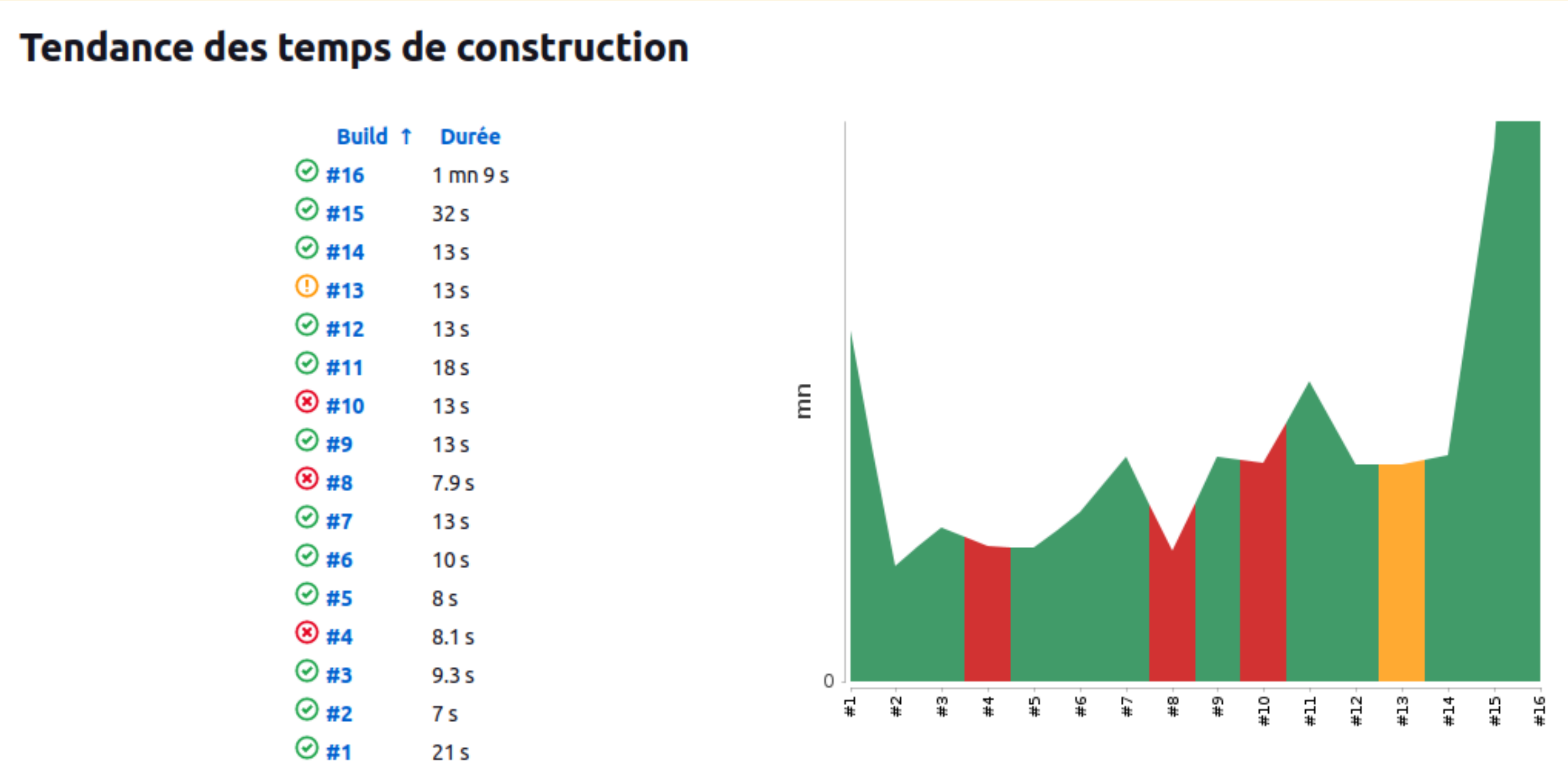
JENKINS – TESTS

- En cas d'échec des tests, l'état du build devient instable
- Le résultat des tests est affiché
- On peut accéder au rapport détaillé d'exécution des tests



JENKINS – TENDANCE

- Visualisation du temps de construction du build



ACTIONS À LA SUITE DU BUILD

Consolider les résultats des tests en aval

Archiver des artefacts

Construire d'autres projets (projets en aval)

Déployer les artefacts dans le repository Maven

Publish HTML reports

Enregistrer les empreintes numériques des fichiers pour en suivre l'utilisation

Git Publisher

Ajouter une action après le build ^

JENKINS – NOTIFICATION

Il est possible (recommandé) d'envoyer une notification en cas d'échec du build

L'échec d'un build indique un problème qu'il faut régler au plus vite.

La notification ne doit pas être considérée comme une simple information mais comme une alerte.

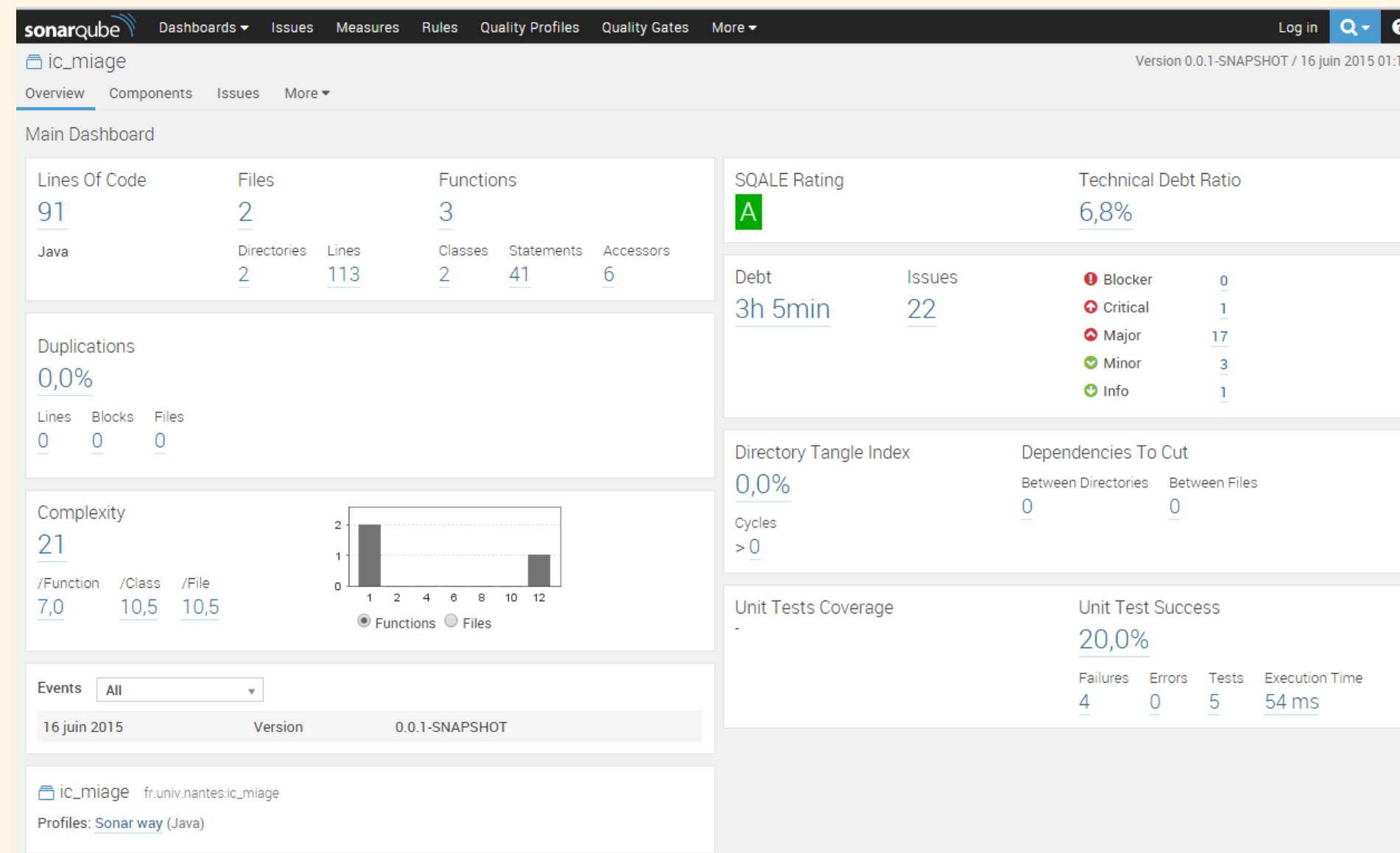
D'autres moyens peuvent être utilisés pour signaler le problème



La notification peut être 'ciblée' sur celui qui a cassé le build (nécessite d'ajouter les utilisateurs)

JENKINS – EXÉCUTION POST BUILD: ANALYSE DE CODE

- D'autres actions peuvent être réalisées en plus de la construction
- On peut déclencher une analyse de code (Sonar, Checkstyle, ...)
- Le rapport d'analyse est généré et accessible via des plugins



JENKINS – ORGANISATION DES BUILDS

- La multiplication des analyses augmente le temps de traitement et la charge du serveur d'intégration
- Certaines analyses ont moins d'intérêt à être réalisées en continue
- Créer plusieurs jobs avec des objectifs et des fréquences différents
- Exemple:
 - En continue : compilation, tests unitaires
 - Quotidien : analyse de code, javadoc, test bout en bout
 - Hebdomadaire: test de charge, test multi-navigateur

JENKINS – MODE MAÎTRE / ESCLAVE

- Le maître
 - définit les esclaves
 - agrège les données
- Les esclaves
 - répartir la charge
 - rendre scalable
 - permet d'avoir des environnements spécifiques (OS, version java, ...)

JENKINS – PIPELINE

- Description du job sous forme de code
- Permet la réutilisation
- Historisation du job dans le gestionnaire de source
- Exécution d'étapes en parallèle

✓ Pipeline < 43

PipelineModificationsTestsArtefacts↺⚙️🔗Déconnexion✕

Branche: —8sAucune modification

Commit: —2 minutes agoDémarré par l'utilisateur admin

StartCheckoutBuild and testPost buildEnd

✓✓✓✓

CheckstyleJavadoc

Post build / Javadoc - <1s

🔄 Restart Post build🔗📄

✓	> Maven — Use a tool from a predefined Tool Installation	<1s
✓	> Fetches the environment variables for a given tool in a list of 'FOO=bar' strings suitable for the withEnv step.	<1s
✓	> mvn javadoc:javadoc — Shell Script	4s
✓	> Publish HTML reports	<1s

✓ Pipeline < 43

PipelineModificationsTestsArtefacts↺⚙️🔗Déconnexion✕

JENKINS – PIPELINE

```
pipeline {
  agent any
  tools {
    maven 'Maven'
  }
  stages{
    stage('Checkout') {
      steps {
        checkout scm: [
          $class: 'GitSCM',
          branches: [[name: '*/master']],
          userRemoteConfigs: [[url: 'file:///home/project/cours_ic']]
        ]
      }
    }
    stage('Build and test') {
      steps {
        sh "mvn install"
      }
    }
    stage('Post build') {
      parallel {
        stage('Javadoc') {
          steps {
            sh "mvn javadoc:javadoc"
            // publish html
            publishHTML target: [
              reportDir: 'target/site/apidocs',
              reportFiles: 'index.html',
              reportName: 'Javadoc'
            ]
          }
        }
        stage('Checkstyle') {
          steps {
            sh "mvn checkstyle:checkstyle"
            publishHTML target: [
              reportDir: 'target/site',
              reportFiles: 'checkstyle.html',
              reportName: 'Checkstyle'
            ]
          }
        }
      }
    }
  }
}
```

QUESTIONS