

Qu'est ce qu'une Intégration Continue ?

2.1 Recherches : qu'est-ce qu'une CI ?	
En tant que :	Je veux :
<i>Non applicable</i>	Savoir en quoi consiste une CI et connaître son utilité
Description : Recherches dans le but de mieux connaître le domaine de l'intégration continue	
Definition Of Done : <ul style="list-style-type: none">- Résumé d'au moins 2 pages : concept, fonctionnement et utilité d'une CI ;- Liste des 10 outils les plus utilisés, avec leurs points positifs et négatifs ;- Conclusion : les 3 outils à retenir et à tester (avec justification).	
Charge estimée :	Responsable :
1 J/H	William Rech

Qu'est ce qu'une Intégration Continue ?

CI : Continuous Integration - Intégration Continue

Le rôle d'une intégration continue est d'intégrer les changements apportés au code d'un projet en les testant de manière continue, afin de détecter et de corriger immédiatement les éventuelles erreurs.

Chaque intégration est vérifiée et testée par un build automatisé (Avec des tests fonctionnels et unitaires).

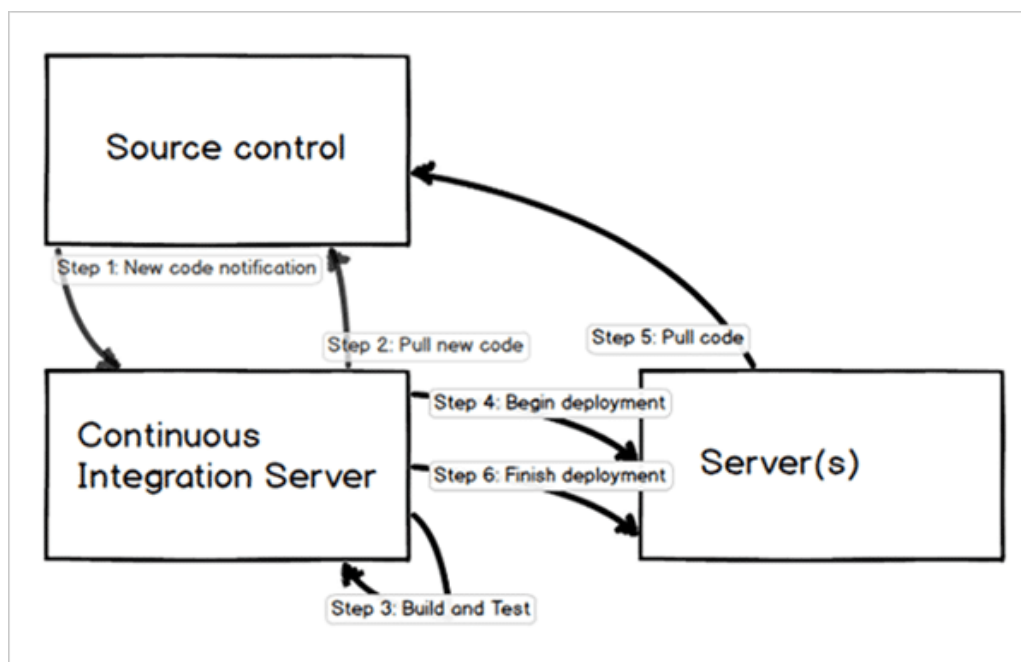
Pourquoi mettre en place une CI :

- Intégration fréquente du nouveau code
- Identification et résolution des bugs plus rapides
- Evite que l'équipe ne travaille à partir de code buggé
- Éviter les problèmes de merge
- Communication de l'avancée (pull request)

En conclusion, le code sur la branche de production est totalement fonctionnel

Une CI nécessite un système de contrôle de version (VCS)

par exemple *Git*, *Mercurial* et *Subversion (svn)*



Étape 1: Il y a un nouveau code écrit par le développeur ou l'Automator. Par conséquent, une nouvelle notification de code arrive au serveur d'intégration continue comme Jenkins. La notification peut être effectuée via des hooks ou le serveur CI peut interroger.

Étape 2: Le serveur CI extrait le nouveau code de l'outil de contrôle de source comme svn ou perforce.

Étape 3: Le serveur CI lance le démarrage des tests de construction. En fonction du pourcentage de réussite, le déploiement commence.

Étape 4: Le processus de déploiement est démarré par le serveur CI. Et le nouveau code est déployé sur le serveur indiqué par **Étape 5**.

Étape 6: Enfin, les services de déploiement sont terminés par le serveur CI.

Qu'est ce qu'une Intégration Continue ?

Vocabulaire

Test Unitaire : procédure permettant de vérifier le bon fonctionnement d'une partie précise du code.

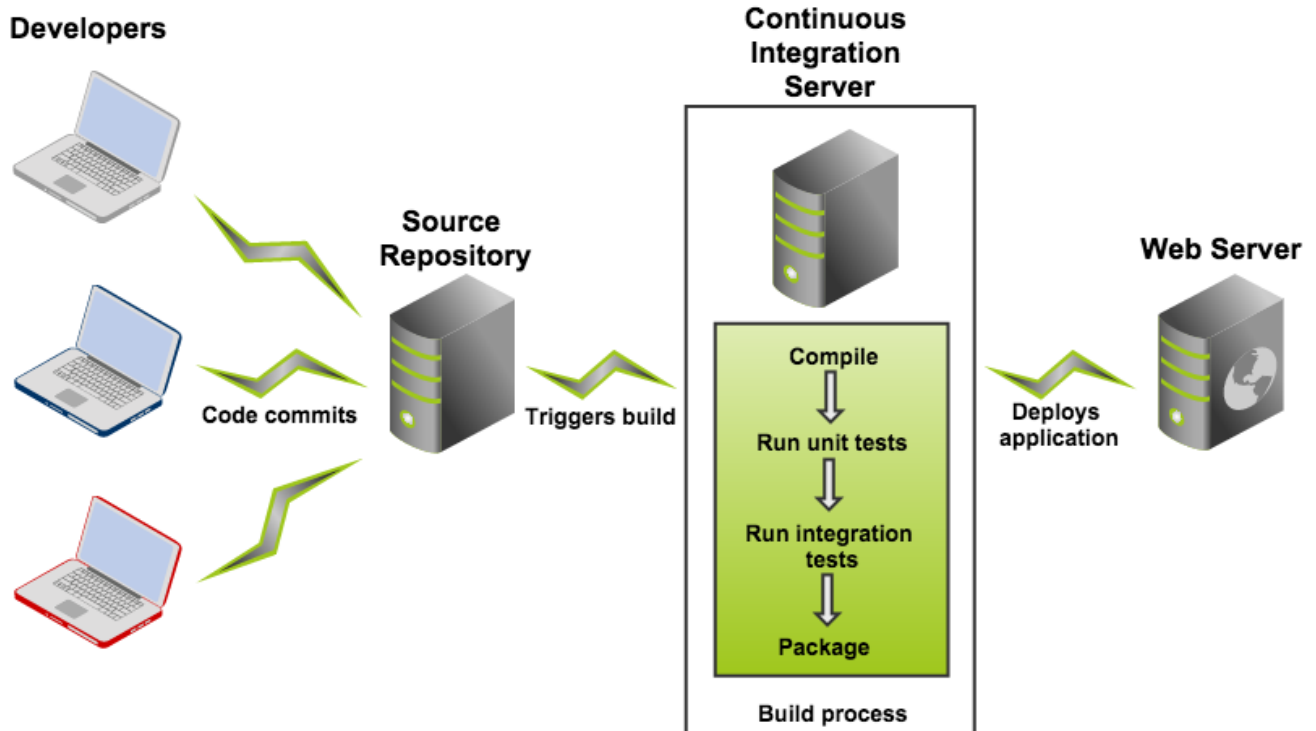
Test fonctionnel : vérification de fonctionnalités afin détecter de potentiels défauts oubliés.

Workflow : flux de travail.

Pipelines : série d'étapes à réaliser en vue de valider l'intégration d'une fonctionnalité.

Jobs : tâche / étape d'un pipeline

Continuous Integration



Qu'est ce qu'une Intégration Continue ?

Voici 10 outils d'intégration continue listés dans l'ordre de popularité :

Les aspects positifs et négatifs sont relatifs aux besoins de notre projet

GitLab CI

Positif :

- Populaire
- Configuration facile
- Version gratuite (restriction)
- Fonctionnalités agiles intégrées

Négatif :

- Limitation version gratuit
- Propre hébergeur

Restrictions :

- 5 GO de stockage
- 10 GO de transfert par mois
- 400 minutes de build par mois
- 5 utilisateurs par namespace



Jenkins

Positif :

- Totalement gratuit
- Simple
- Enrichissement des fonctionnalités à l'aide de plugins

Négatif :

- Nécessite un hébergeur



CircleCI

Positif :

- Diversité des builds (Build for Docker, Windows, Linux, Arm, and macOS)
- Version gratuite (restreinte)
- Hébergeable (restreint)

Négatif :

- Limitation version gratuite

Restriction :

- 6 000 minutes de build par mois
- 30 jobs à la fois et parallélisme des tests
- 2 GO de stockage
- 1 GO de transfert
- 40 jobs simultanés maximums



Qu'est ce qu'une Intégration Continue ?

GitHub Action

Positif :

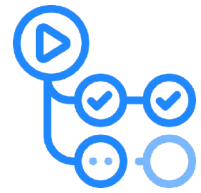
- Version gratuite
- Intégré à Github

Négatif :

- Limitation version gratuite

Restriction :

- 1 GO de stockage
- 2 000 minutes de build par mois
- Multiplicateur de minute pour les distributions non Linux (ex : Windows => x2)



Bamboo

Positif :

- /

Négatif :

- Payant



Buddy

Positif :

- Version gratuite (restreinte)
- Facile à utiliser

Négatif :

- Limitation version gratuite

Restrictions :

- 5 projects
- 120 pipeline lancés par mois
- 512 MO Cache storage
- 1 GO RAM
- 2 CPU virtuel (vCUP)
- 1 pipeline simultané
- 1 action parallèle



TeamCity

Positif :

- /

Négatif :

- Payant ou nécessite un hébergeur



GoCD

Positif :

- Totalement gratuit

Négatif :

- Nécessite un hébergeur



Qu'est ce qu'une Intégration Continue ?

Codship

Positif :

- Simple d'utilisation

Négatif :

- Payant



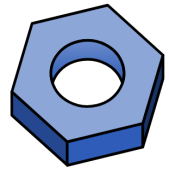
Buildbot

Positif :

- Prise en charge de subversion (svn)

Négatif :

- Peu utilisé
- Nécessite un hébergeur
- Ne prend pas en charge Github



Pour déterminer les 3 choix finaux, nous avons priorisé la simplicité et la gratuité (hors version d'essai) de la solution d'intégration continue.

Les outils retenus sont :

- **GitLab CI**
 - Facilement configurable
 - Populaire
 - Version gratuite intéressante
 - Fonctionnalités agiles intéressantes
- **Github Action**
 - Pour son intégration directe à Github
 - Version Pro (gratuite pour Epitech) intéressante
- **Circle CI**
 - Populaire
 - Version gratuite intéressante
 -



Qu'est ce qu'une Intégration Continue ?

Notes supplémentaires

Sources ressources Top CI :

<https://bitbar.com/blog/top-continuous-integration-tools-for-devops/>
<https://www.atlassian.com/fr/continuous-delivery/continuous-integration/tools>
<https://www.guru99.com/top-20-continuous-integration-tools.html>
<https://geekflare.com/fr/best-ci-tools/>
<https://katalon.com/resources-center/blog/ci-cd-tools>

Compteur de références :

GitLab CI 6
Jenkins 5
CircleCI 5
GitHub Action ?
Bamboo 5

Buddy 4
TeamCity 3
GoCD 3
Codeship 3
Buildbot 2
Wercker 2
Semaphore 2
Nevercode 2
TravisCI 2

Bitbucket Pipelines 1
AWS CodePipeline 1
Azure Pipelines 1
Integrity 1
Strider 1
Autorabit 1
Final builder 1
Buildkite 1
CruiseControl 1
Bitrise 1
Urbancode 1
AppVeyor 1
Harness 1
Kraken 1
Spinnaker 1

<https://mobiskill.fr/blog/conseils-emploi-tech/tests-unitaires-vs-tests-fonctionnels-queelles-differences/#:~:text=Les%20tests%20fonctionnels%20sont%20%C3%A9galement,unitaires%20et%20des%20tests%20fonctionnels.>

<https://resources.github.com/ci-cd/>

Pour aller plus loin :

<https://youtu.be/h9K1NngwUvE>
<https://youtu.be/ws1qGuFMYlc>