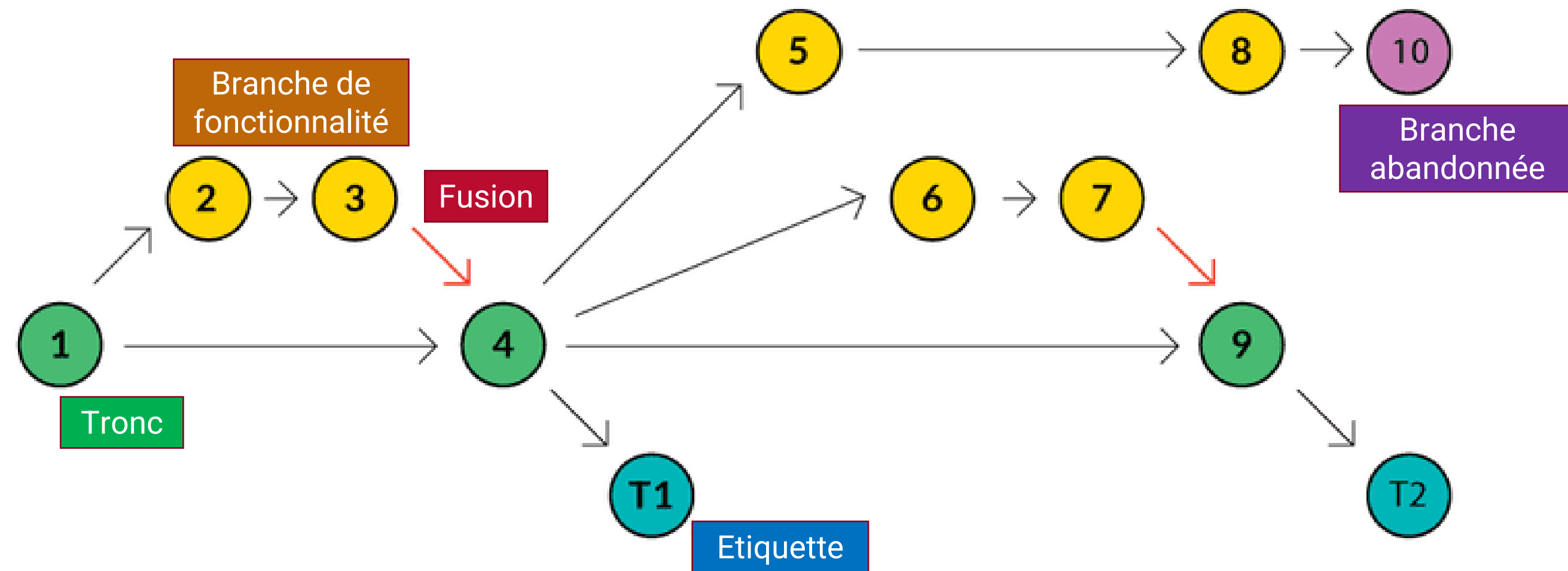
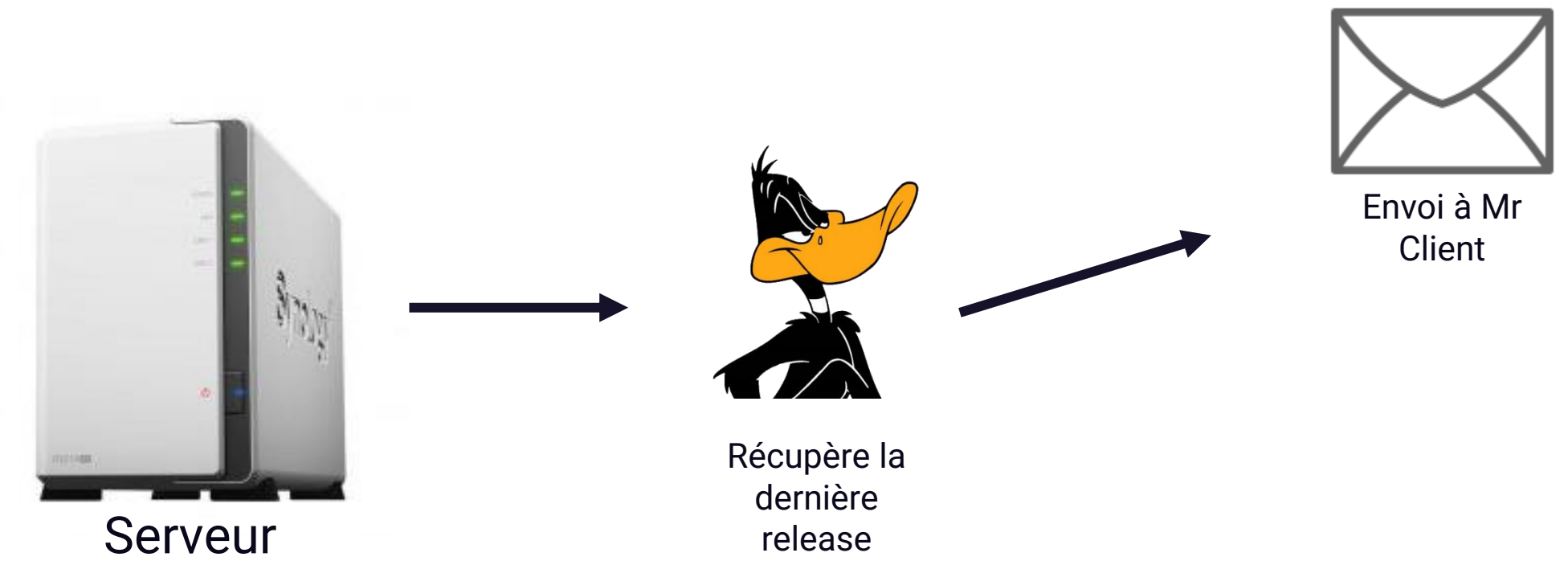
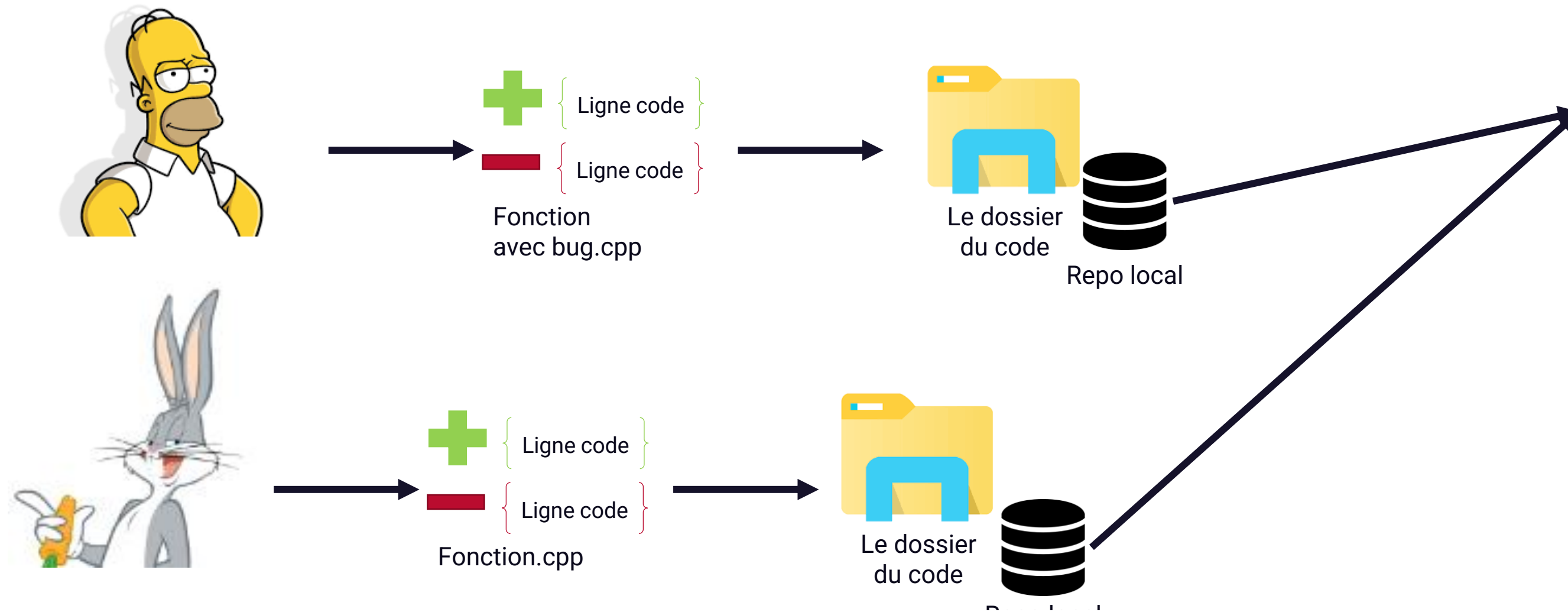


GIT : infrastructure et utilisation

- Git : rappels
- Infrastructure Tiflex
- Utilisation

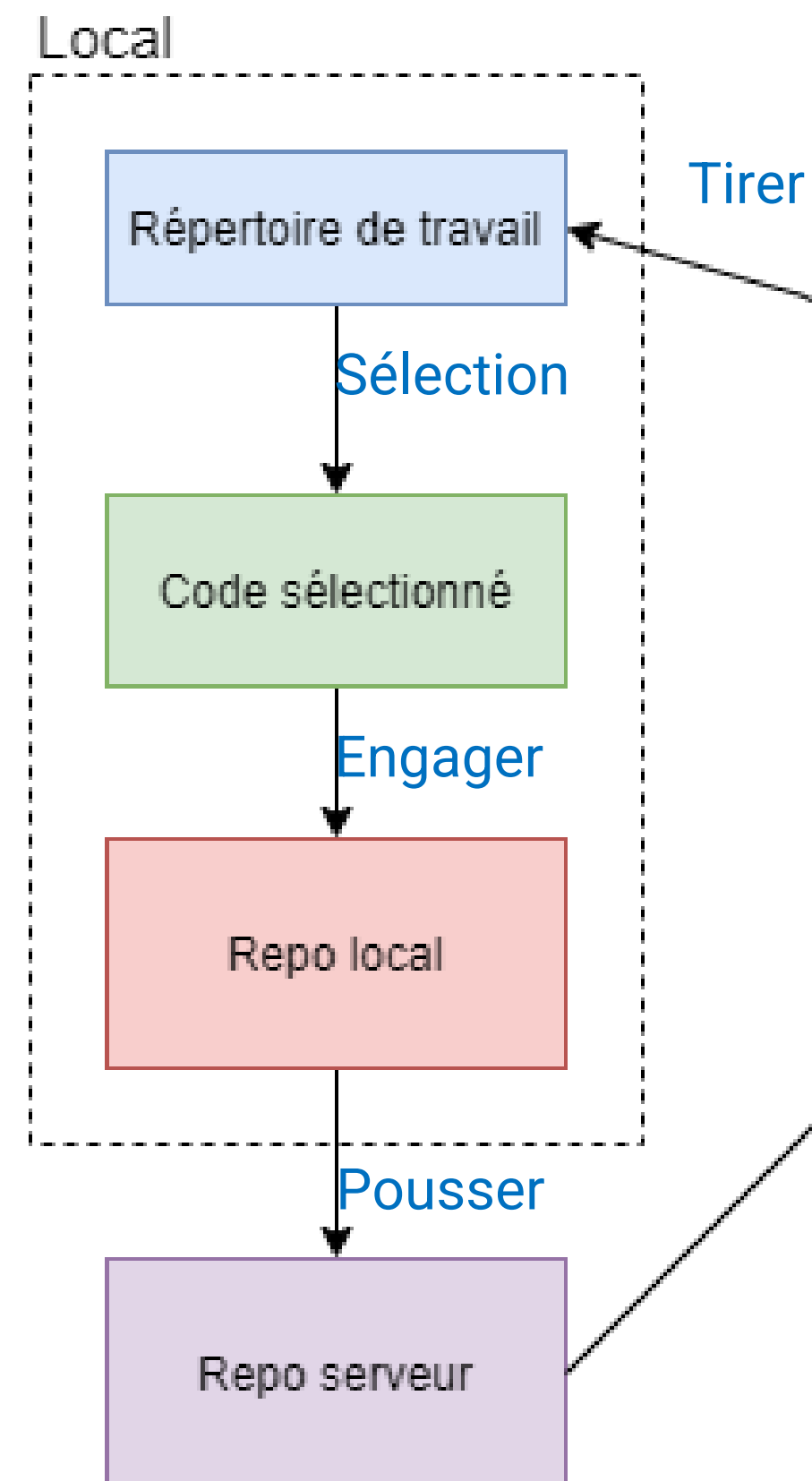
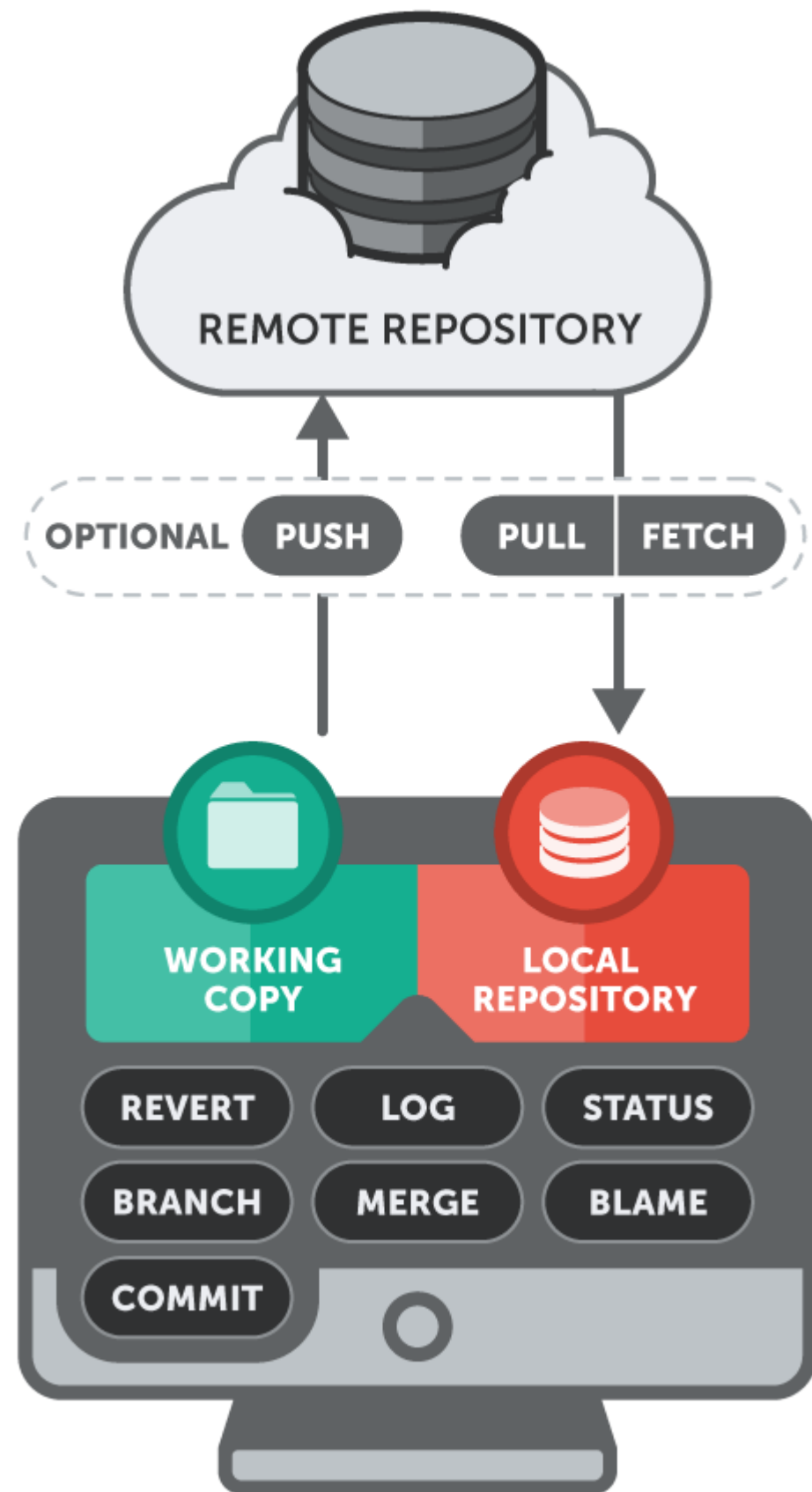
Versionning avec Git



Dépôt	Des copies de dépôt, présentes localement, dans lesquelles il est possible de travailler
Droit d'accès	Pour le répertoire complet
Suivi des modifications	Enregistre des données
Journal de modifications des données	Le dépôt et les copies de travail contiennent l'historique complet.
Connectivité au réseau	Nécessaire seulement pour réaliser une synchronisation
Vitesse de transfert	Très rapide pour des fichiers textes standards, moins pour des binaires très lourds
Granularité	Sélection des lignes de codes unitaires ou par fichier Pas de gestion des répertoires vides
Exclusions	Par fichier .gitignore

Termes et procédures

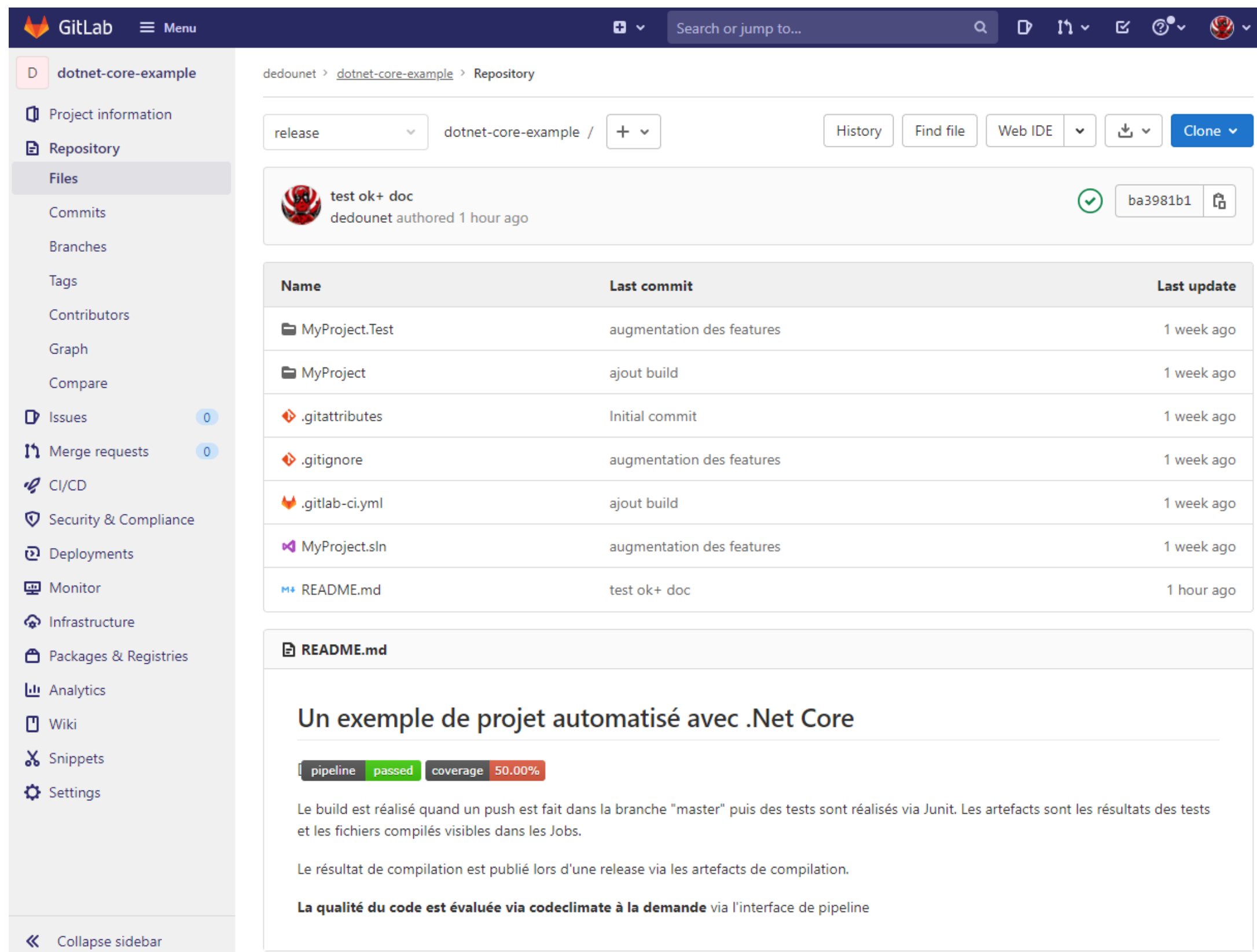
Cheatsheet:
<https://ndpsoftware.com/git-cheatsheet.html>



Terme anglais	Traduction
Branch	Branche
Tag	Etiquette
Release	Version
Merge	Fusion
Merge Request	Demande de fusion
Commit	Soumettre
Push	Pousser/Envoyer
Checkout	Télécharger les fichiers
Pull	Récupérer les objets dans la branche courante
Repository (Repo)	Dépôt (base de donnée)
Diff (Change view)	Evaluation des modifications entre 2 commit/versions
Blame	Voir les auteurs de chaque modifications
Revert	Défait le commit spécifié puis fait un commit du résultat.
Fetch	Récupérer les objets et historiques du serveur
Stash	Remise les modifications en cours sans commit

Infrastructure

GITLAB



gitlab.local



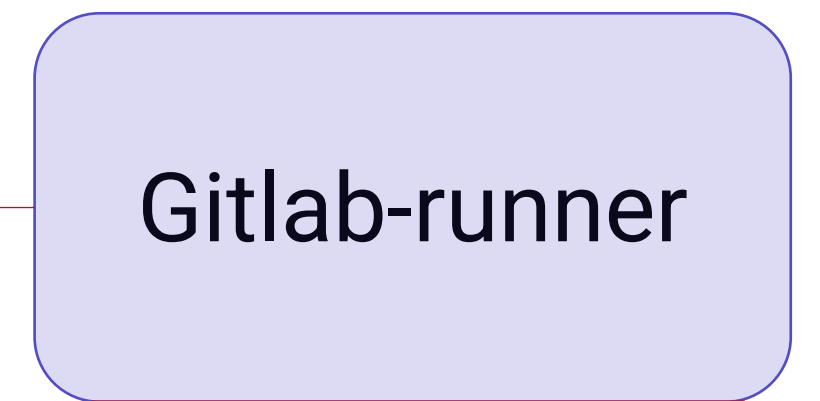
Stockage code
Lien vers les tickets
Intégration Continue
Stockage de nuget
Stockage de VM docker

jira.local/



Gestionnaire de ticket

gitlab-runner.local



Exécuteur Docker

Aller plus loin

Tests unitaires et validation du code



Mise en place d'un « pipeline » (processus) en fonction des caractéristiques (eg: push sur branche release, ajout de tag, variable) d'un commit :

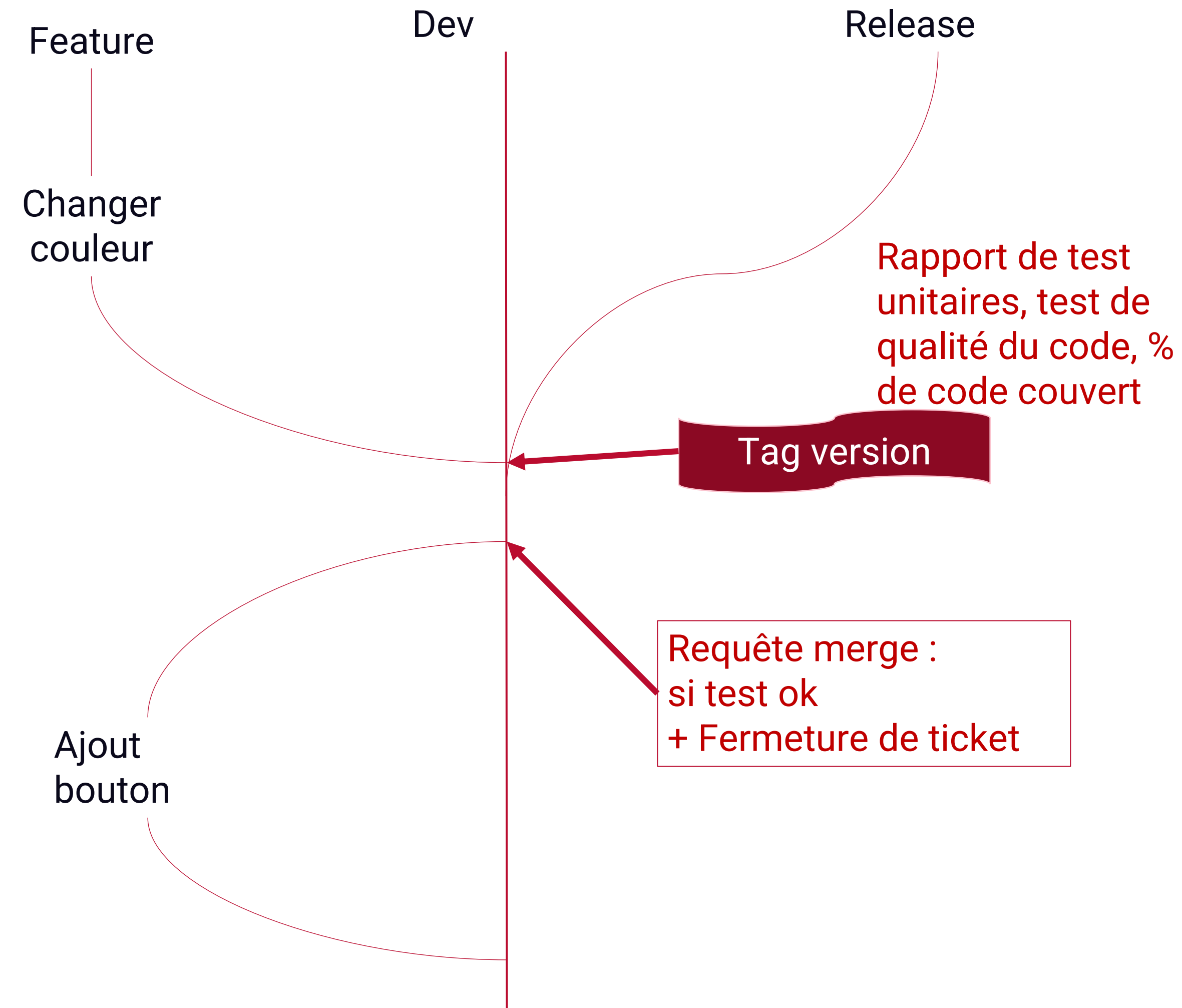
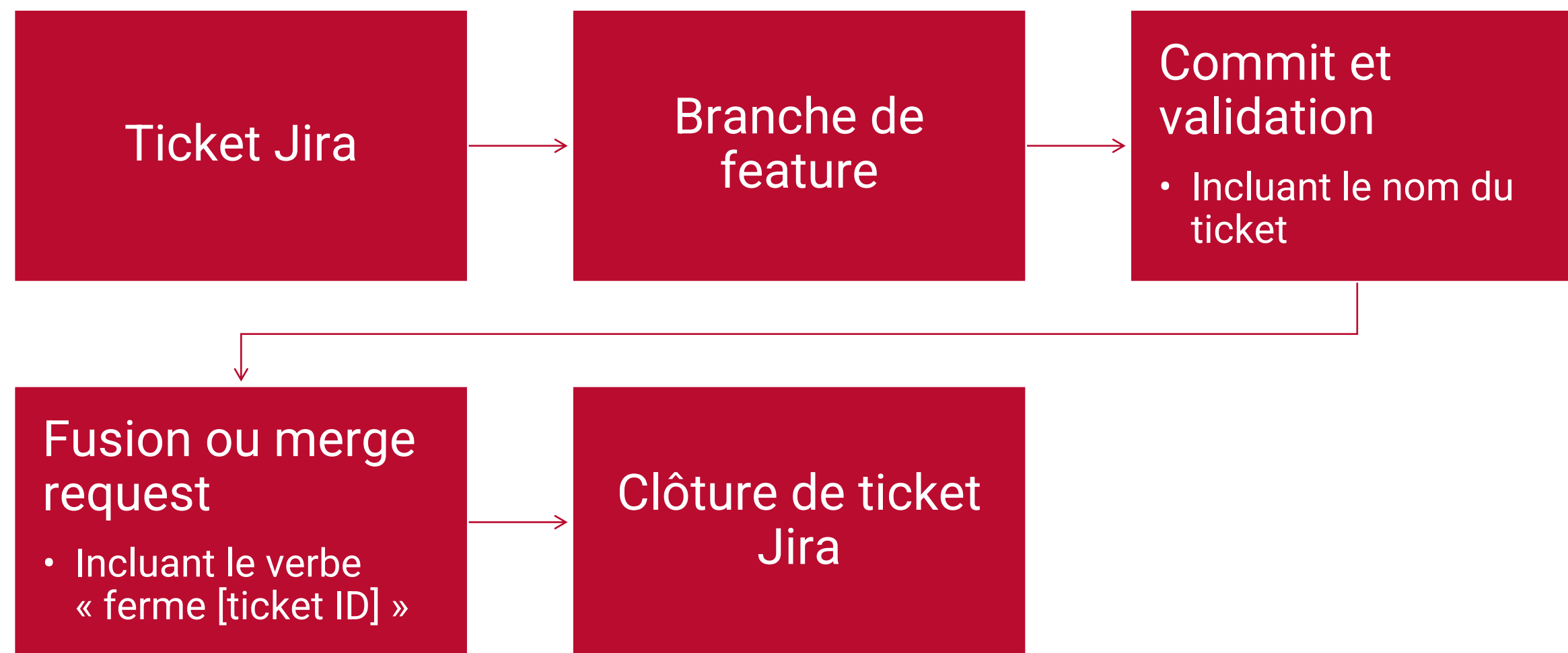
- Tests unitaires + couverture
- Qualité du code + rapport
- Compilation
- Mise à dispo du résultat de compilation (release)
- Documentation automatisée

Fermeture automatique de tickets correspondants

```
public class BasicMaths
{
    public double Add(double num1, double num2)
    {
        return num1 + num2;
    }
    public double Subtract(double num1, double num2)
    {
        return num1 - num2;
    }
    public double divide(double num1, double num2)
    {
        return num1 / num2;
    }
    public double Multiply(double num1, double num2)
    {
        // To trace error while testing, writing + operator instead of * operator.
        return num1 + num2;
    }
}
```

Job	Duration	Failed	Errors	Skipped	Passed	Total
debug	13.21ms	0	0	0	3	3

Processus BE



Actions choisies

Choisir les fichiers nécessaires
UNIQUEMENT

Economiser l'espace, Transférer
uniquement le nécessaire

Réaliser ses « commit » sur des
fonctions simples / bug uniques

Avoir une bonne granularité
de versions

Respecter les
branches

DOCUMENTER SES COMMIT

Pour la CLARTE des explications:
Le code/fichiers montre ce qui a
changé, les messages de commit
expliquent POURQUOI

Penses bêtes

<https://guillaumebriday.fr/comment-jutilise-git-mes-astuces-et-bonnes-pratiques>

<https://git-scm.com/docs>

<https://ndpsoftware.com/git-cheatsheet.html#loc=workspace;>

OBOULOT

Le TP !

Terminal	<pre>git config --global user.name bob git config --global user.email mon@email.com</pre>
Putty	Créer une clé avec puttygen Ajouter la clé dans Gitlab
SourceTree ou autre gui	Installation Création du compte
Interface interne VS2019	Repo déjà configuré

Créer un dépôt

Localement

Créer un dossier
git init
Créer un fichier.txt
git status
Git add fichier.txt
Git commit -m « ajout de fichier.txt »

Ajouter un répertoire /release
Ajouter un fichier /release/monbinaire.bin
Ajouter un fichier readme.md
Vérifier le status
Ajouter un fichier .gitignore
Ajouter les fichiers et vérifier le status

Sur Gitlab

Créer un dépôt
Utiliser l'IDE en ligne / Créer un fichier
Ajouter la modifications
Commiter le contenu

Utiliser l'interface pour créer des fichiers

Cloner un dépôt

Terminal

Récupérer l'adresse du dépôt
Git clone gitlab.tiflex.com

Sur un Gui

Ajouter le repo avec l'interface

Modifier les fichiers pour un commit

Terminal

Créer un fichier
Monfichier.txt
Voir le status des fichiers
git status
Ajouter des fichiers
git add monfichier.txt
Annuler l'ajout
Git reset monfichier.txt

Sur un Gui

Choisir via l'interface
Indexer des changements
Rejeter des changements

Remiser du code

Terminal

Choisir les fichiers à remiser
git stash push 'nom du stash'

Sur un Gui

Choisir les fichiers via l'interface
Donner un nom et remiser

Créer une branche

Terminal

Créer une branche depuis une autre branche
`git checkout -b <mabranche>`
Créer une branche qui suit une branche distante
`git branch -track <mabranche> <mabranchesuleserveur>`

Sur un Gui

Ajouter via l'interface

Faire une fusion de branches

Terminal

Se placer sur la branche à fusionner
git switch <mabranche>
Fusionner
git merge <brancheavecunefeature>

Sur un Gui

Choisir via l'interface