git flow : release planifiée



v0.4 **V0.3 v**0.2 git flow V0.1 develop master

La branche master

La branche develop

historique officiel du projet, versions finales

utilisation de tags



v0.3 **v0.2** git flow develop feature

Les branches de maintenance

les seules créées depuis master, bug fix sur la branche de production, merge dans master (version) et dans develop

Les branches de release

création de la branche de release depuis develop, merge dans master (nouvelle version) et dans develop

Les branches de feature

les features ne sont jamais reversées dans master



github: une branche par feature



github flow: simplification

Le modèle

master

feature

d'une

développement

fonctionnalité ou d'un bug fix

feature branches:

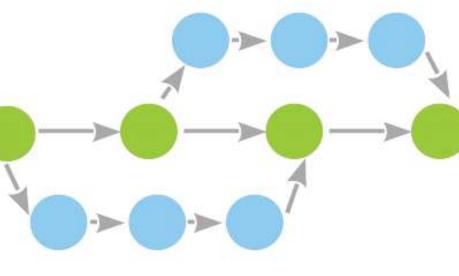
master: production

déploiements fréquents

manque de guide sur la façon de déployer, les environnements de release

est présuppose que le merge d'une feature

fonctionnel et déployable



gitlab: branches d'environnement et d'état



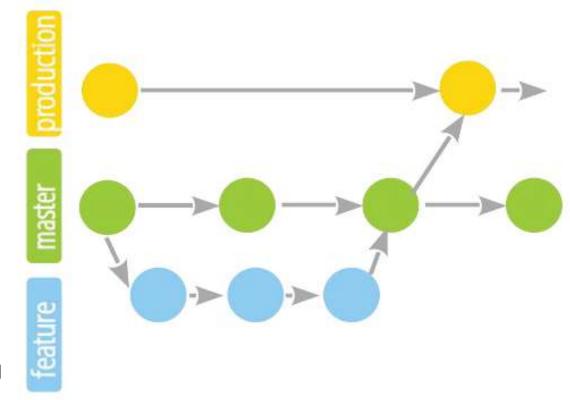
gitlab flow: branche de production

Définition

reflette le code déployé

commit de merge = déploiement

utilisation de tag pour acter le déploiement (plus de précisions sur le moment du déploiement)

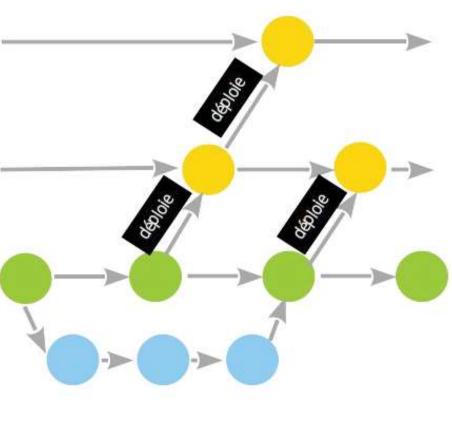




gitlab flow: branche d'environnement

Définition

feature master définit des branches par environnement de déploiement





gitlab flow : branche de release

Définition

contient les versions mineures

ne sont versés que des bugs fixes sérieux :

- merge dans master puis cherry-pick dans la branche de release («upstream first »)
- •1 bug fix => incrémente numéro **PATCH**
- •utilisation de tags: MAJOR.MINOR.PATCH

