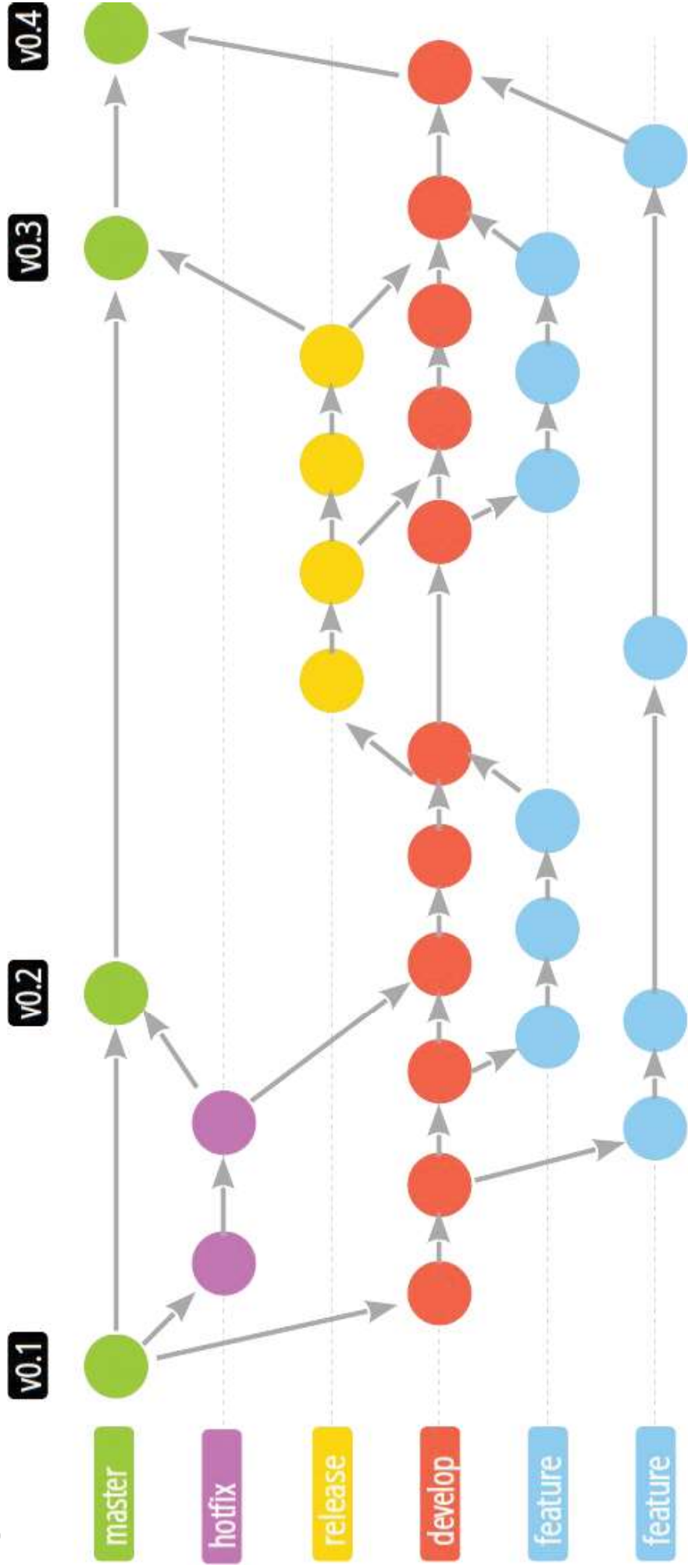


git flow : release planifiée



git flow



◆ La branche master

historique officiel du projet, versions
finales

◆ La branche develop

branche d'intégration pour les features
développées

utilisation de tags



github : une branche par feature



Le modèle

feature

master

feature

présuppose que le merge d'une feature est fonctionnel et déployable

gitlab : branches d'environnement et d'état



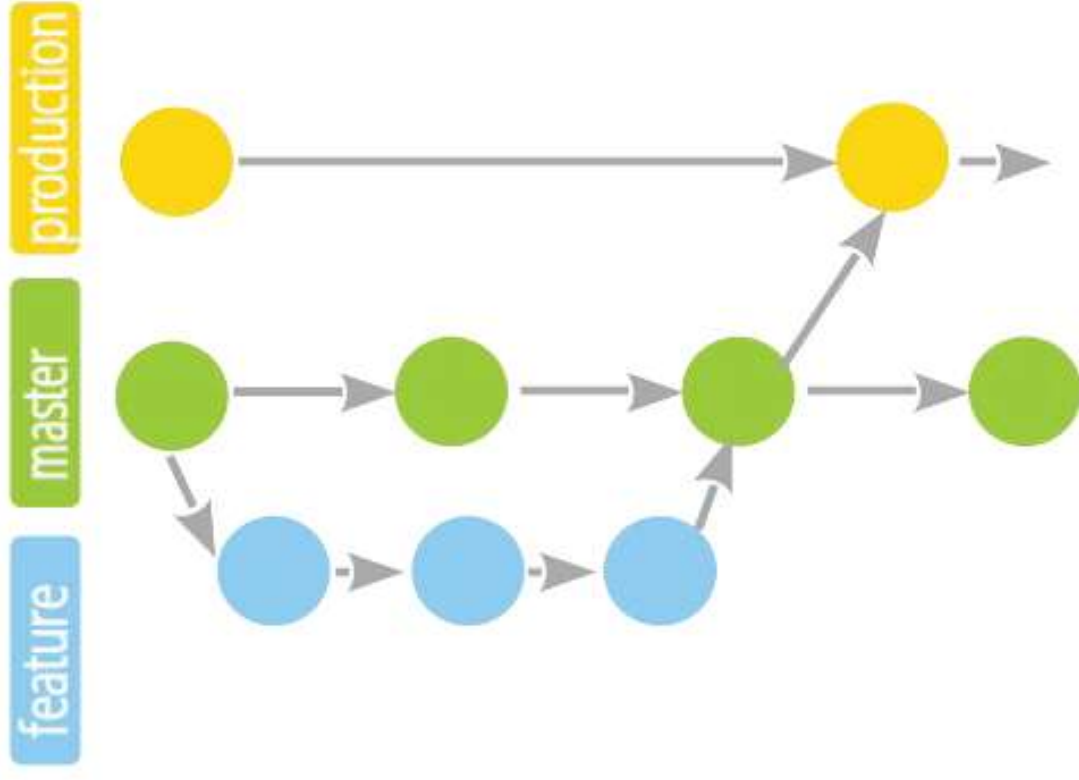
gitlab flow : branche de production

◆ Définition

reflette le code déployé

commit de merge = déploiement

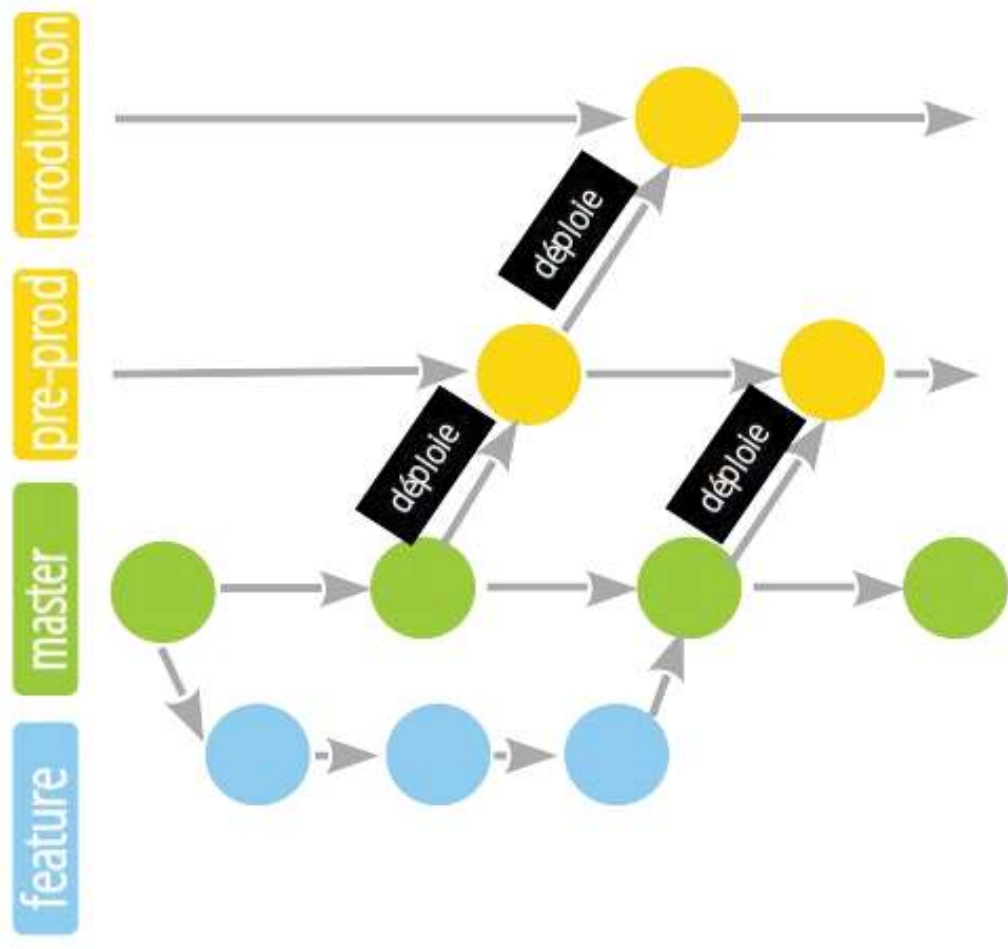
utilisation de tag pour acter le déploiement
(plus de précisions sur le moment du
déploiement)



gitlab flow : branche d'environnement

◆ Définition

définit des branches par environnement
de déploiement



- ◆ Définition
 - 2-3-stable
 - master
 - 2-4-stable

contient les versions mineures

- merge dans master puis cherry-pick dans la branche de release («upstream first»)

- 1 bug fix => incrémenter numéro PATCH
- utilisation de tags : `MAJOR.MINOR.PATCH`

