

# **GIT**

## WORKFLOW

# LE WORKFLOW

DIT

## "GITHUB FLOW"

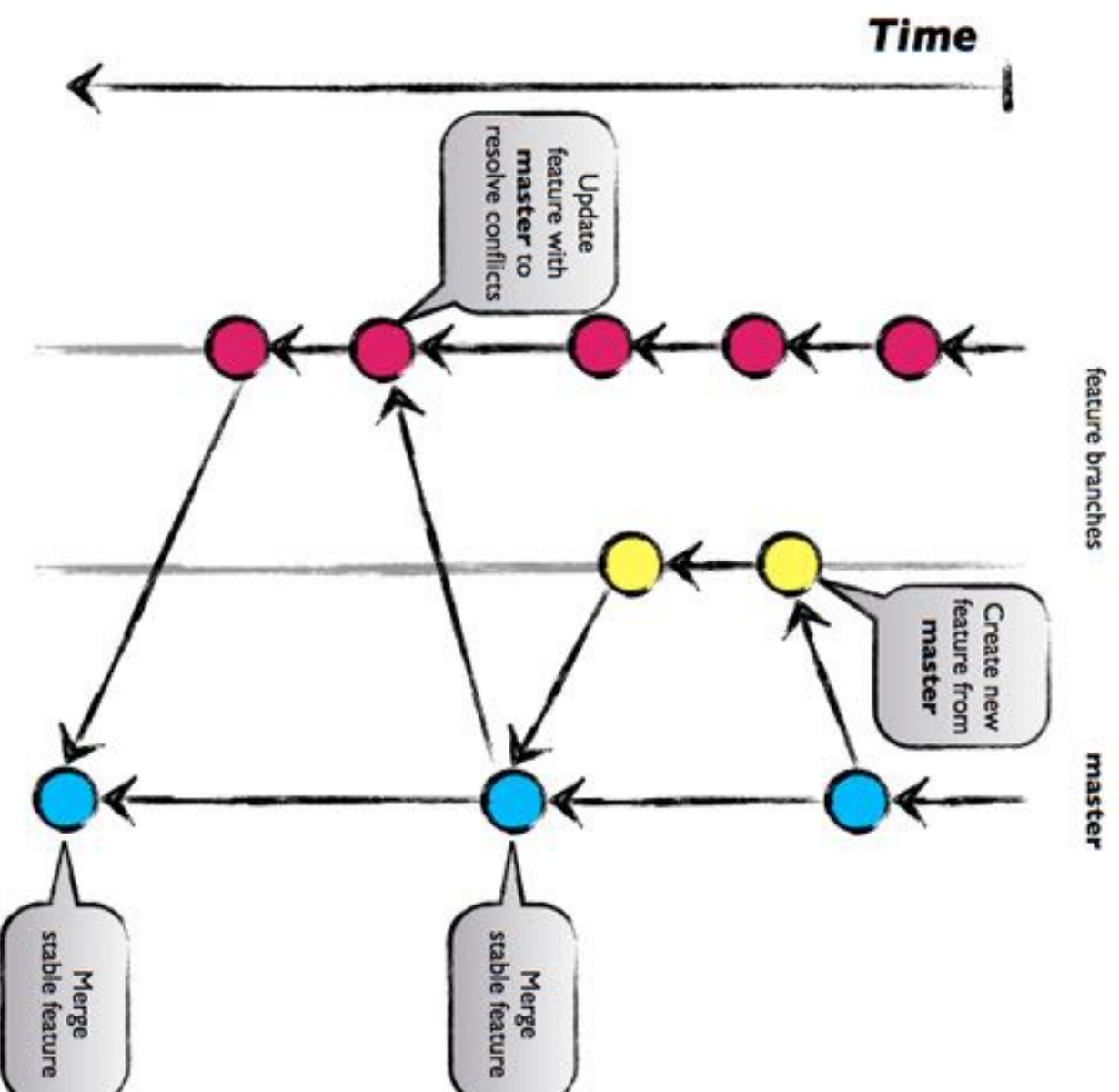
Idéal pour des projets ne nécessitant de versions.

La branche master évolue au fur et à mesure  
des merges et la prod suit cette évolution

# LES BRANCHES

- **Master**
  - Pour la production
  - Doit toujours être stable
  - Toute création de branche est basée sur elle
- **Feature(s)**
  - On en crée autant que l'on souhaite
  - On pousse sur Origin même si elle n'est pas stable
  - On utilise le pull-request
  - On fait faire la revue de code du PR par un autre

# SCHÉMATISATION



# LE WORKFLOW

DIT

## "GIT FLOW"

Ce workflow est plus complet que le précédent.  
Cependant il est plus adapté sur de gros projets.

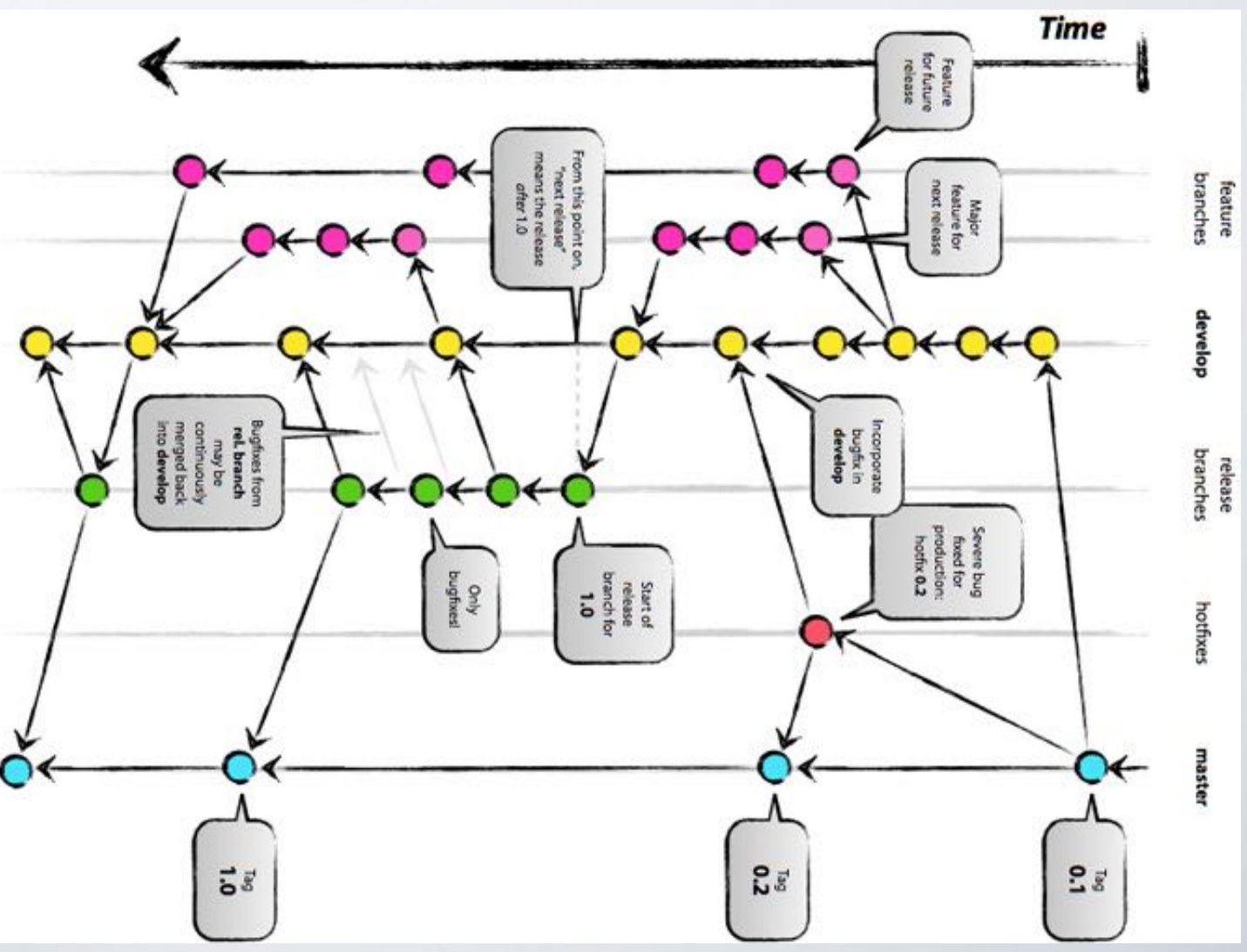
# LES BRANCHES PRINCIPALES

- **master** est la branche où tout est stable. Chaque commit correspond à une release qui peut être déployée en production.
- **develop** est la branche sur laquelle s'effectue le développement

# LES BRANCHES SECONDAIRES

- **release** part de **develop** et se merge dans **master** et **develop**.
- **feature** part de **develop** et se merge dans **develop**.
- **hotfix** part de **master** et se merge dans **master** et **develop**

# SCHÉMA TISATION





# QUELQUES LIENS

Les 2 workflows disposent d'extensions à utiliser en ligne de commande.

- <https://github.com/github-flow/github-flow>
- <https://github.com/nvie/gitflow>