A Git Workflow Model

The big picture

Original blog by Vincent Driessen

http://nvie.com/git-model
http://nvie.com/git-model
http://nvie.com/git-model
Repository relationships

http://nvie.com/git-model
Main branches

**master** used for production releases

**develop** used for integration and regression testing

[http://nvie.com/git-model](http://nvie.com/git-model)
Feature branches

http://nvie.com/git-model
Release branches

- Branched from develop when it almost reflects the desired new release
- Bug fixes for the release are applied here
- No new features may be added to this branch
- New feature development can continue in parallel on the develop branch
Hotfix Branches

Are like release branches, but for bug fixes

http://nvie.com/git-model
Support Branches

• If master has moved on to a new and a hotfix must be applied to a older version (e.g. 1.x):
  – create a support-1.x branch (if none exists) based on the newest 1.x tag in master
  – create a branch (e.g. hotfix-1.1.1), based on support-1.x
  – fix the bug and merge hotfix-1.1.1 back into support-1.x
  – Do this for other older major releases as necessary

• The support branch effectively becomes a master branch for a past version.