GITFLOW WORKFLOW

Amnon Ben Zion

About Me

- Amnon Ben Zion
- 28 Years Old
- DevOps Engineer @ CyberArk
- Former DevOps Engineer @ Cybereason
  - *Dozens of production releases*
- Escaped more than 100 rooms

Agenda

- Gitflow Intro
- Gitflow use cases
- A “real” example
- Summary
- Tips

Introduction

- Gitflow is a **branching strategy** that dictates what kind of branches to set up and how to merge them together
- First introduced by *Vincent Driessen* in 2010
- Contains **5 different branch types**

Use Cases

- When **should** I use it
  - *Support **multiple versions** at the same time*
  - *Scheduled release cycle*

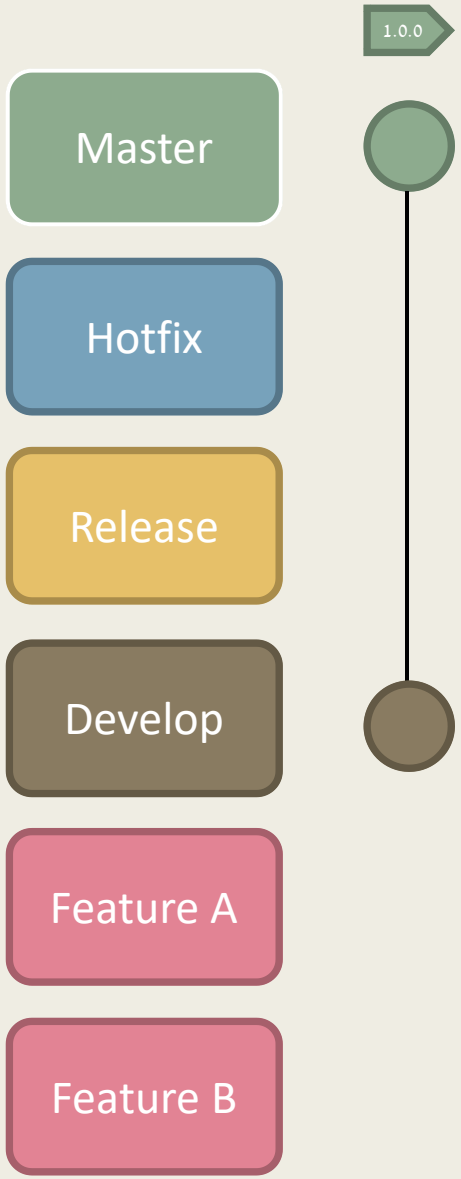
Use Cases

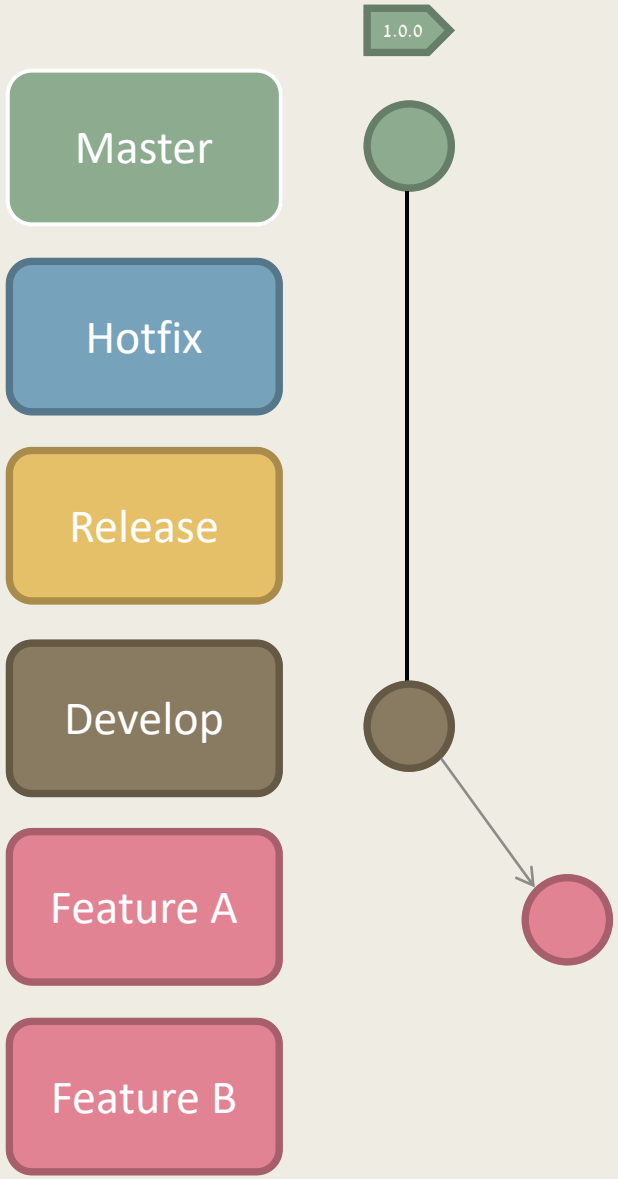
- When **should** I use it
  - *Support **multiple versions** at the same time*
  - *Scheduled release cycle*

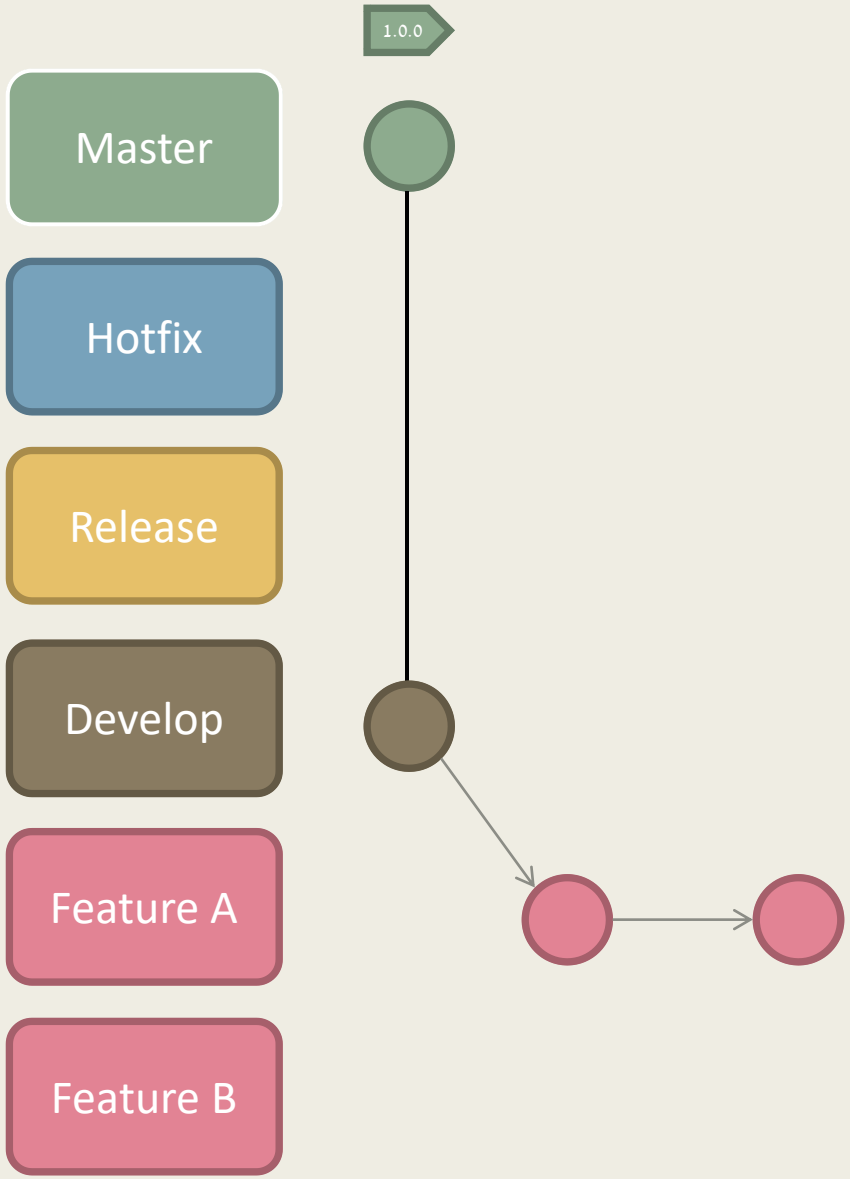
- When **shouldn't** I use it
  - ***SaaS** applications (one available version)*
  - *When deployment happen **daily***

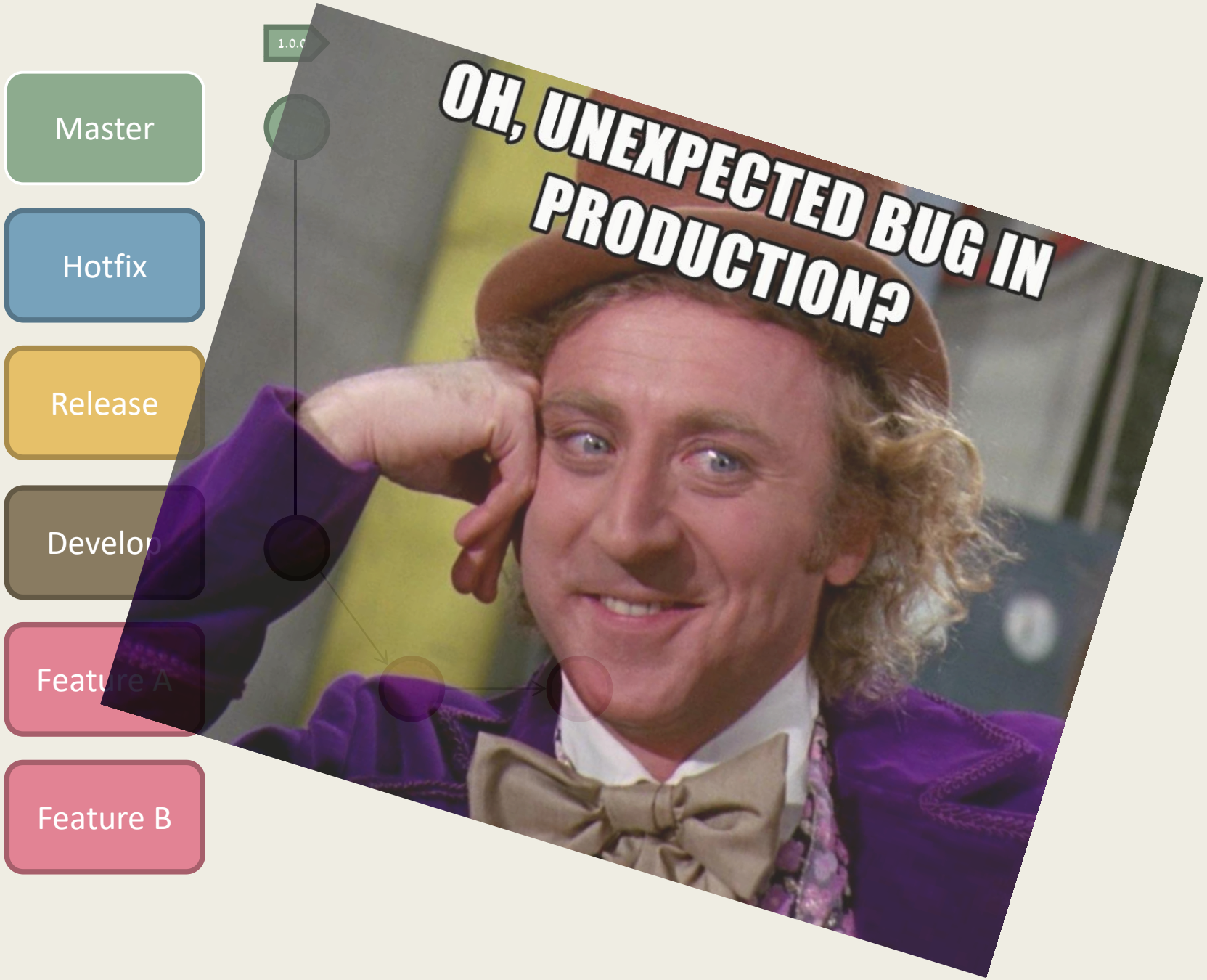
LET ME TELL YOU A LITTLE STORY
memegenerator.net

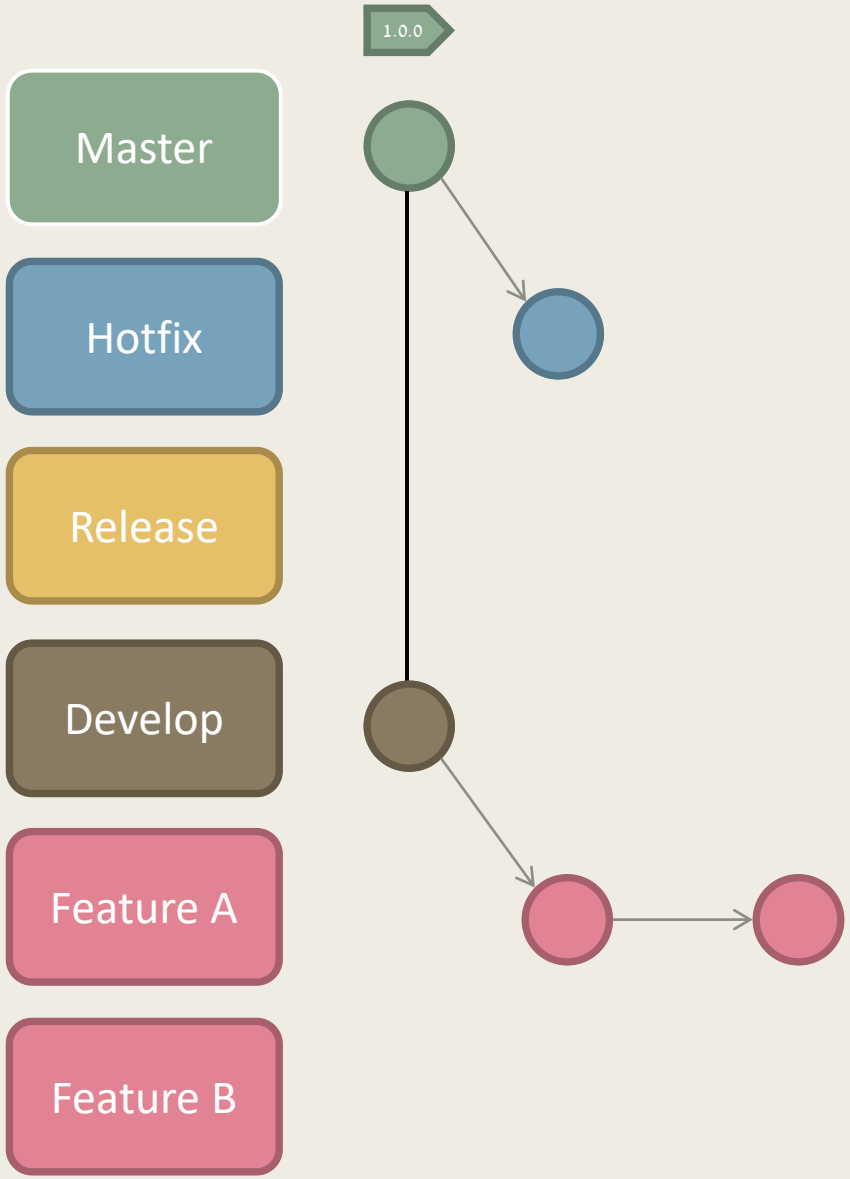
Master
Hotfix
Release
Develop
Feature A
Feature B

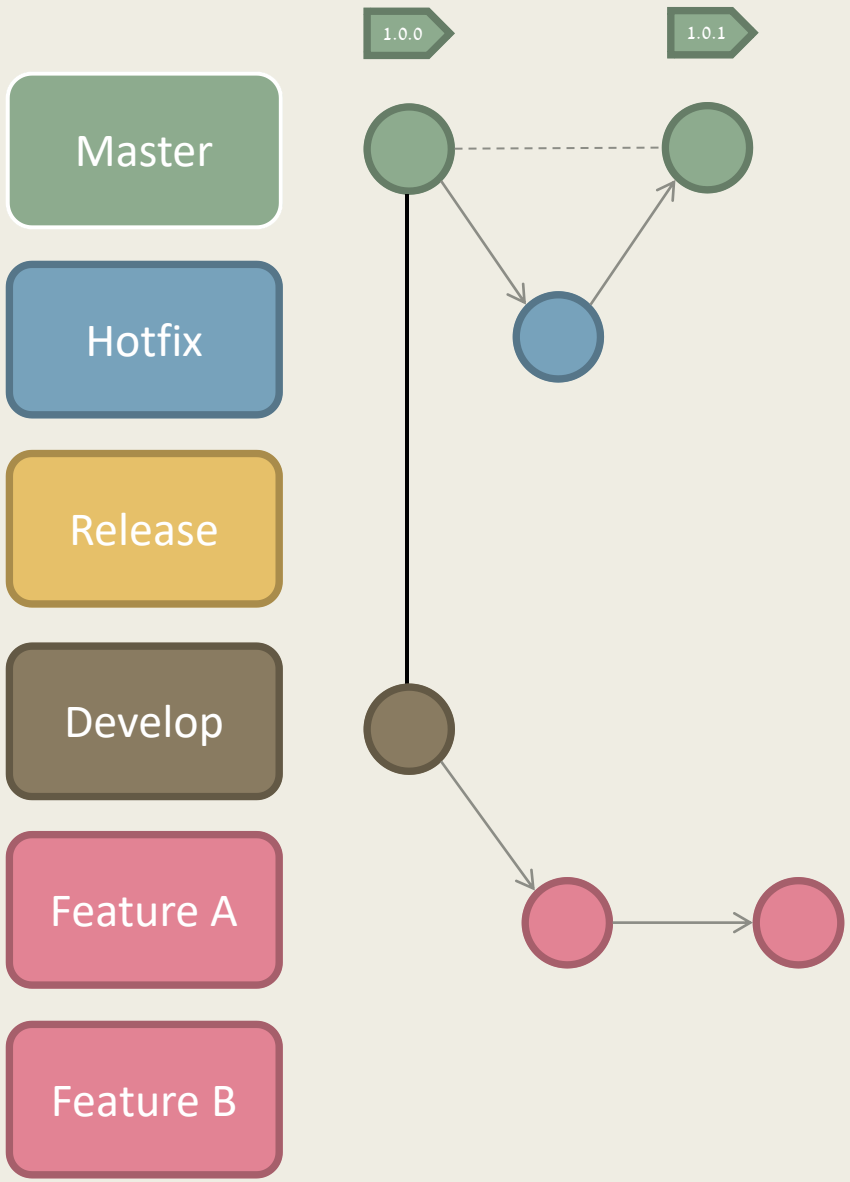

1.0.0
Master
Hotfix
Release
Develop
Feature A
Feature B

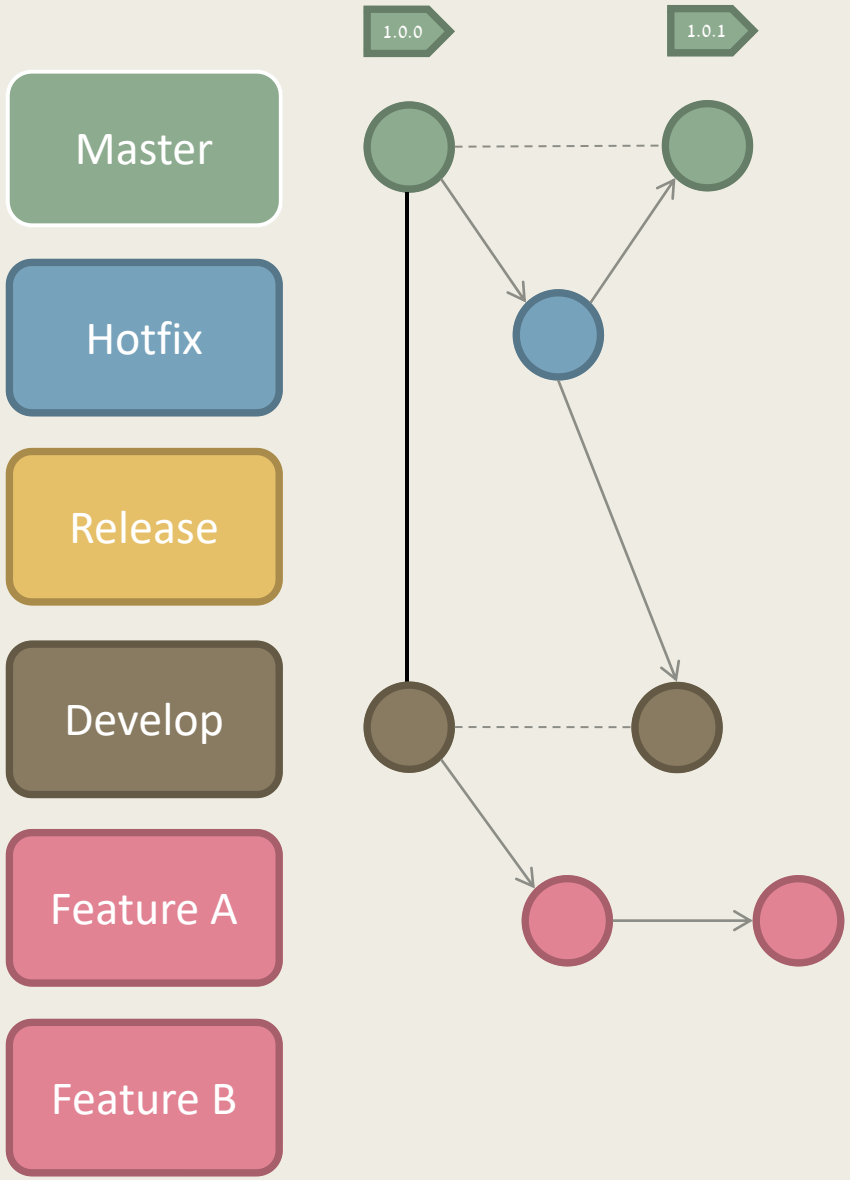

1.0.0
Master
Hotfix
Release
Develop
Feature A
Feature B

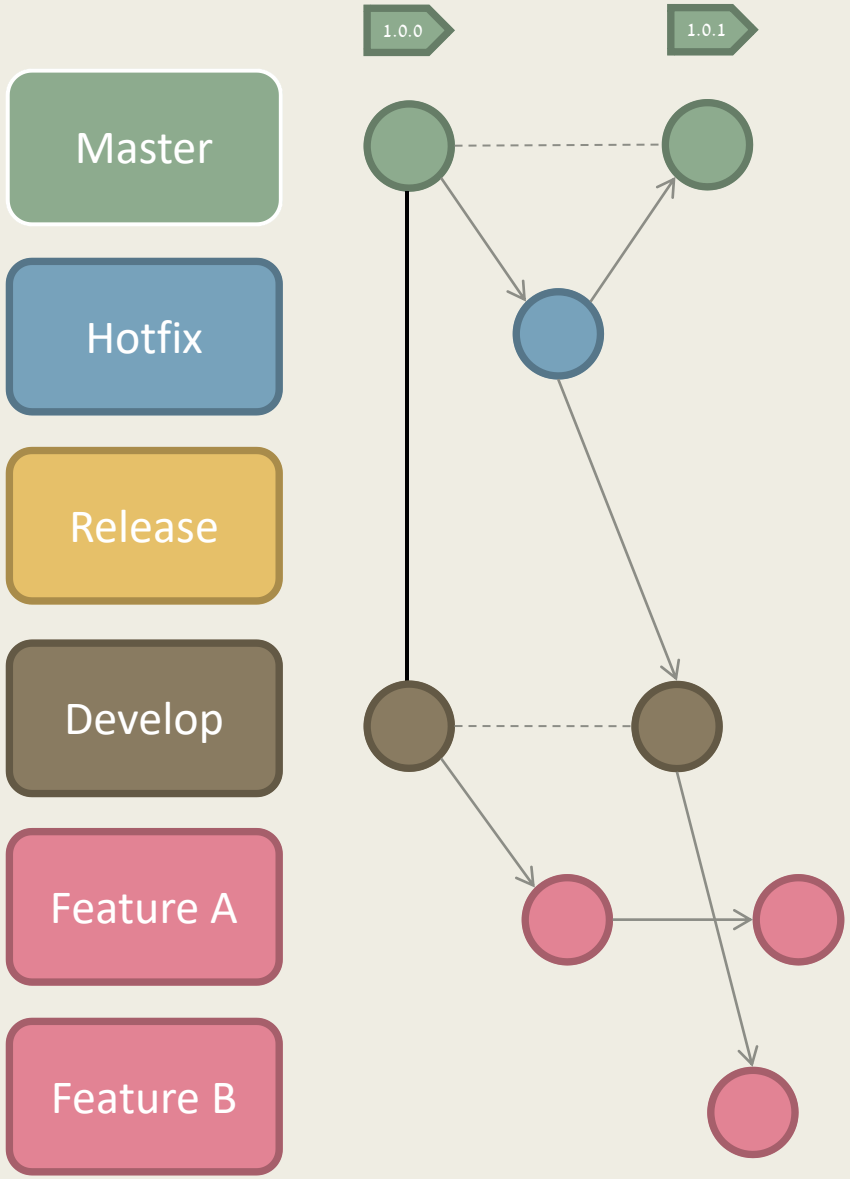

1.0.0
Master
Hotfix
Release
Develop
Feature A
Feature B

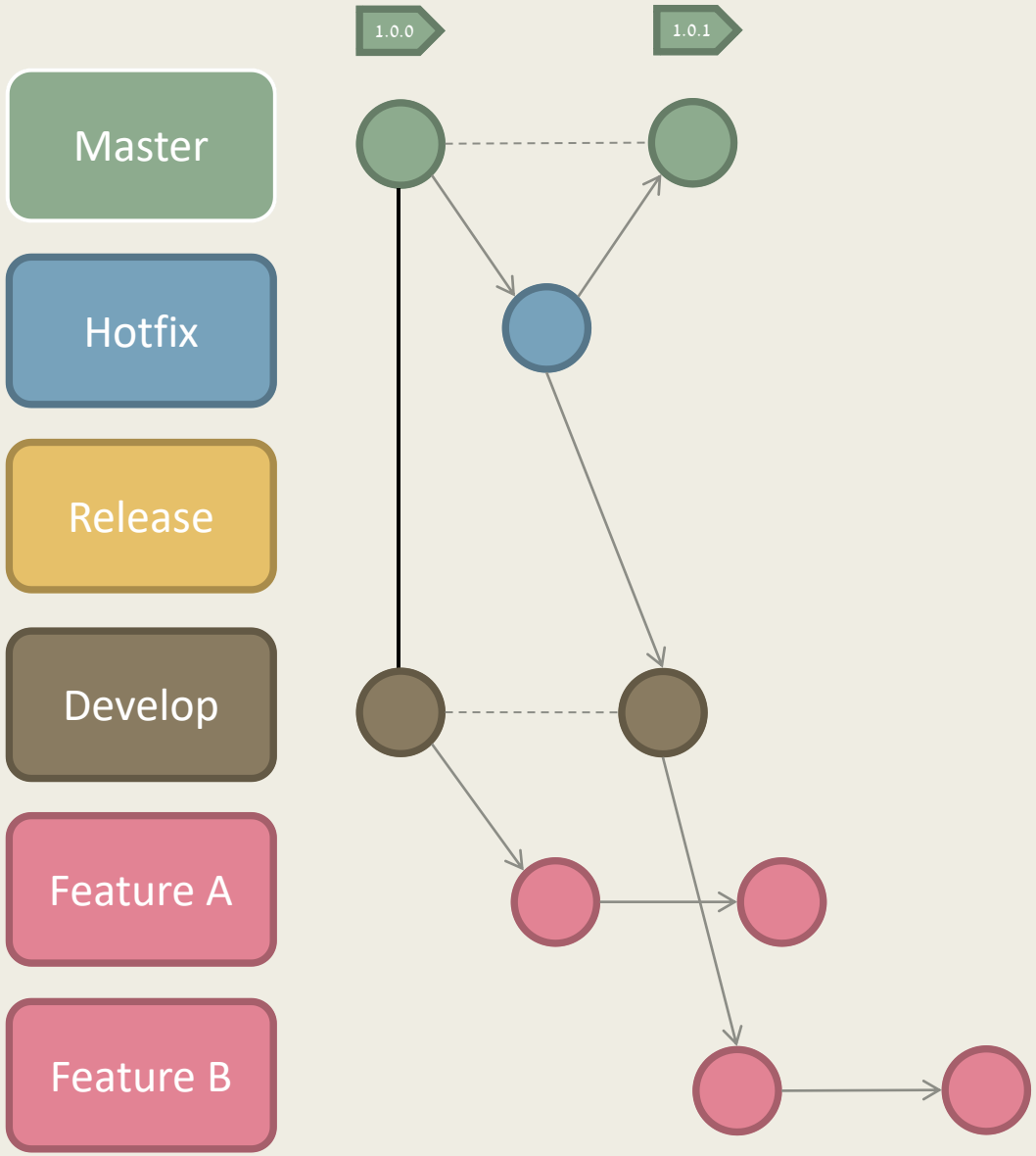

Master
Hotfix
Release
Develop
Feature A
Feature B
1.0.0
OH, UNEXPECTED BUG IN PRODUCTION?

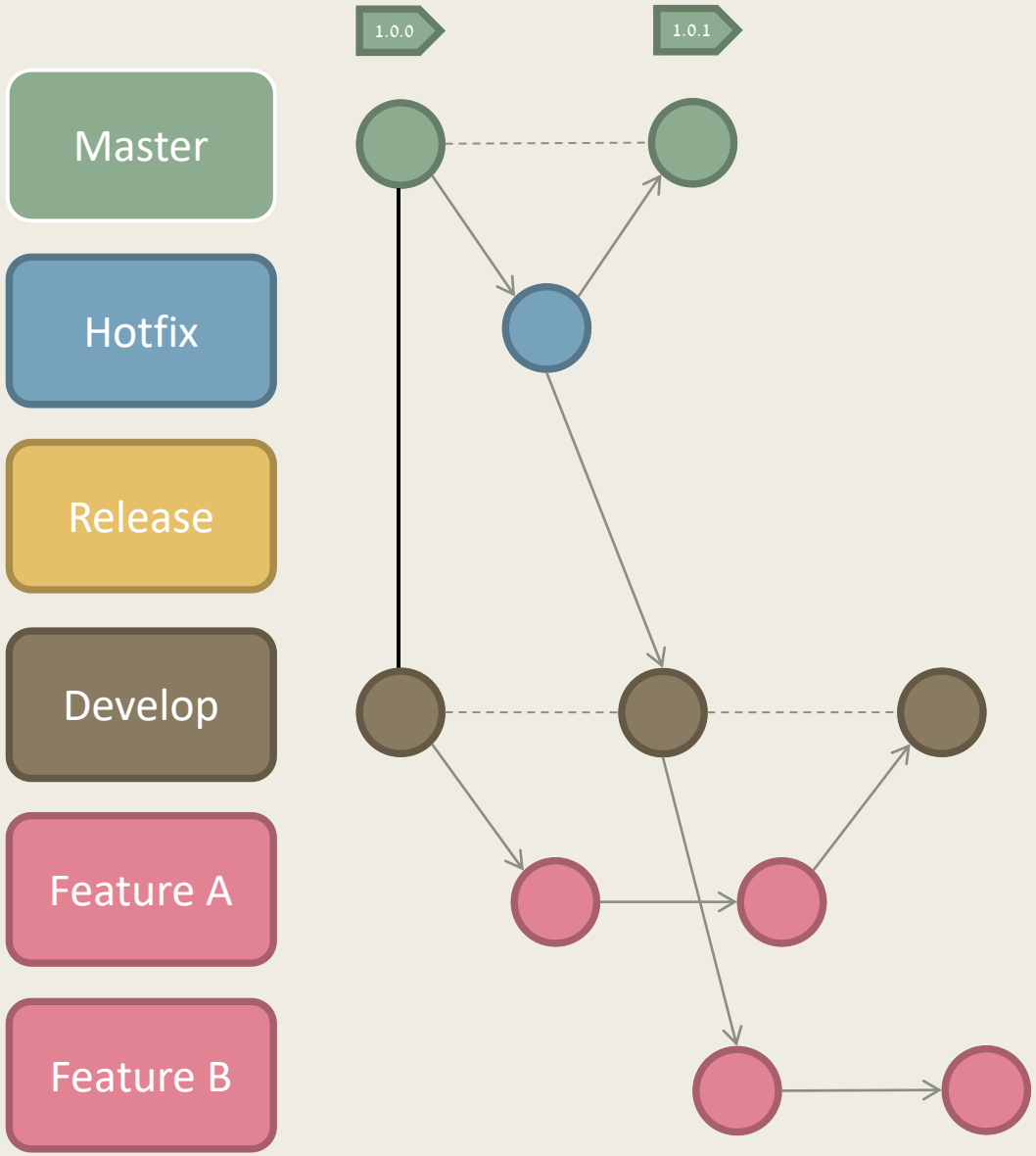

1.0.0
Master
Hotfix
Release
Develop
Feature A
Feature B

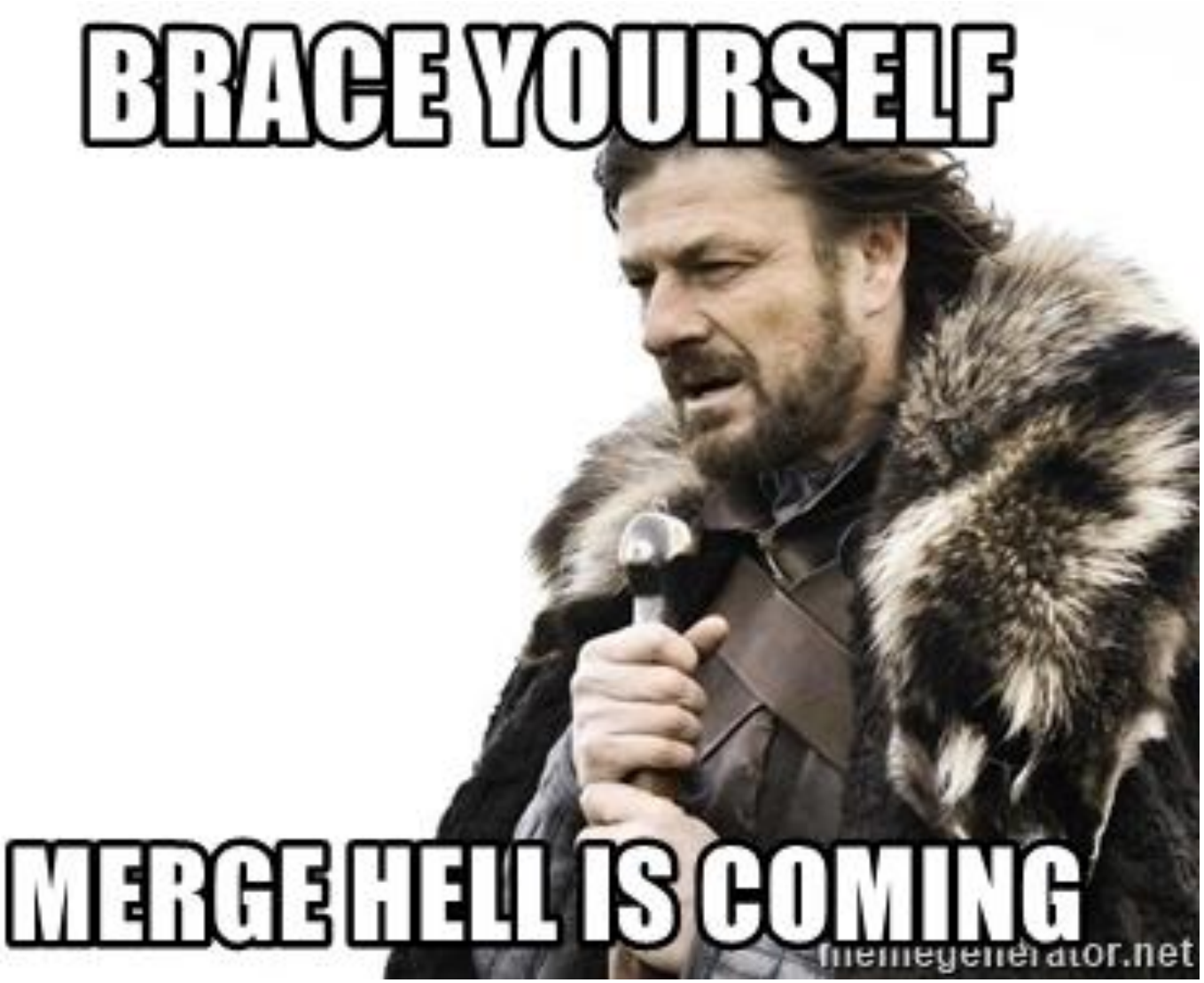

1.0.0
1.0.1
Master
Hotfix
Release
Develop
Feature A
Feature B

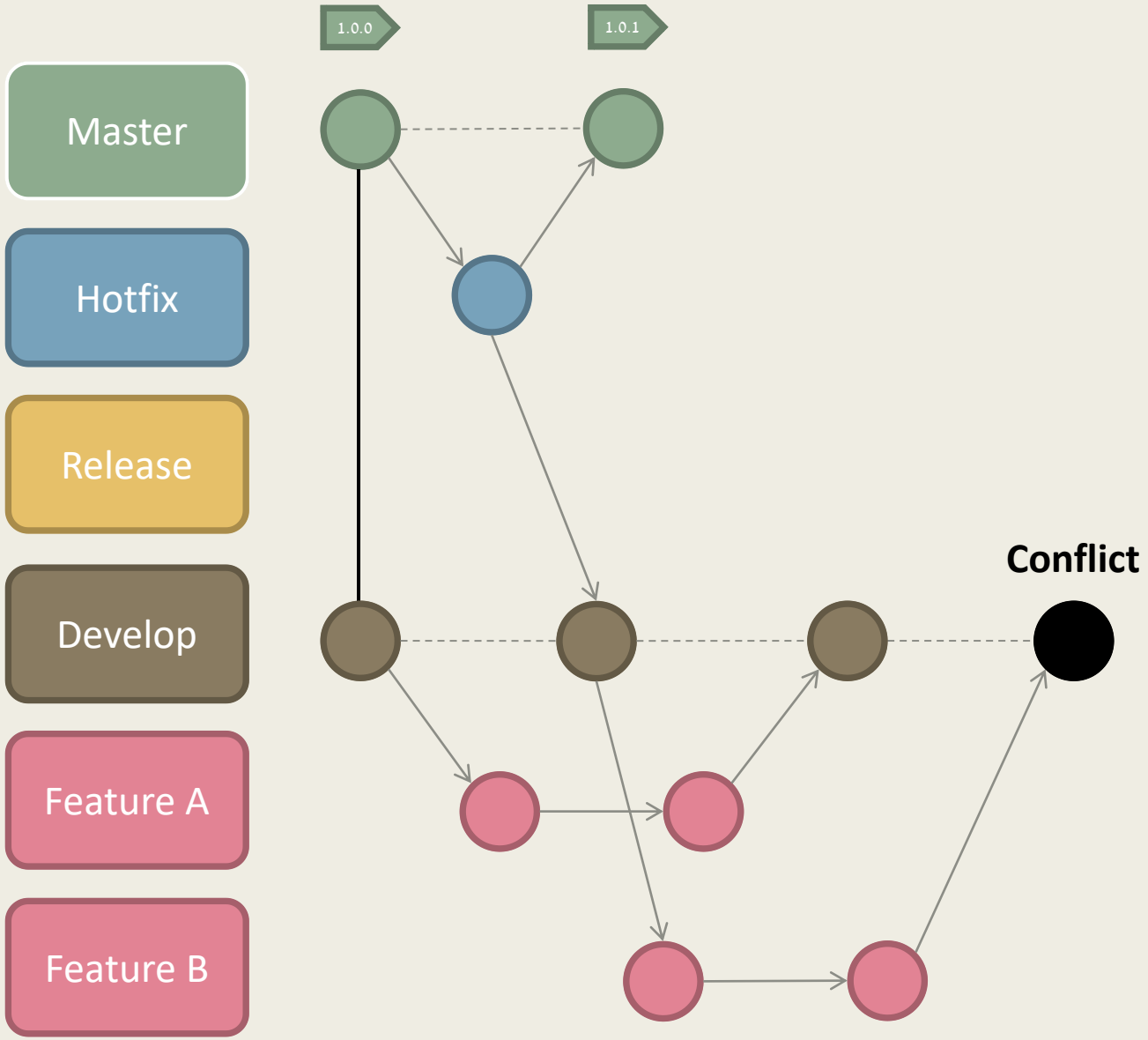

1.0.0
1.0.1
Master
Hotfix
Release
Develop
Feature A
Feature B

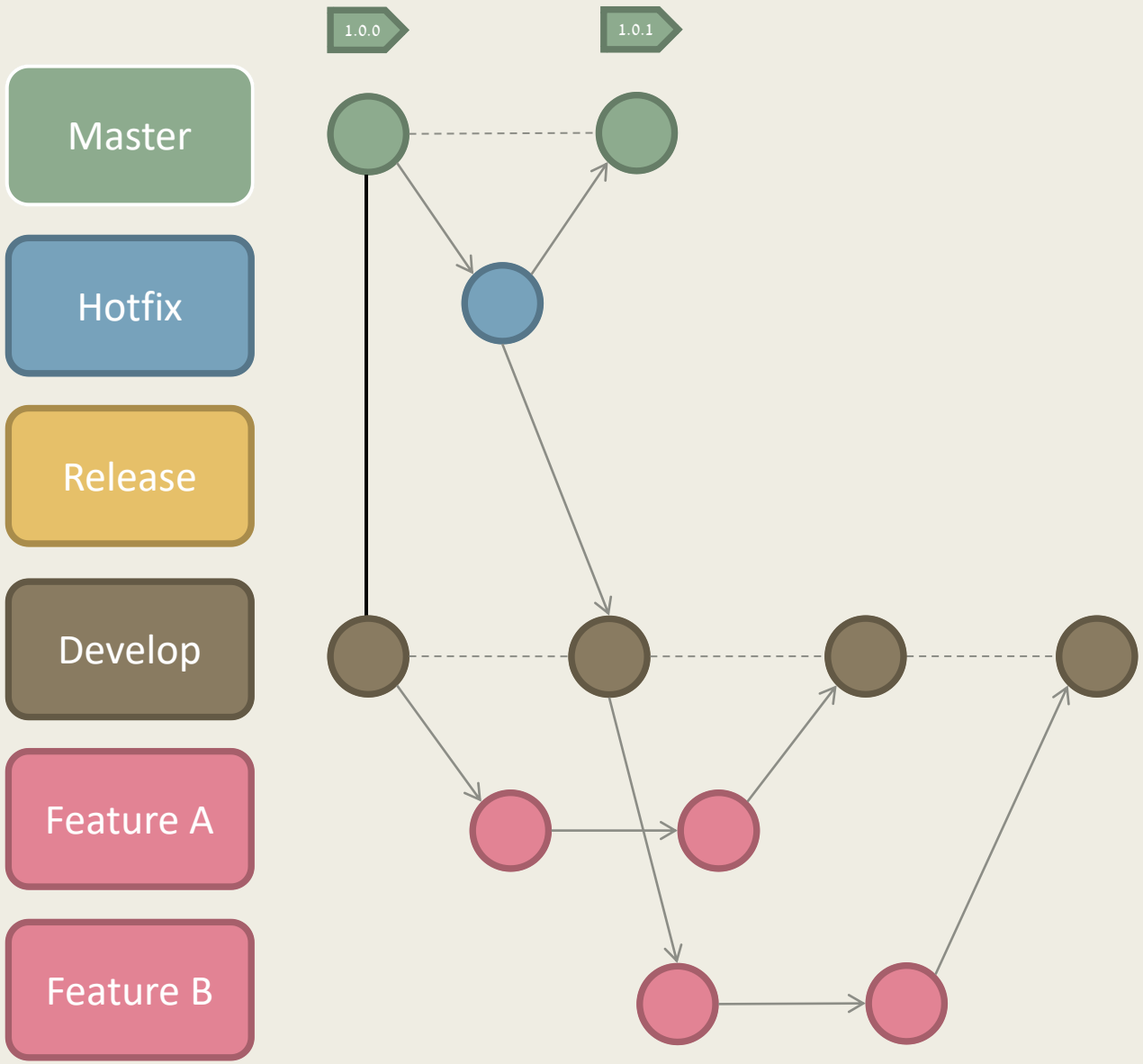

1.0.0
1.0.1
Master
Hotfix
Release
Develop
Feature A
Feature B

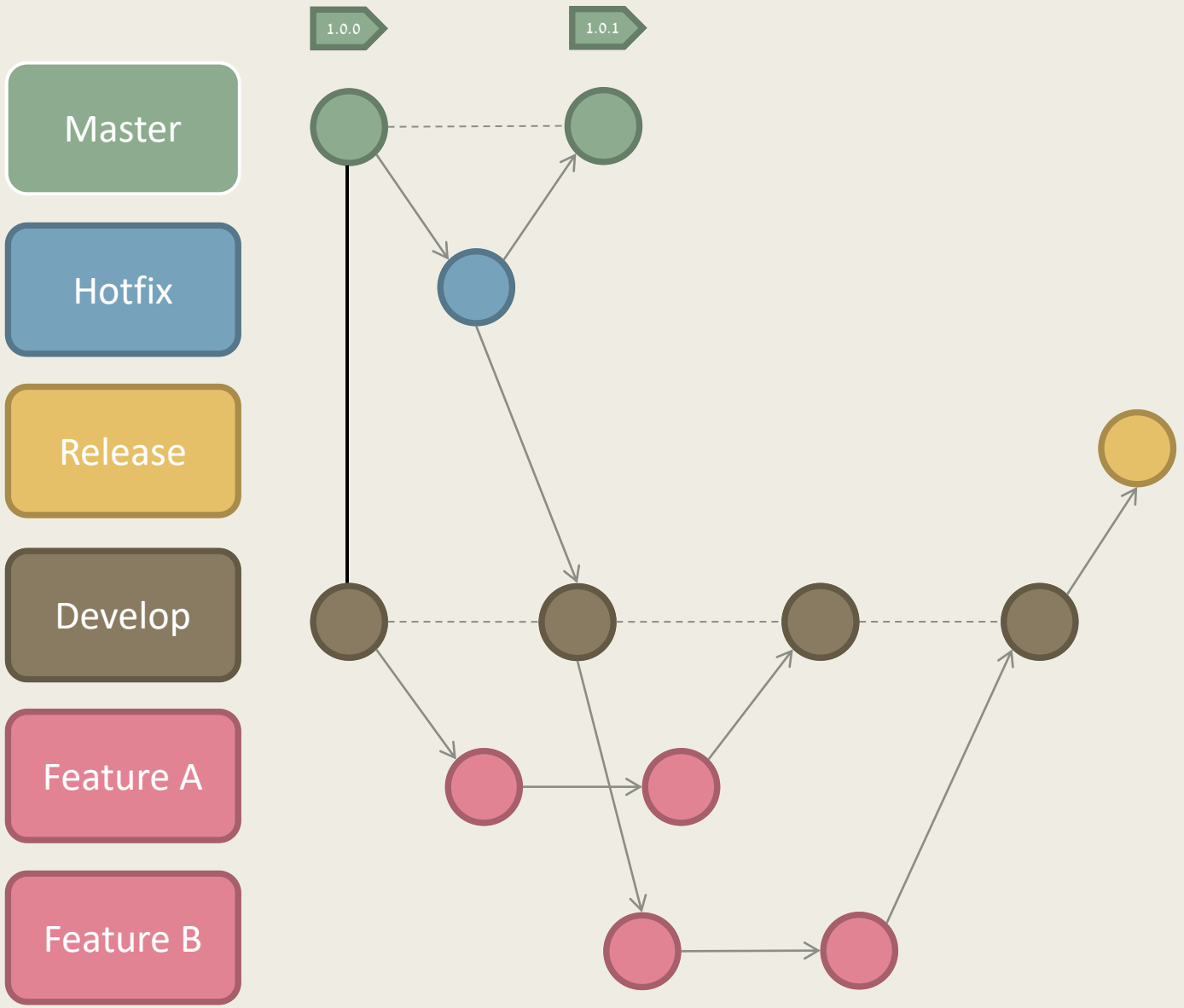

1.0.0
1.0.1
Master
Hotfix
Release
Develop
Feature A
Feature B

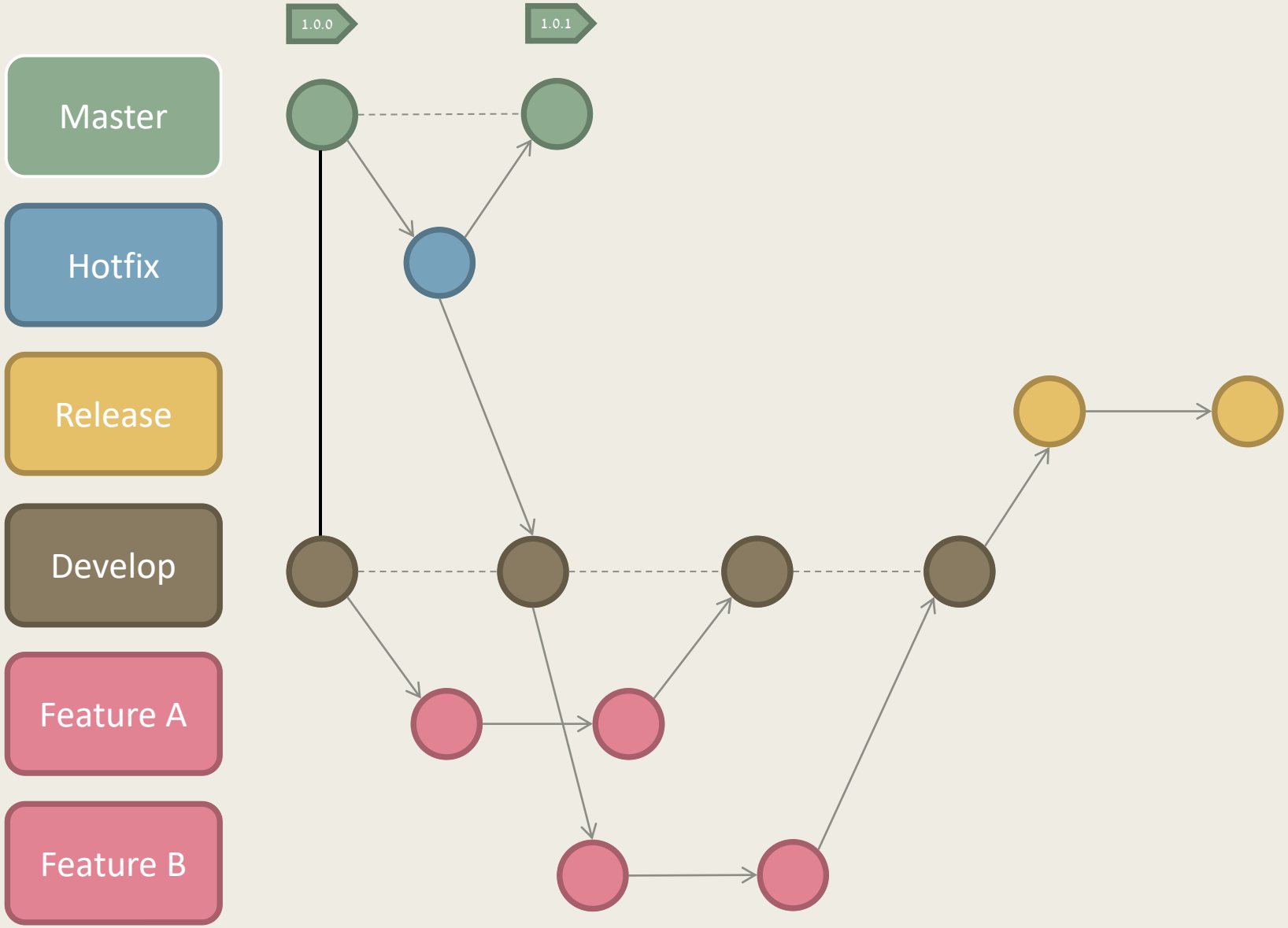

1.0.0
1.0.1
Master
Hotfix
Release
Develop
Feature A
Feature B

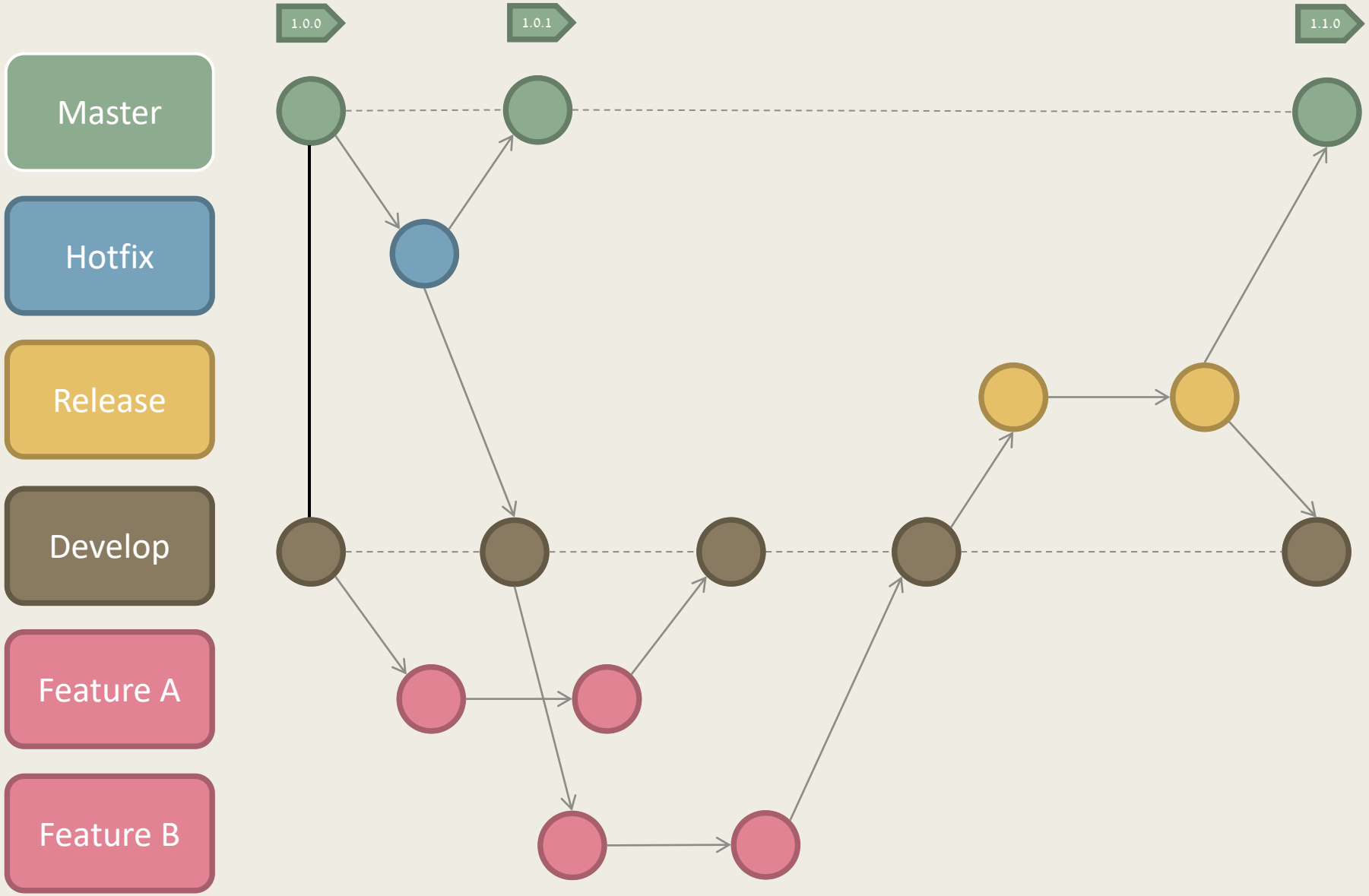

BRACE YOURSELF
MERGE HELL IS COMING


Master
Hotfix
Release
Develop
Feature A
Feature B
1.0.0
1.0.1
Conflict


Master
Hotfix
Release
Develop
Feature A
Feature B
1.0.0
1.0.1


1.0.0
1.0.1
Master
Hotfix
Release
Develop
Feature A
Feature B


1.0.0
1.0.1
Master
Hotfix
Release
Develop
Feature A
Feature B


Master
Hotfix
Release
Develop
Feature A
Feature B
1.0.0
1.0.1
1.1.0

Summary

	Feature Branches	Release Branches	Hotfix Branches
From	develop	develop	Master
To	develop	develop + Master	Master + Develop (or Release)

Summary

Pros

Parallel Development

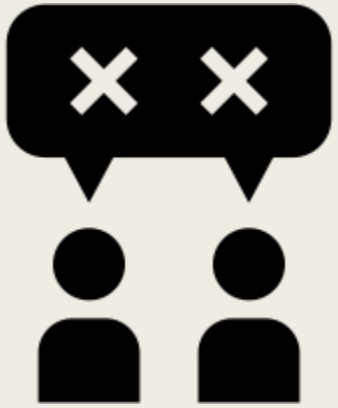
Quickly Fix Production Issues

Different Phases

Summary

Cons

Overhead



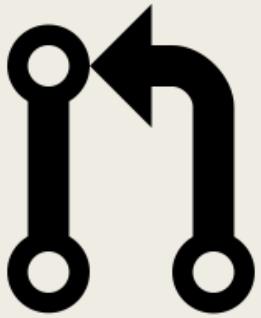
Conflicts

Complex

Some Tips

Branch Protection Rules

Squash



Pull Requests

Release Notes

FB Lifecycle

References

- Git Flow by Vincent Driessen –
  - *https://nvie.com/posts/a-successful-git-branching-model/*
- SemVer –
  - *https://semver.org/*
- Git-flow cli –
  - *https://danielkummer.github.io/git-flow-cheatsheet/*
- MemeGenerator –
  - *https://memegenerator.net/*
- TheNounProject –
  - *https://thenounproject.com/*

Thanks ☺