





Sippakorn Raksakiart
Software Engineer @ **WISESIGHT**



Trunk Based Development



Sippakorn Raksakiart
Software Engineer @ **WISESIGHT**

Agenda

Agenda

.



Agenda

Introduction

- Who am I ?

Source Control Branching Model

- Git flow
- Github flow

Trunk Based Development

- Why Trunk Based
- What's Trunk Based

My Experience about Trunk Based

- Implement TB in real life

Source control branching model

Branching model is a **Risk Management** of software development

Source control branching model

Branching model is a **Risk Management** of software development

Eg. GitFlow, Github flow or Trunk Based

GitFlow

By Vincent Driessen

GitFlow

By Vincent Driessen

GitFlow

By Vincent Driessen



Master / Main / Trunk
Production Ready stage

GitFlow

By Vincent Driessen

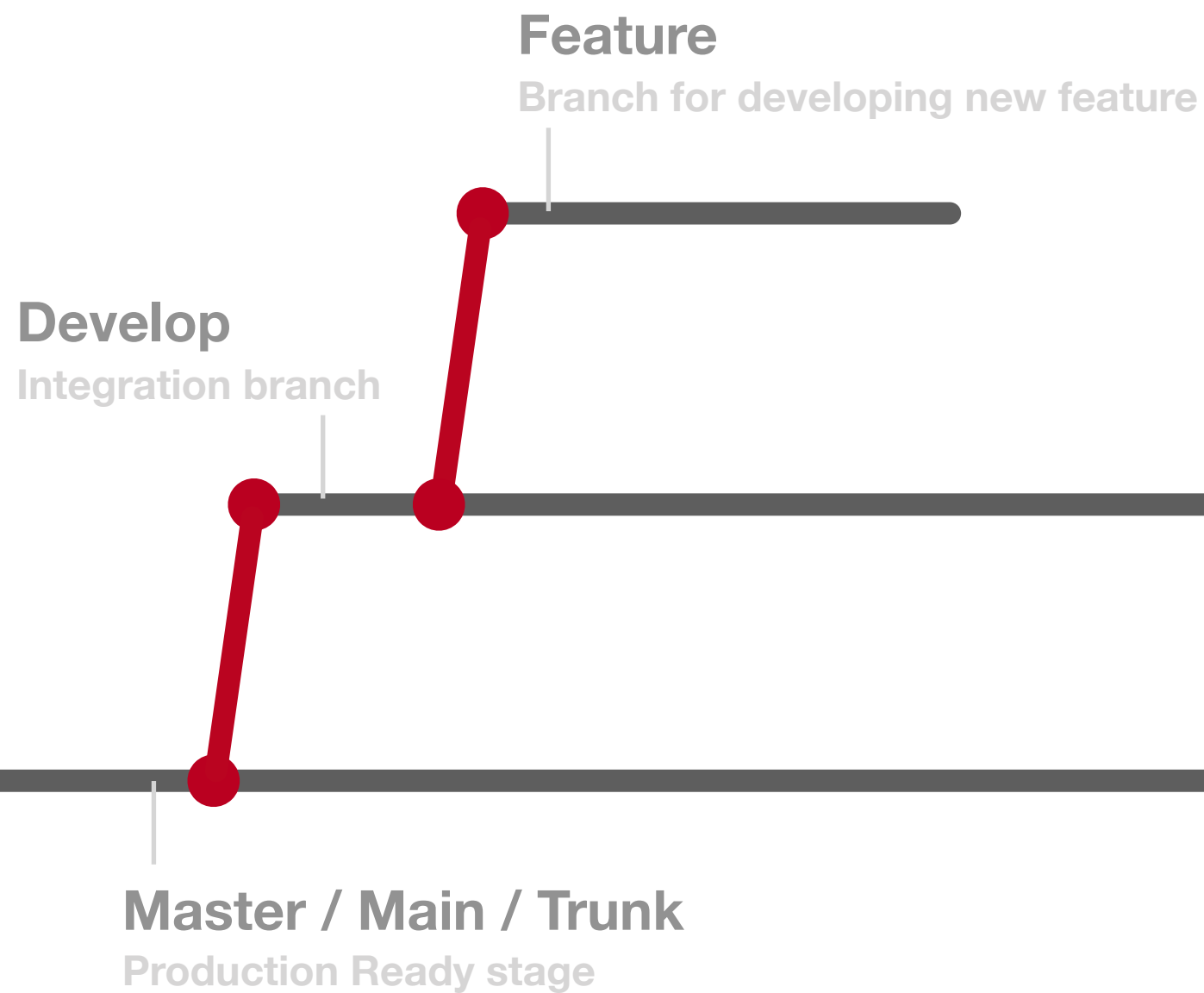
Develop
Integration branch

Master / Main / Trunk
Production Ready stage



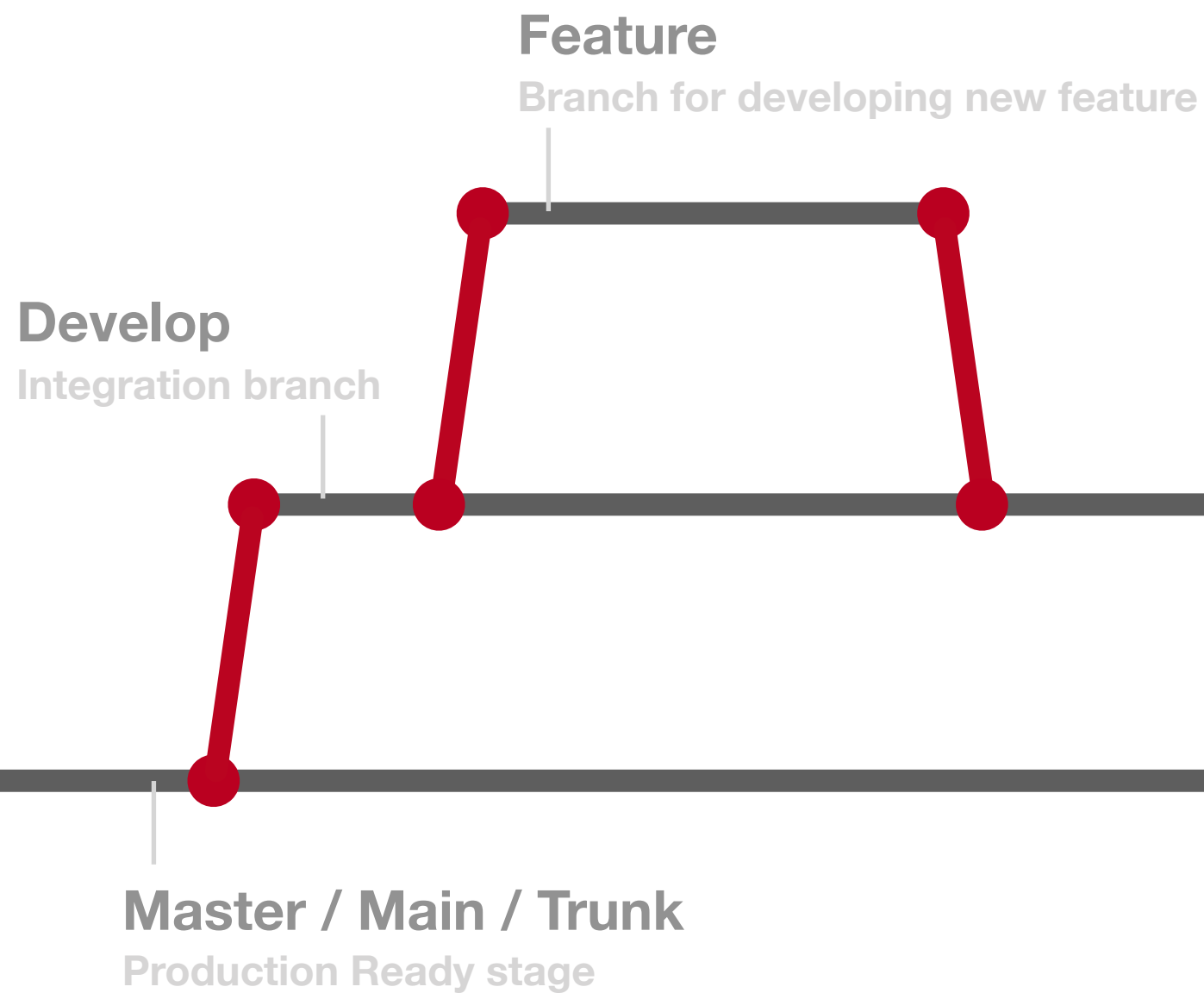
GitFlow

By Vincent Driessen



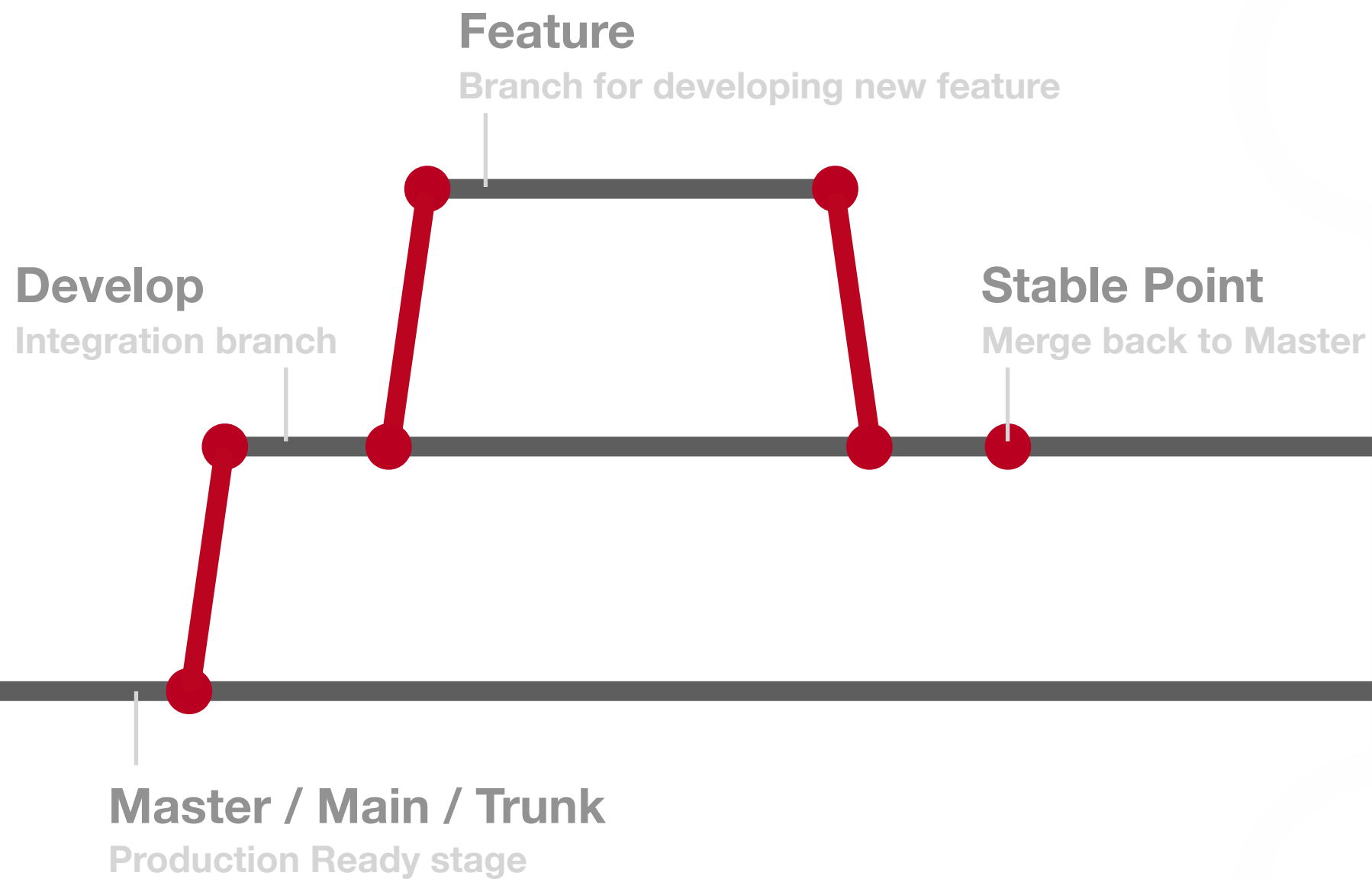
GitFlow

By Vincent Driessen



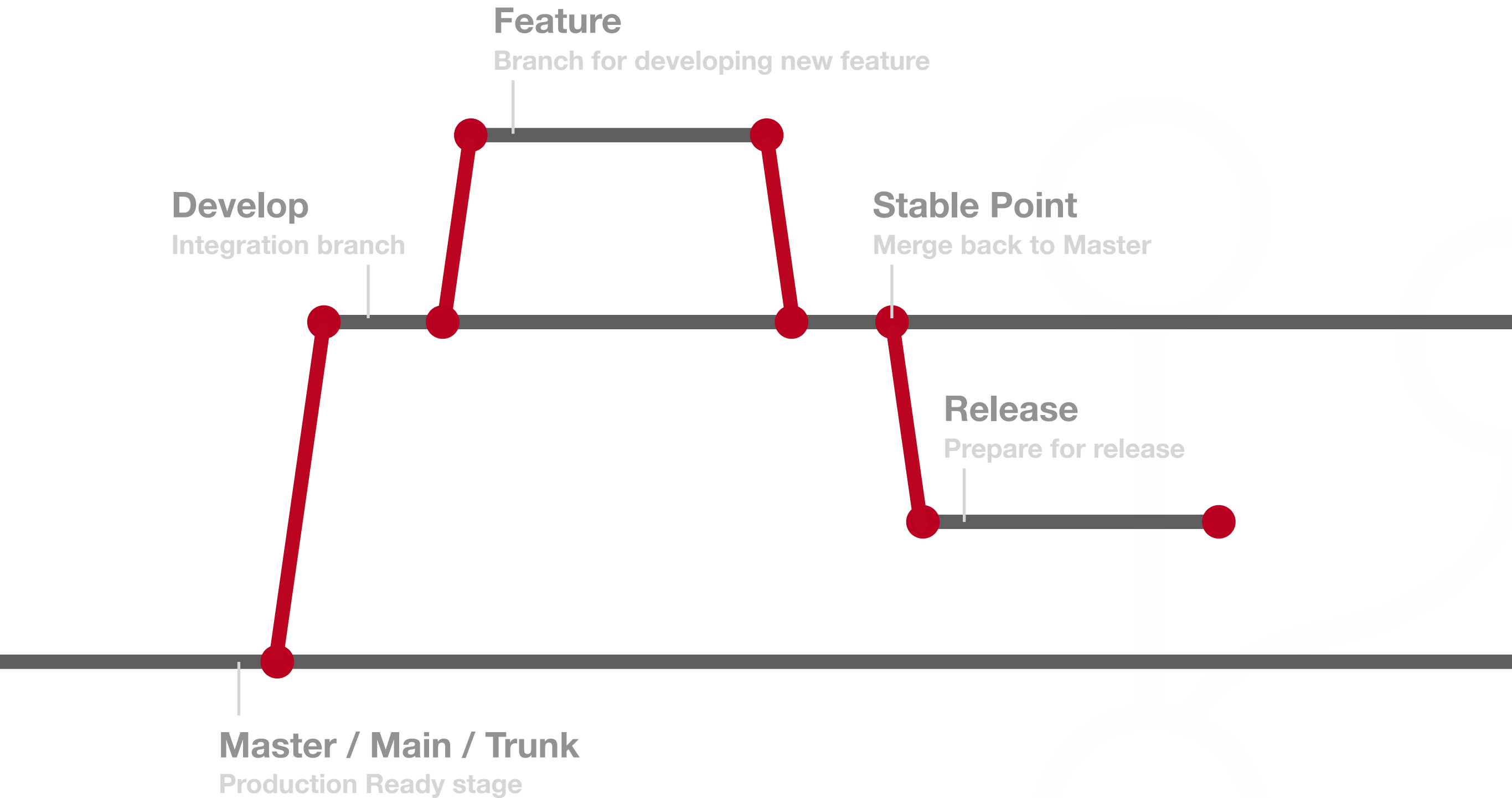
GitFlow

By Vincent Driessen



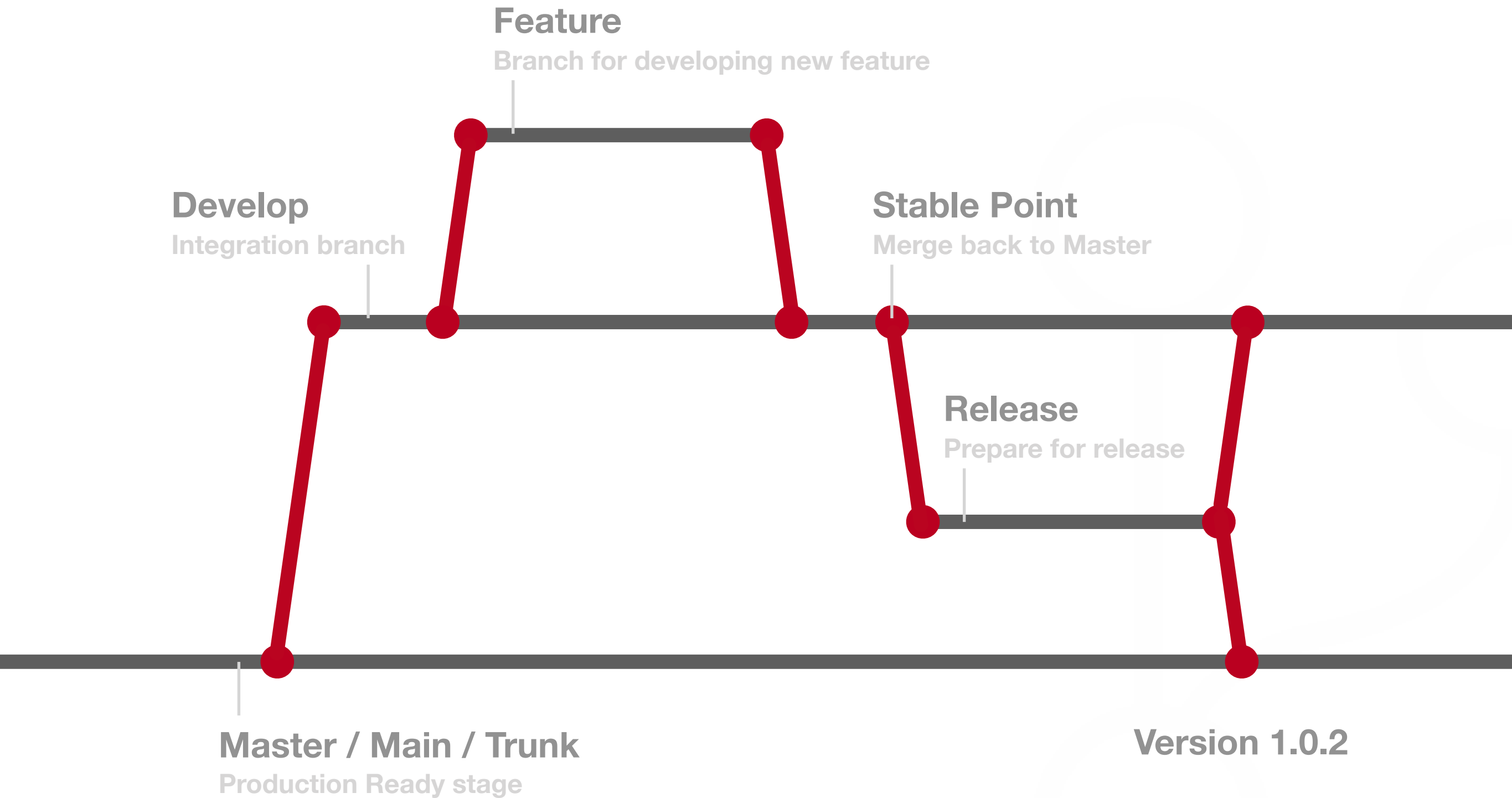
GitFlow

By Vincent Driessen



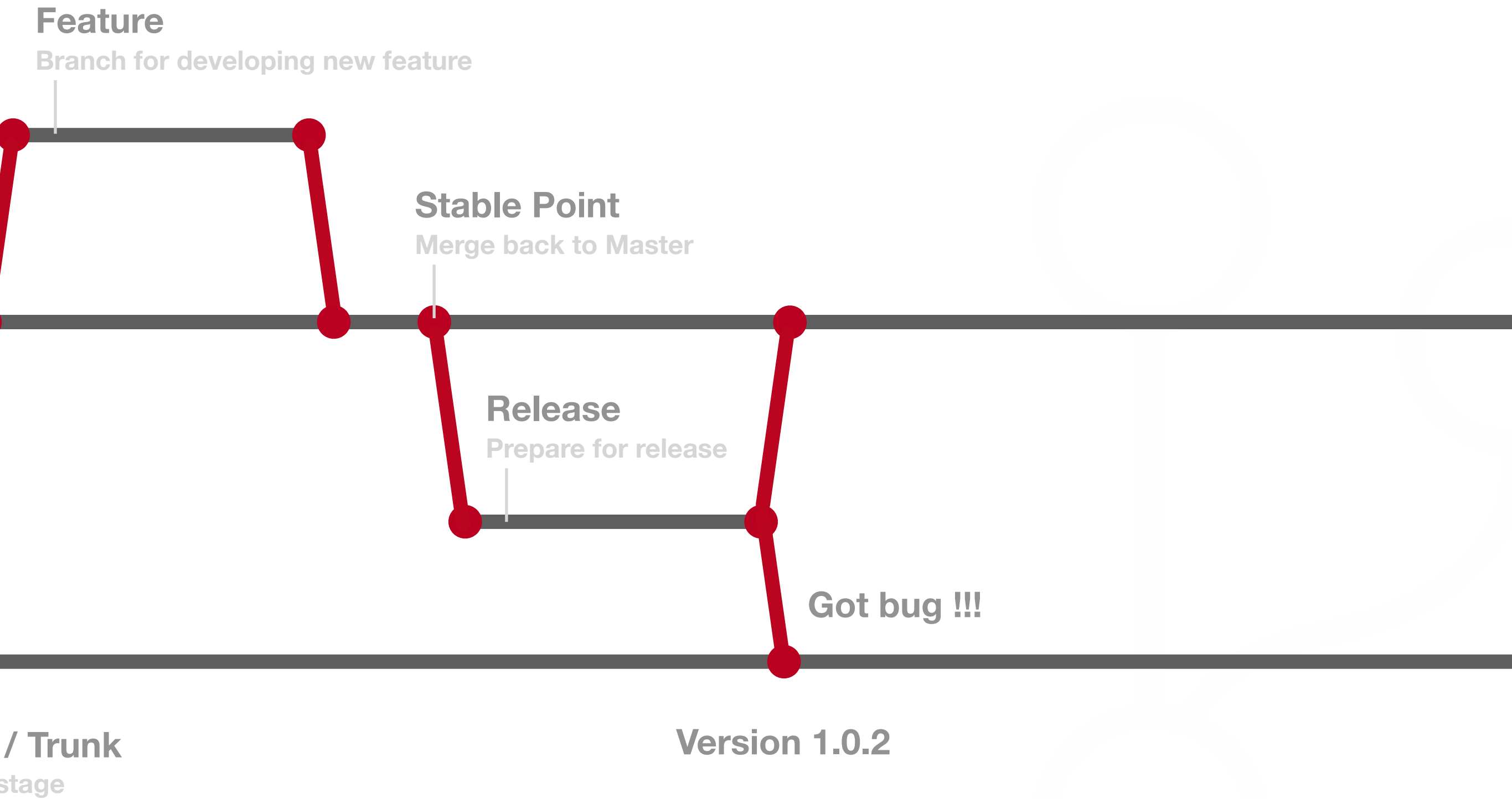
GitFlow

By Vincent Driessen



GitFlow

By Vincent Driessen



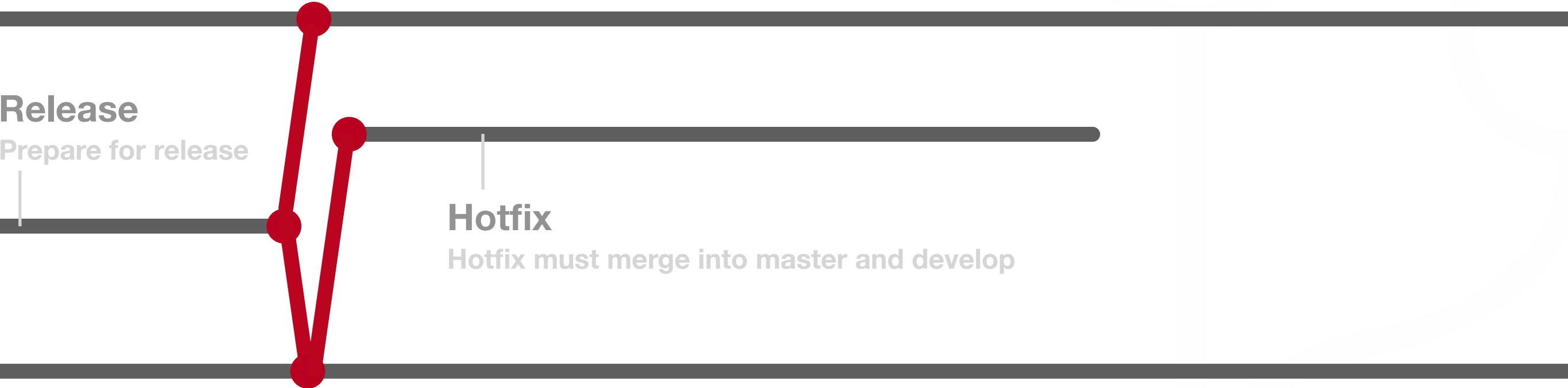
Point
back to Master

Release
Prepare for release

Hotfix
Hotfix must merge into master and develop

Version 1.0.2

Version 1.0.3



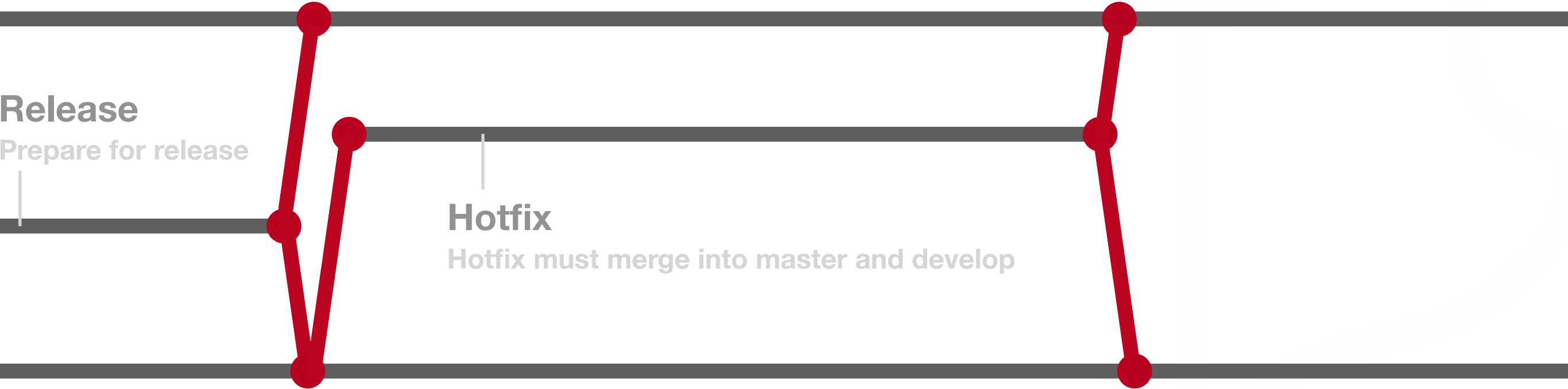
Point
back to Master

Release
Prepare for release

Hotfix
Hotfix must merge into master and develop

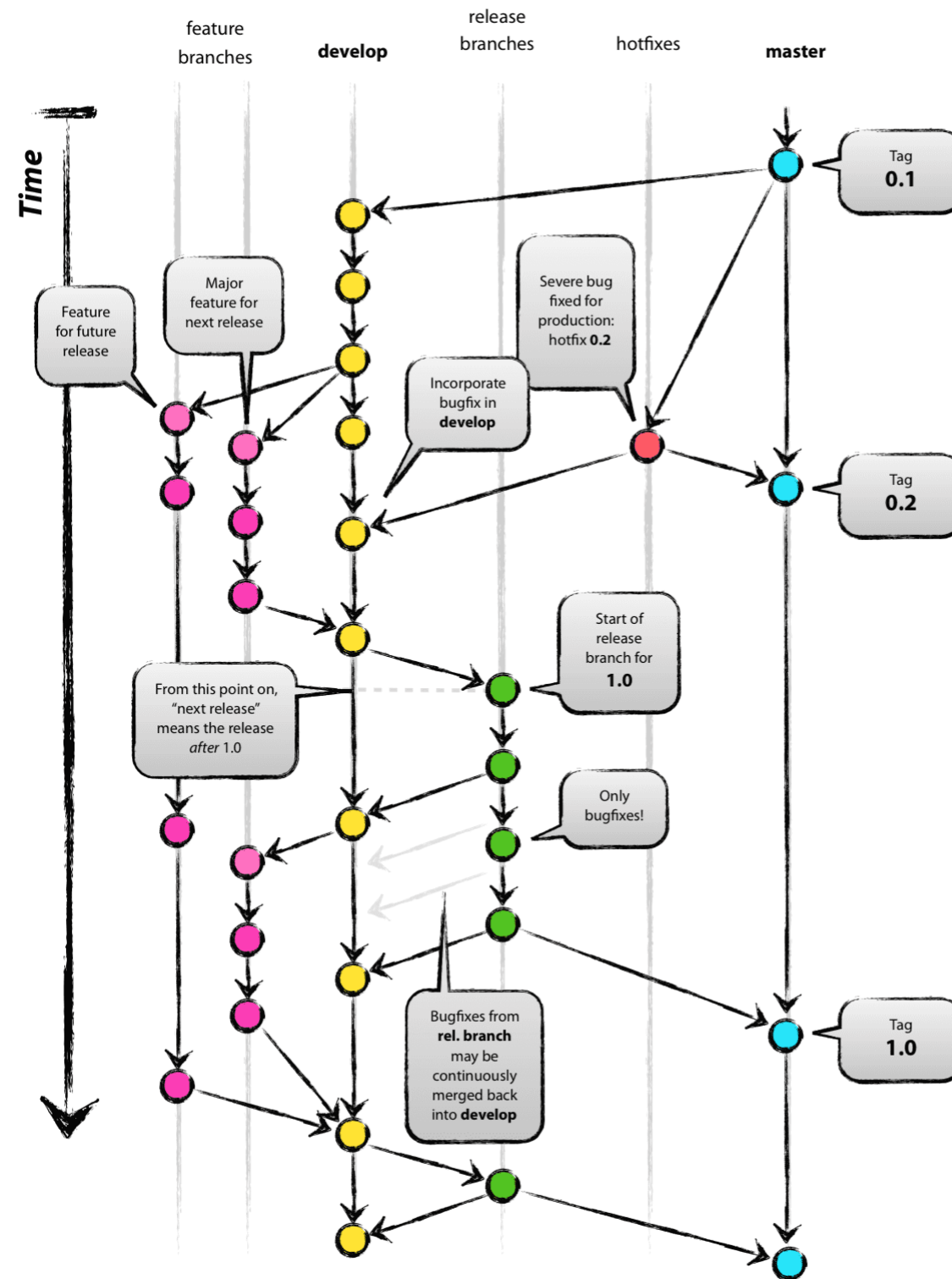
Version 1.0.2

Version 1.0.3



GitFlow

By Vincent Driessen



GitHub Flow

By Github :P

GitHub Flow

By Github :P



Master / Main / Trunk

Production Ready stage

GitHub Flow

By Github :P

Supporting Branch

Develop own feature

Master / Main / Trunk

Production Ready stage



GitHub Flow

By Github :P



spksoft commented 5 days ago • edited ▾



Implement elastic query builder of refactoring match-worker

This PR will add *elasticsearch update and upsert query builder*, *starter command* and call pre-process ze-matcher

- ✓ Add GenerateChannelIndexName
- ✓ Add GenerateCampaignIndexName
- ✓ Add BuildBulkUpdateIndexQuery
- ✓ Add get config into `main` pkg
- ✓ Add get command param into `main` pkg
- ✓ Polish some code
- ✓ Add comment to each step inside main
- ✓ Add preprocessZocialEyeMessage



clozed2u requested changes 2 days ago

[View changes](#)

match-worker/cmd/main.go | Outdated | Hide resolved

```
207 + <-exitSignalChannel
208 + for processingJobs > 0 {
209 + }
210 + runtime.Goexit()
```



clozed2u 2 days ago



return should work fine with this case.



Reply...

Unresolve conversation

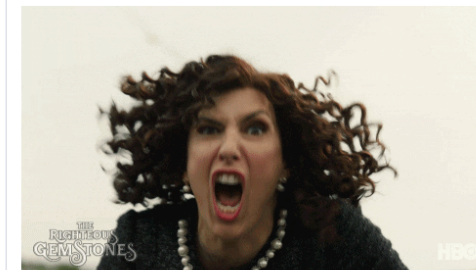
spksoft marked this conversation as resolved.



protal approved these changes yesterday

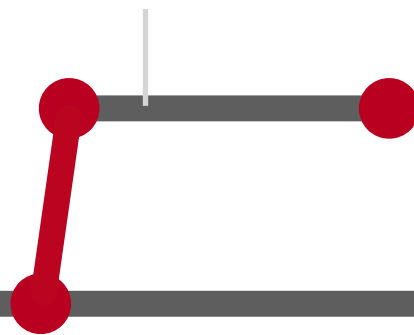
[View changes](#)

protal left a comment



Supporting Branch

Develop own feature



Master / Main / Trunk

Production Ready stage

GitHub Flow

By Github :P



spksoft commented 5 days ago • edited ▾

Implement elastic query builder of refactoring match-worker

This PR will add *elasticsearch update and upsert query builder*, *starter command* and call pre-process ze-matcher

- ✓ Add GenerateChannelIndexName
- ✓ Add GenerateCampaignIndexName
- ✓ Add BuildBulkUpdateIndexQuery
- ✓ Add get config into `main` pkg
- ✓ Add get command param into `main` pkg
- ✓ Polish some code
- ✓ Add comment to each step inside main
- ✓ Add preprocessZocialEyeMessage



clozed2u requested changes 2 days ago

[View changes](#)

match-worker/cmd/main.go | Outdated | Hide resolved

```
207 + <-exitSignalChannel
208 + for processingJobs > 0 {
209 + }
210 + runtime.Goexit()
```



clozed2u 2 days ago

return should work fine with this case.

+ 😊 ...



Reply...

[Unresolve conversation](#)

spksoft marked this conversation as resolved.

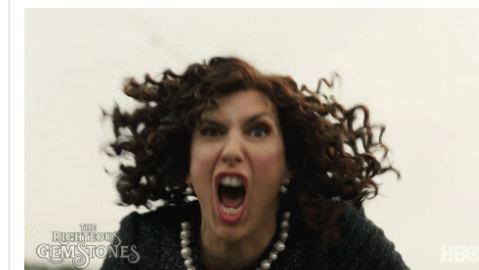


protal approved these changes yesterday

[View changes](#)

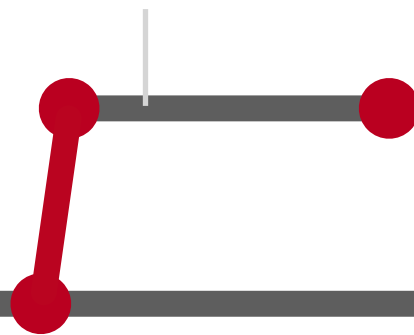
protal left a comment

+ 😊 ...



Supporting Branch

Develop own feature



Master / Main / Trunk

Production Ready stage

GitHub Flow

By Github :P



spksoft commented 5 days ago • edited ▾

Implement elastic query builder of refactoring match-worker

This PR will add *elasticsearch update and upsert query builder*, *starter command* and call pre-process ze-matcher

- ✓ Add GenerateChannelIndexName
- ✓ Add GenerateCampaignIndexName
- ✓ Add BuildBulkUpdateIndexQuery
- ✓ Add get config into `main` pkg
- ✓ Add get command param into `main` pkg
- ✓ Polish some code
- ✓ Add comment to each step inside main
- ✓ Add preprocessZocialEyeMessage



clozed2u requested changes 2 days ago

[View changes](#)

match-worker/cmd/main.go | Outdated | Hide resolved

```
207 + <-exitSignalChannel
208 + for processingJobs > 0 {
209 + }
210 + runtime.Goexit()
```



clozed2u 2 days ago

return should work fine with this case.

+ 😊 ...



Reply...

[Unresolve conversation](#)

spksoft marked this conversation as resolved.

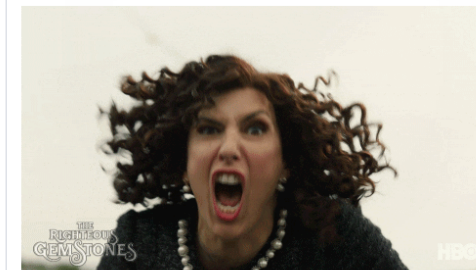


protal approved these changes yesterday

[View changes](#)

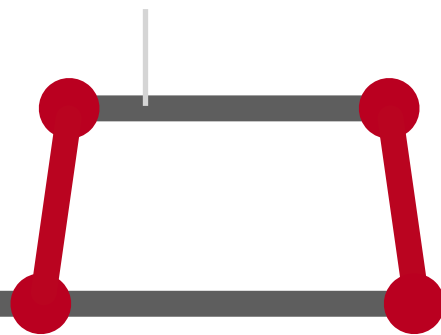
protal left a comment

+ 😊 ...



Supporting Branch

Develop own feature

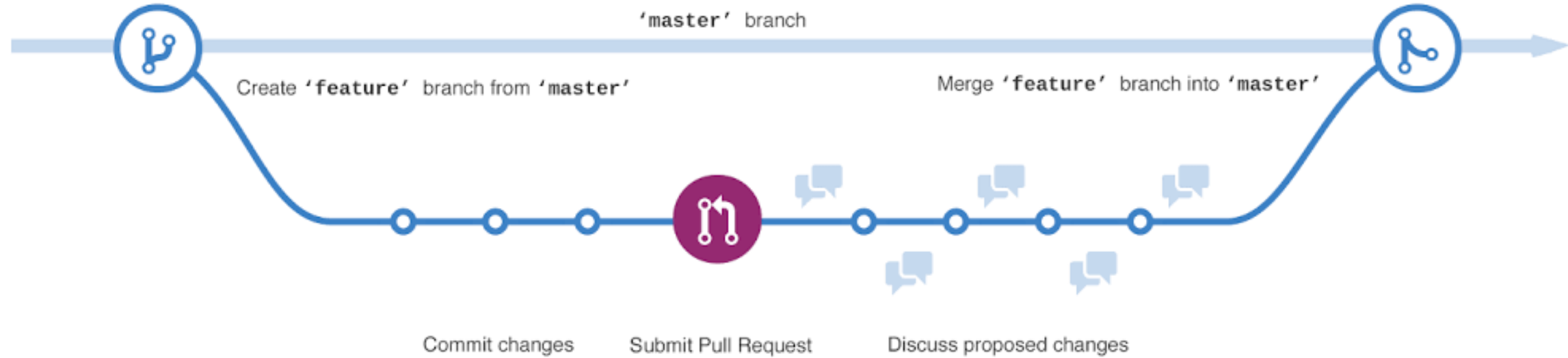


Master / Main / Trunk

Production Ready stage

GitHub Flow

By Github :P



Trunk Based Development

Trunk Based Development

Why ?

“ Branches create distance between developers and
we do not want that ”

— Frank Compagner, Guerrilla Games

Trunk Based Development

Why ?

Frank's 'distance' is about the distance to the integration of code from multiple components/modules/sub-teams

Trunk Based Development

Why ?

Break something unexpected once merged

Trunk Based Development

Why ?

Break something unexpected once merged

Be difficult to merge in

Trunk Based Development

Why ?

Break something unexpected once merged

Be difficult to merge in

Not show that work was duplicated until it is merged

Not show problems of incompatibility/undesirability that does not break the build


Trunk Based Development

Why ?

Trunk-Based Development is a branching model that reduces this distance to the minimum

Trunk Based Development

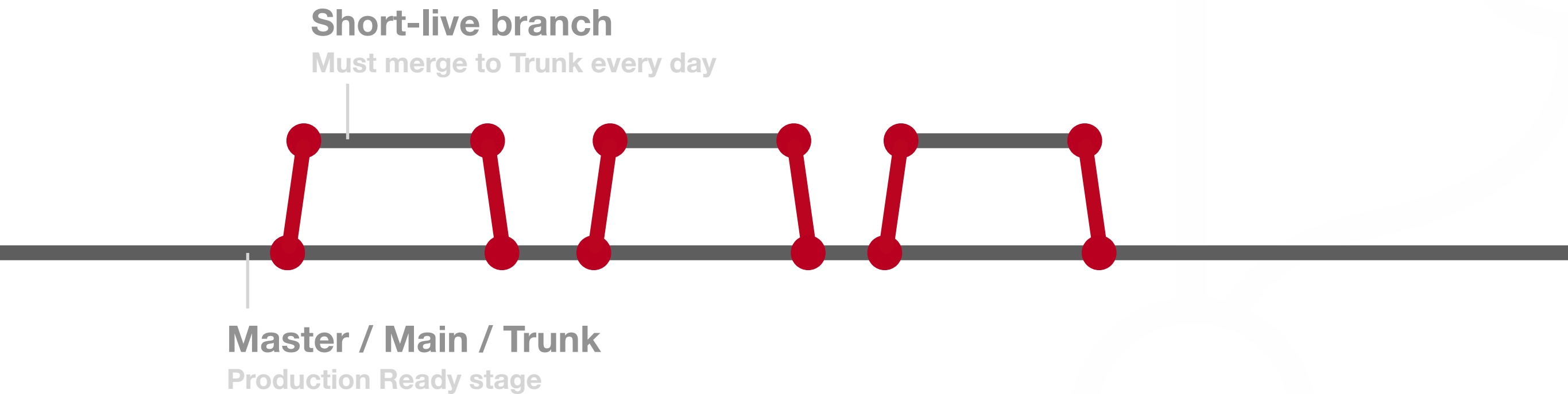
What ?



Master / Main / Trunk
Production Ready stage

Trunk Based Development

What ?

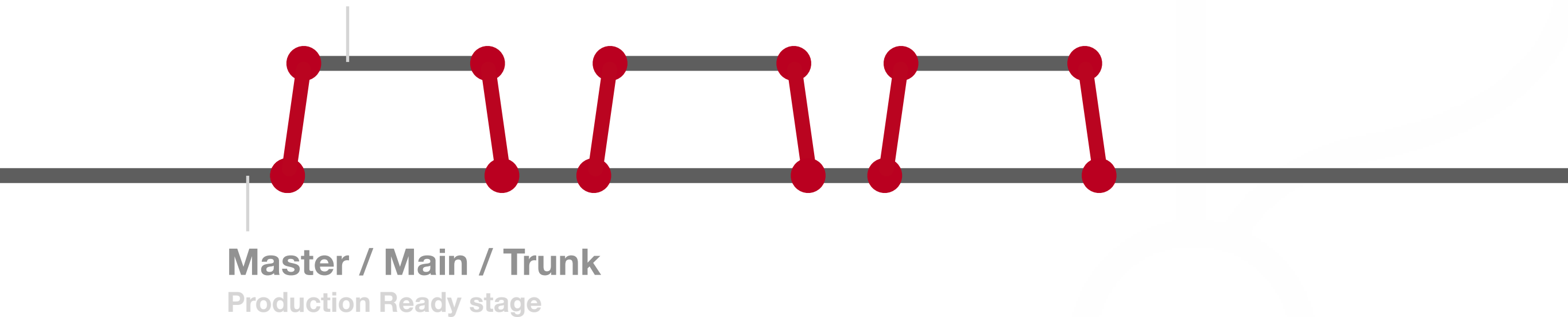


Trunk Based Development

What ?

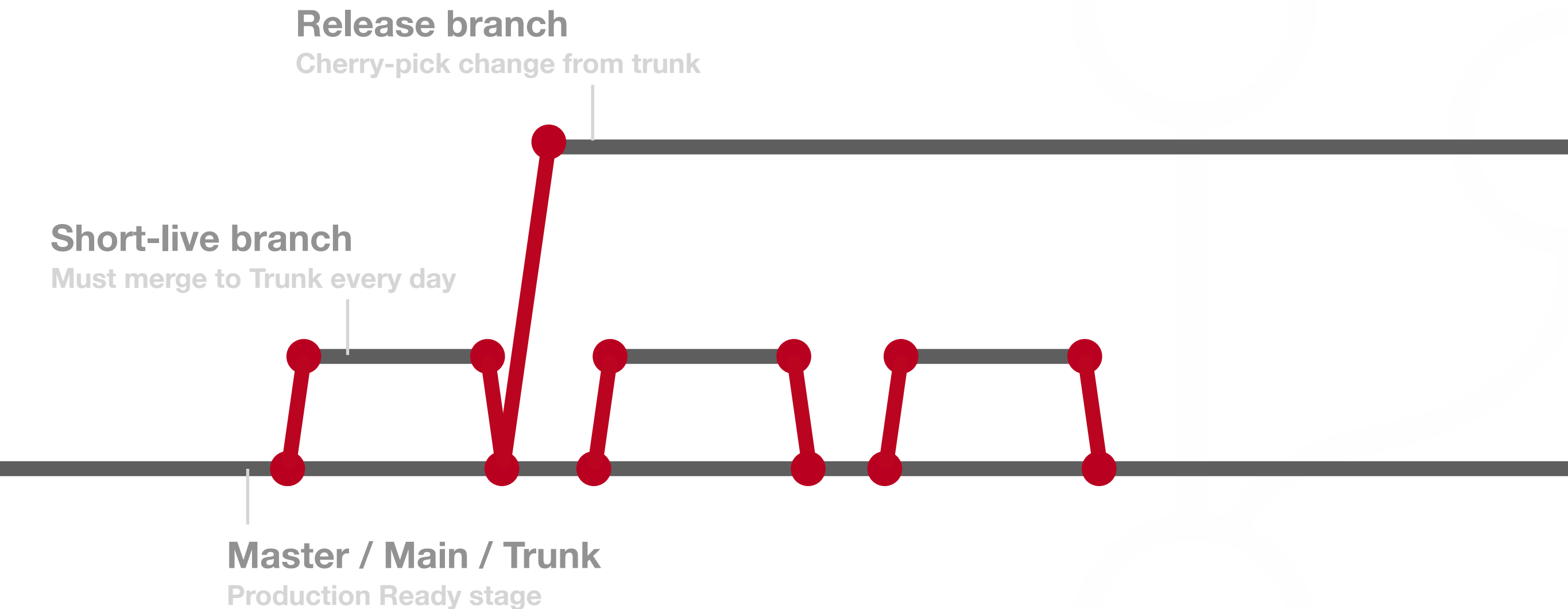
Short-live branch

Must merge to Trunk every day



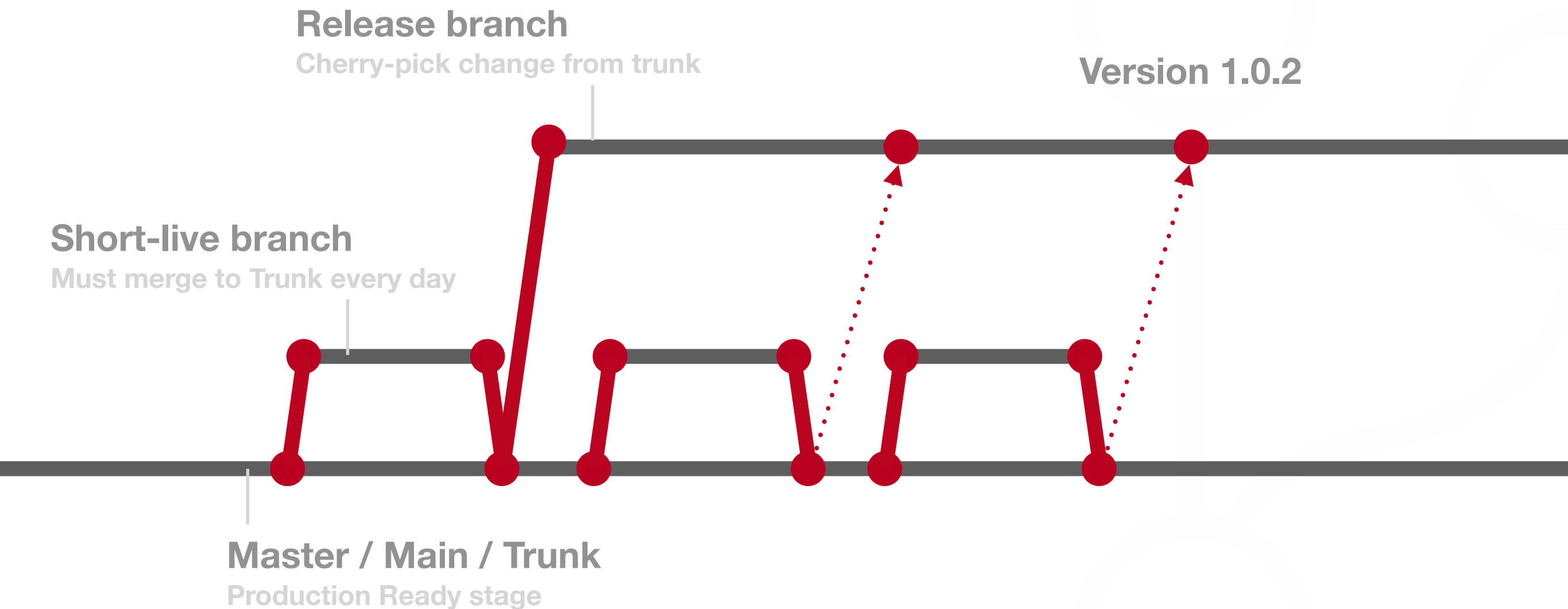
Trunk Based Development

What ?



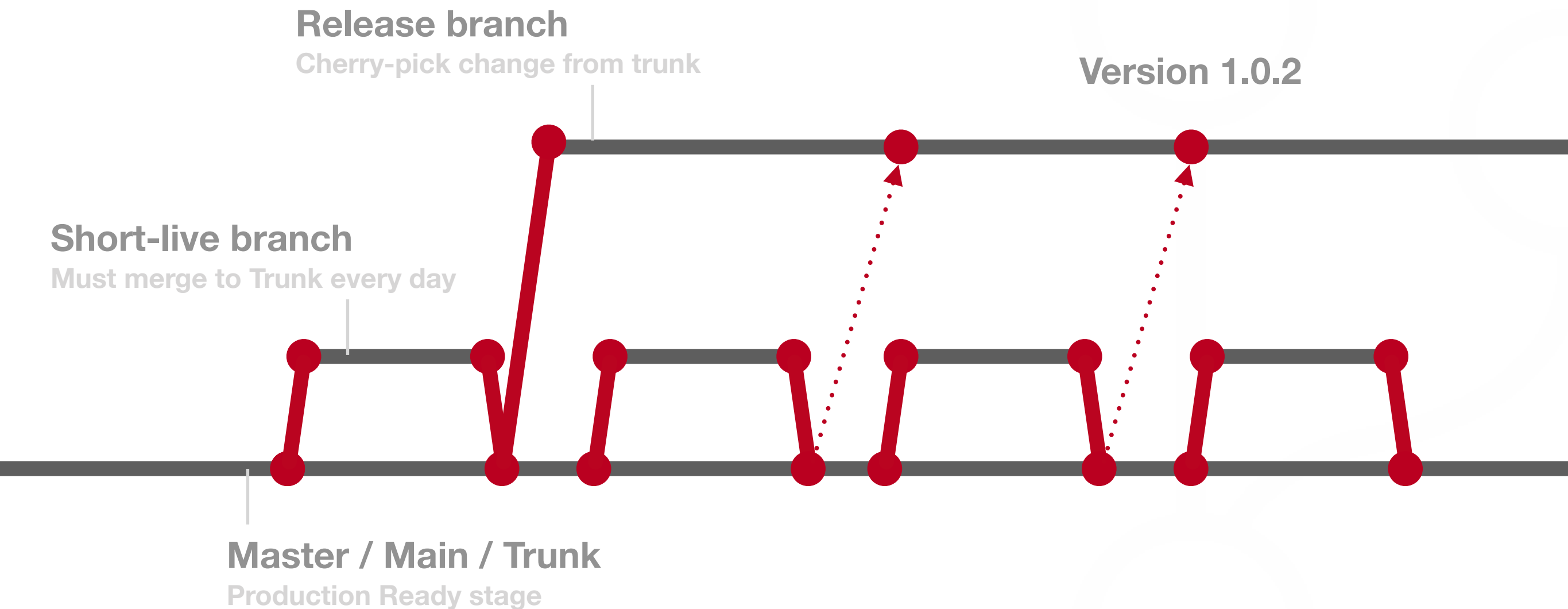
Trunk Based Development

What ?



Trunk Based Development

What ?



Trunk Based Development

What ?

Checking out / cloning

Developer will update/pull/sync from *Trunk* branch many times a day (*hour-by-hour*), knowing that the build passes

Trunk Based Development

What ?

Checking out / cloning

Developer will update/pull/sync from **Trunk** branch many times a day (*hour-by-hour*), knowing that the build passes

Committing

The developer needs to run the build, to prove that they did not break anything with the commit **before** the commit is pushed anywhere

Trunk Based Development

What ?

Checking out / cloning

Developer will update/pull/sync from **Trunk** branch many times a day (*hour-by-hour*), knowing that the build passes

Committing

The developer needs to run the build, to prove that they did not break anything with the commit **before** the commit is pushed anywhere

Code Reviews

The developer needs to get the commit reviewed. Some teams will count the fact that the code was 'pair programmed' as an automatic review

Trunk Based Development

What ?

Checking out / cloning

Developer will update/pull/sync from **Trunk** branch many times a day (*hour-by-hour*), knowing that the build passes

Committing

The developer needs to run the build, to prove that they did not break anything with the commit **before** the commit is pushed anywhere

Code Reviews

The developer needs to get the commit reviewed. Some teams will count the fact that the code was 'pair programmed' as an automatic review

A safety net

Continuous Integration (CI) daemons are set up to watch the **trunk** (and the **short-lived** feature branches used in review).

Trunk Based Development

What ?

Developer team commitments

Need to consider the impact of their potentially larger commits, especially where *renames* or *moves* were wholesale, and adopt techniques to allow those changes to be more easily consumed by teammates

Trunk Based Development

What ?

Developer team commitments

Need to consider the impact of their potentially larger commits, especially where *renames* or *moves* were wholesale, and adopt techniques to allow those changes to be more easily consumed by teammates

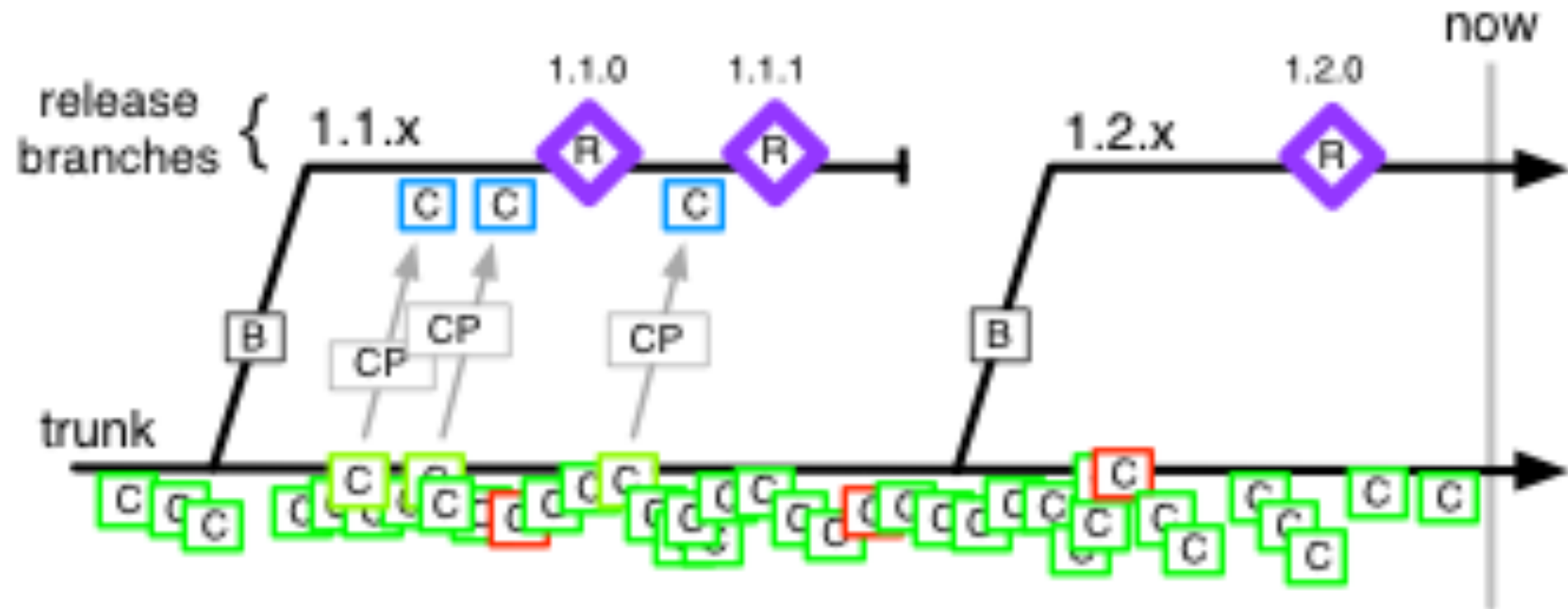
Drilling into 'Distance'

Late merges of development that happened more than a couple of days ago.

A breaking build that lowers development team throughput, and diverts resources while it is being fixed

Trunk Based Development

What ?



My Experience about Trunk Based

My Branching Model = Git flow + Github flow + ***Trunk Based***

My Experience about Trunk Based

Settings up GitHub repo

After pull requests are merged, you can have head branches deleted automatically.



Automatically delete head branches

Deleted branches will still be able to be restored.

- Automatic Remove Head Branch After Merged -

My Experience about Trunk Based

Settings up GitHub repo

☒ **Require pull request reviews before merging**

When enabled, all commits must be made to a non-protected branch and submitted via a pull request with the required number of approving reviews and no changes requested before it can be merged into a branch that matches this rule.

Required approving reviews: 1 ▼

☒ **Dismiss stale pull request approvals when new commits are pushed**

New reviewable commits pushed to a matching branch will dismiss pull request review approvals.

☐ **Require review from Code Owners**

Require an approved review in pull requests including files with a designated code owner.

☒ **Require status checks to pass before merging**

Choose which [status checks](#) must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.




☒ **Require branches to be up to date before merging**


This ensures pull requests targeting a matching branch have been tested with the latest code. This setting will not take effect unless at least one status check is enabled (see below).

- Before merge any PR must be passed status check -



My Experience about Trunk Based

Settings up GitHub repo

☐  match ocr with social seeing enabled campaign condition ✓  

 New feature

#63 opened 22 hours ago by suradidchao • Review required

☐  Feature/api core integration test ✗ 

#12 opened on Sep 18 by protal • Review required

- Make PR Readable -

My Experience about Trunk Based

Settings up GitHub repo


Conversation 4

Commits 27

Checks 0

Files changed 14

+608 -69



spksoft commented 5 days ago • edited

Implement elastic query builder of refactoring match-worker

This PR will add *elasticsearch update and upsert query builder* , *starter command* and call pre-process ze-matcher

✓

Add GenerateChannelIndexName

✓

Add GenerateCampaignIndexName

✓

Add BuildBulkUpdateIndexQuery

✓

Add get config into `main` pkg

✓

Add get command param into `main` pkg

✓

Polish some code


✓

Add comment to each step inside main

✓


Add preprocessZocialEyeMessage

Reviewers

 clozed2u

✓

Assignees

 spksoft

Labels

Engineering

✓ Ready to merge



Projects


None yet

- Make PR Readable -

My Experience about Trunk Based

Settings up GitHub repo

 **clozed2u** requested changes 2 days ago [View changes](#)




match-worker/cmd/main.go Outdated  Hide resolved

207 + <-exitSignalChannel


208 + for processingJobs > 0 {

209 + }

210 + runtime.Goexit()


 **clozed2u** 2 days ago  

return should work fine with this case.

 Reply...

Unresolve conversation

spksoft marked this conversation as resolved.




match-worker/cmd/main.go Outdated  Hide resolved

8 - fmt.Println("Execute ...")


9 - // Read flag

10 - // Read config

121 + defer os.Exit(0)

 **clozed2u** 2 days ago  

You don't need `os.Exit(0)` if return at the end of func.

 Reply...



Unresolve conversation


spksoft marked this conversation as resolved.

- Make PR Readable -

My Experience about Trunk Based

Settings up GitHub repo

 **clozed2u** requested changes 2 days ago [View changes](#)




match-worker/cmd/main.go Outdated  Hide resolved

207 + <-exitSignalChannel


208 + for processingJobs > 0 {

209 + }

210 + runtime.Goexit()


 **clozed2u** 2 days ago  

return should work fine with this case.

 Reply...

Unresolve conversation

spksoft marked this conversation as resolved.




match-worker/cmd/main.go Outdated  Hide resolved

8 - fmt.Println("Execute ...")


9 - // Read flag

10 - // Read config

121 + defer os.Exit(0)

 **clozed2u** 2 days ago  

You don't need `os.Exit(0)` if return at the end of func.

 Reply...

Unresolve conversation

spksoft marked this conversation as resolved.

- Make PR Readable -

Download Slide



ternng.life/ws-tb.pdf

Ref. <https://trunkbaseddevelopment.com/> , <https://nvie.com/posts/a-successful-git-branching-model/> , <https://guides.github.com/introduction/flow/> , <https://paulhammant.com/2014/01/08/googles-vs-facebooks-trunk-based-development/> , <https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow> , <http://www.somkiat.cc/git-branch-strategy/>