Automate merges to keep builds healthy

@ankitxg

Ankit Jain

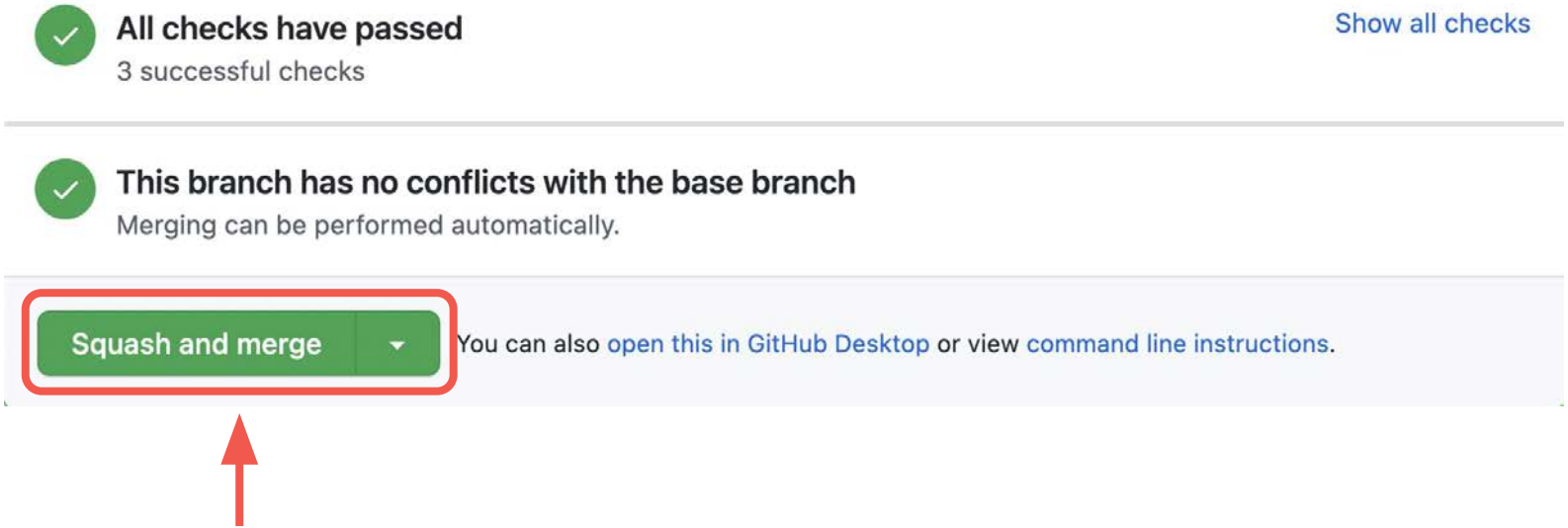
Cofounder of Aviator

Building Developer Workflow Automation Platform

Previously Engineer at:

Google | Adobe | Shippo | Homejoy | Sunshine

Merging? How hard can it be?



Monorepo

- Easier dependency management
- Simpler refactoring of cross-project changes
- Vulnerability management
- Standardized tooling
- Code sharing

Polyrepo

- Simpler CI / CD management
- Independent build pipelines
- Build failures are localized

Monorepo: How often do your mainline builds fail?

Is your current CI system enough?

Stale Dependencies

Implicit conflicts

Infrastructure issues

Internal dependencies

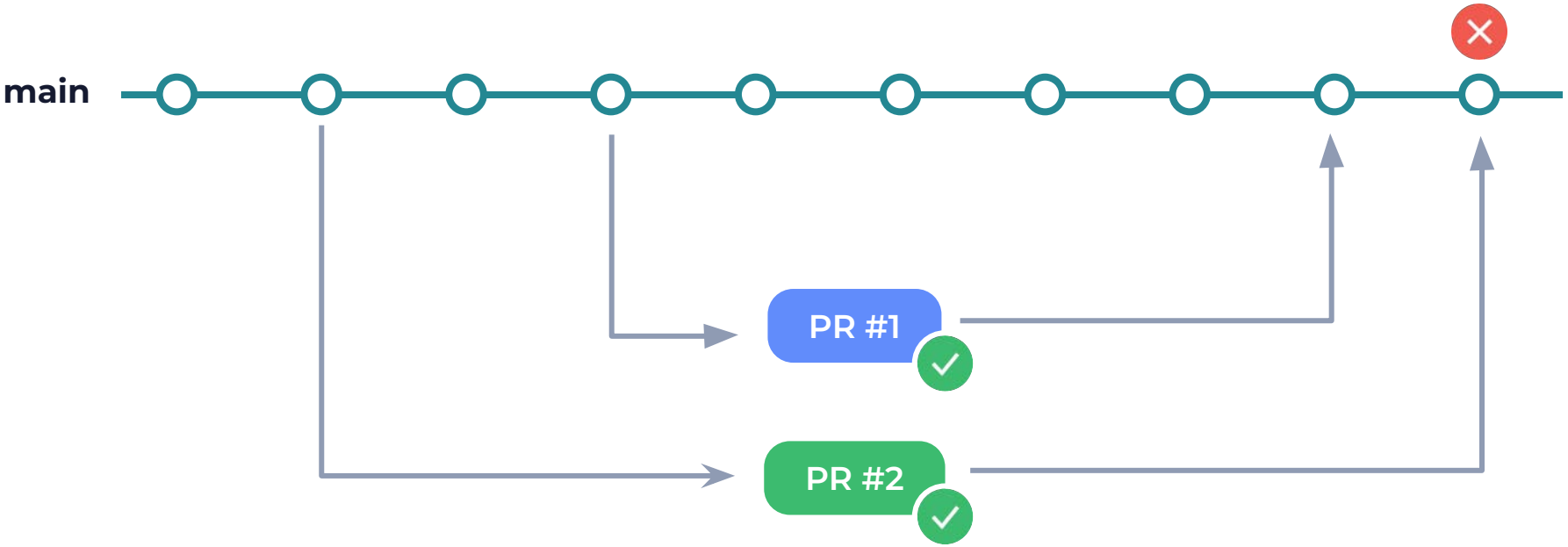
Timeouts

Third-Party Dependencies

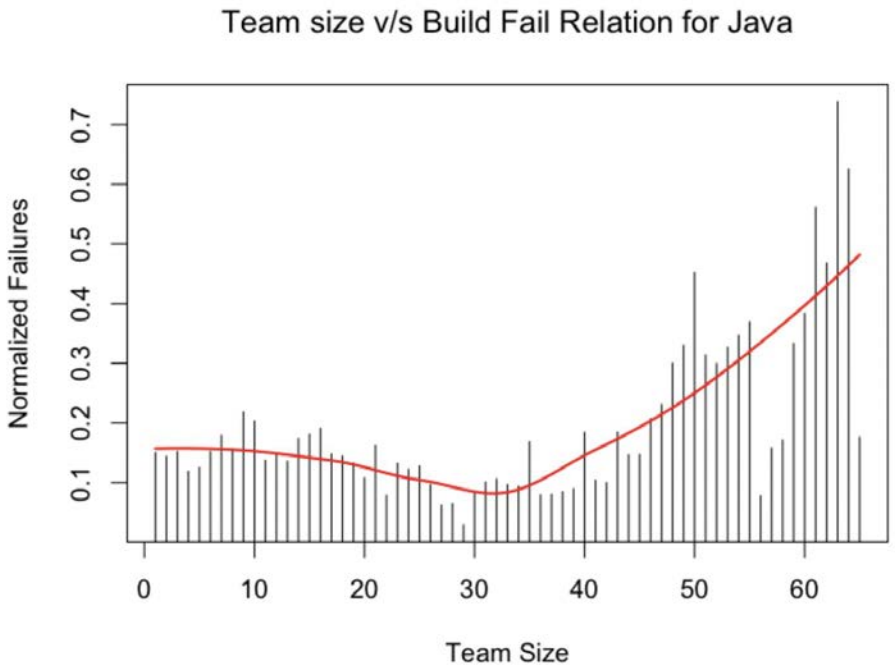
Race Conditions

Shared state / Concurrency Issues

Monorepo: Merging challenges



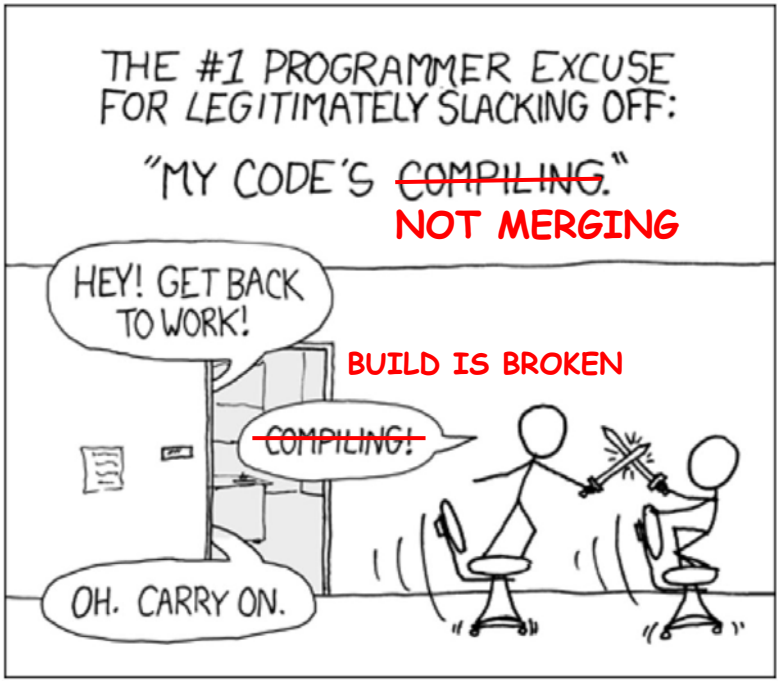
Impact of build failures on developer productivity



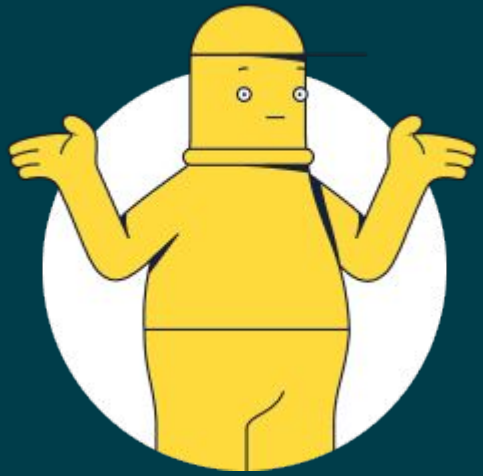
- Build Police
- Rollbacks
- Delayed releases
- Chain Reaction Failures

Source: ResearchGate

How often is your team stuck due to broken builds?



Source: [xkcd]



So, what’s the solution?

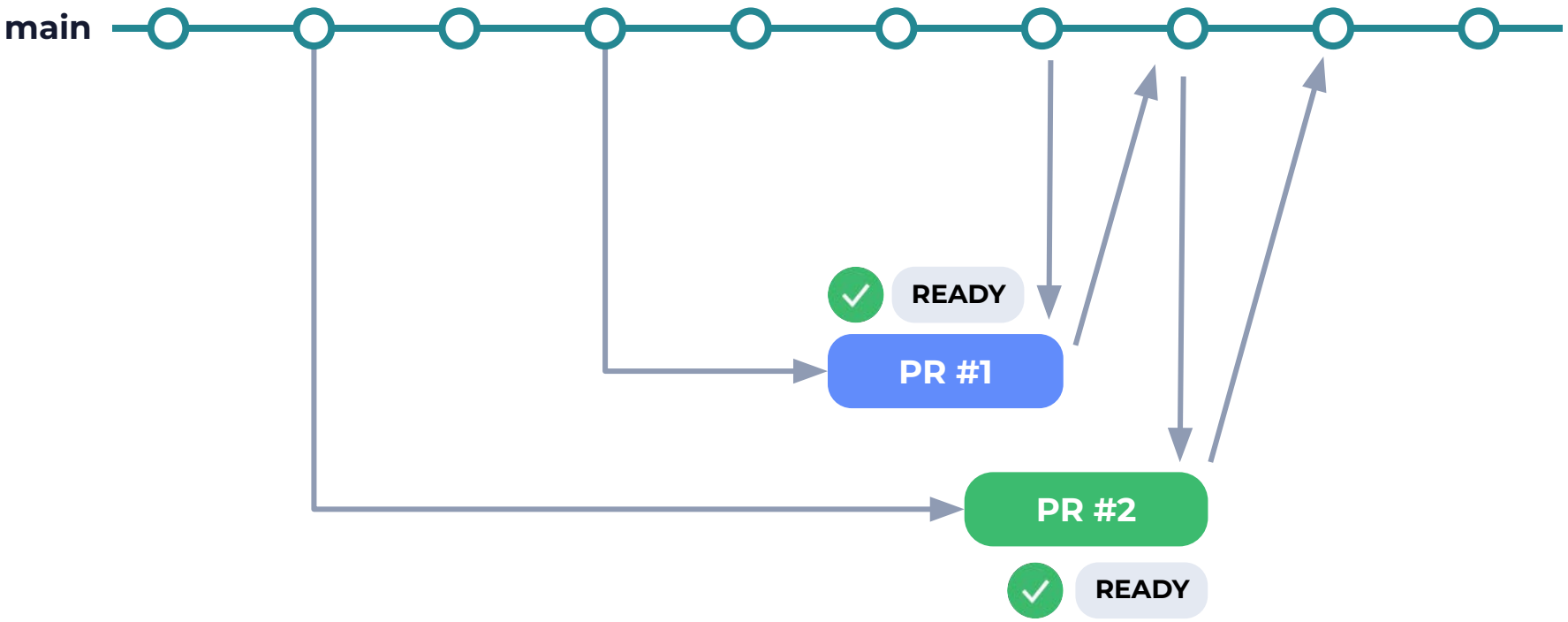
Merge automation

Merge Queue

Merge Train

Open Source

Simple Merge Queue



Simple Merge Queue - Performance Review

Small team

CI time = 30 mins

PRs per day = 10

Total merge time = **5 hours**

Total CI runs = **50**

Large team

CI time = 30 mins

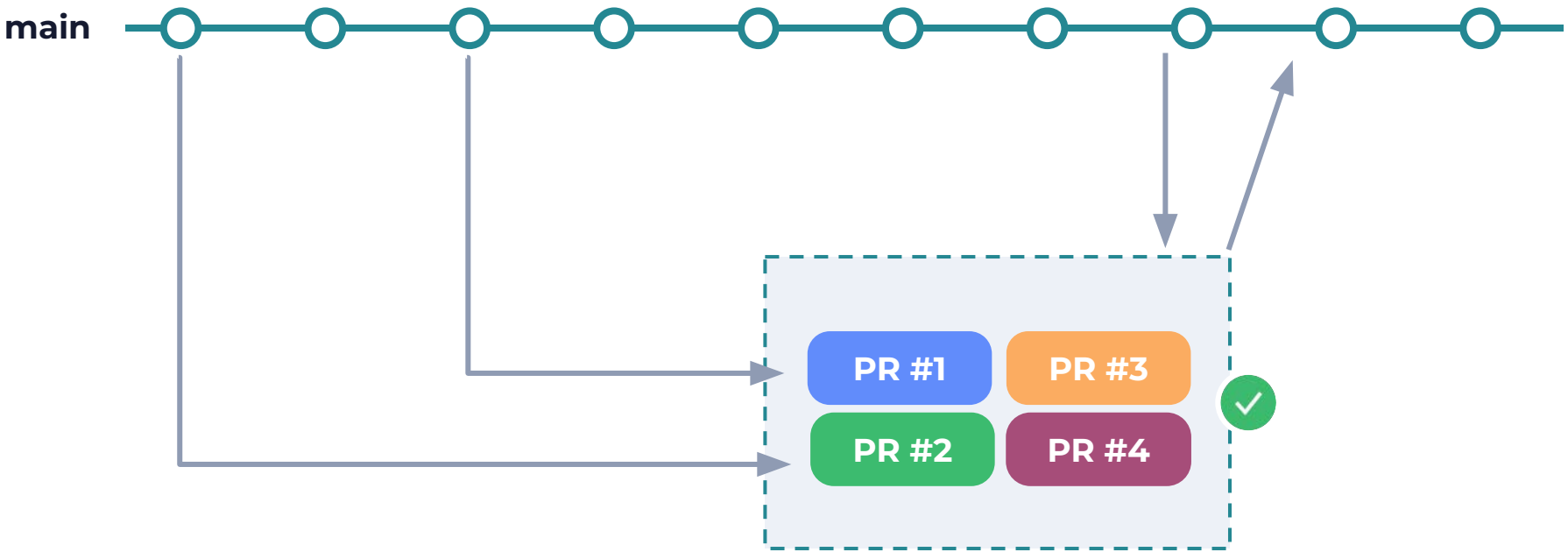
PRs per day = 100

Total merge time = **50 hours**

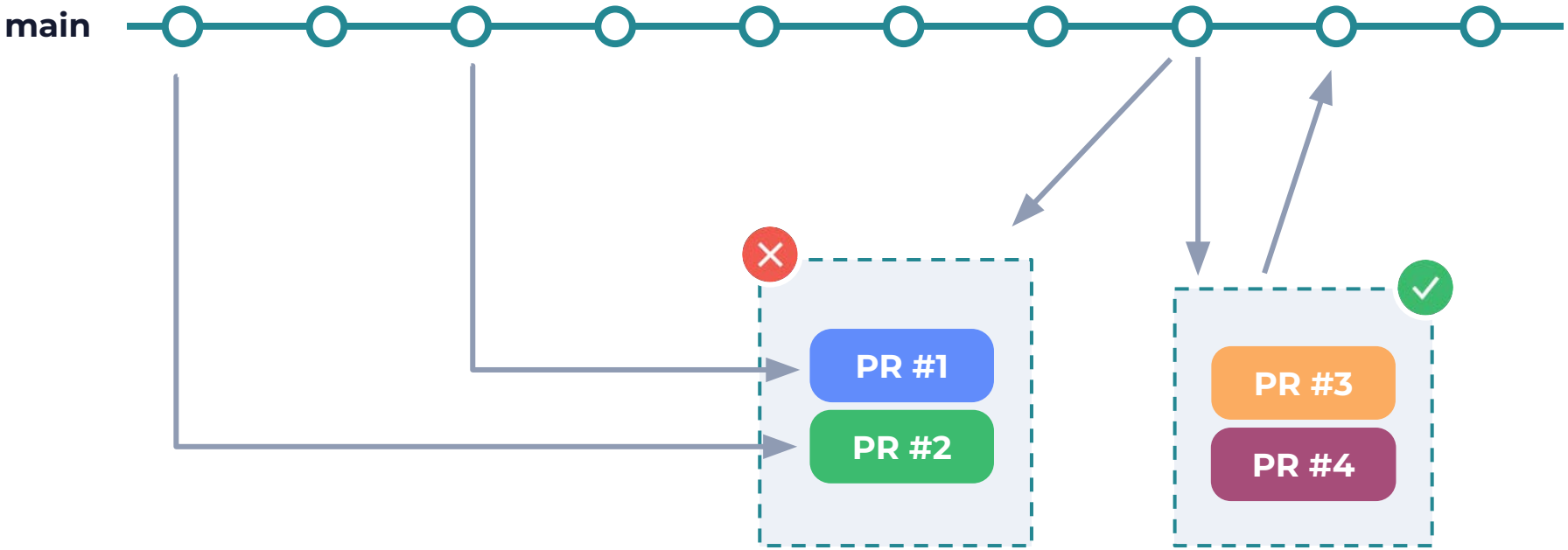
Total CI runs = **50**

Can we do better?

Batching Changes



Batching Changes - Bisect when fails



Batching Changes - Performance Review

Best case

CI time = 30 mins

PRs per day = 100

Batch size = 4

Total merge time = **12.5 hours**

Total CI runs = **25**

Median case

CI time = 30 mins

PRs per day = 100

Batch size = 4

Failure rate = 10%

Total merge time = ~**24 hours**

Total CI runs = ~**48**

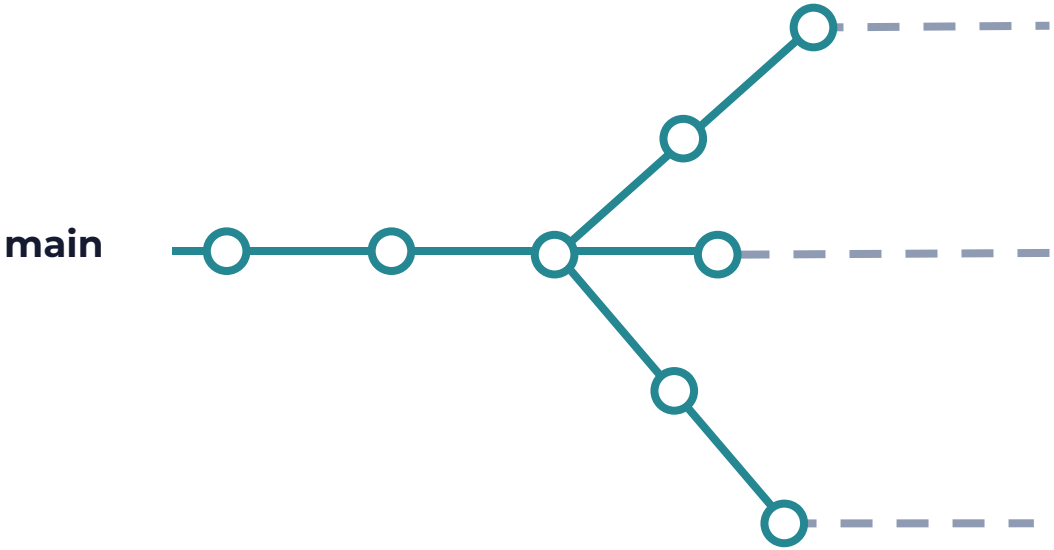
Can we still do better?

Reimagine merges

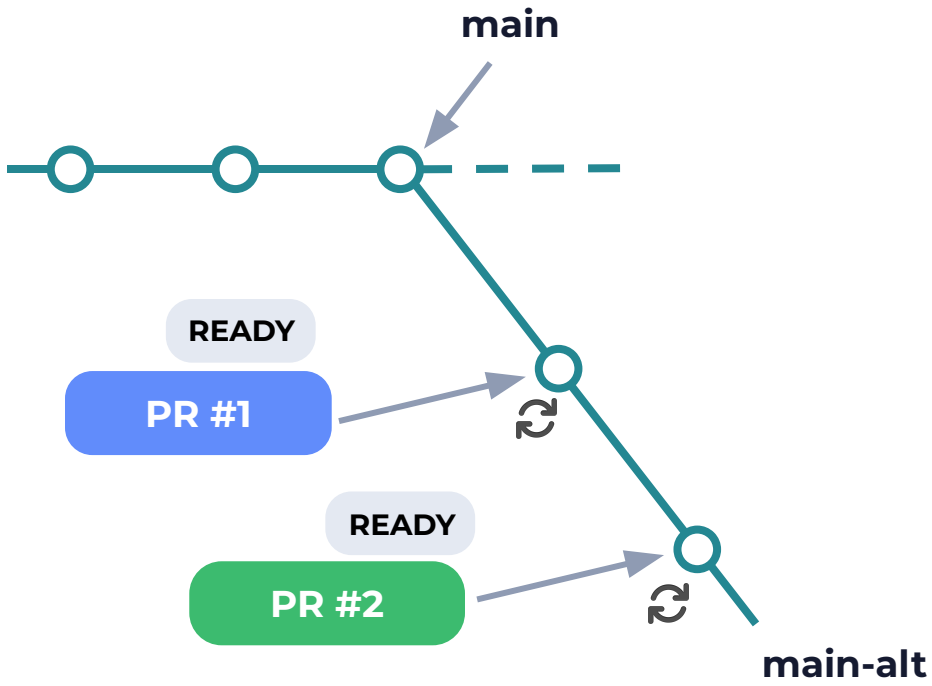
Reimagining merges

Parallel Universes

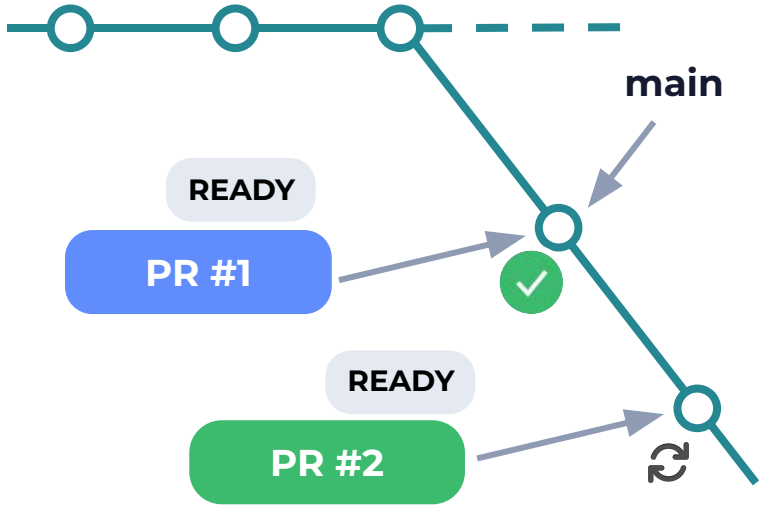
Reimagining merges



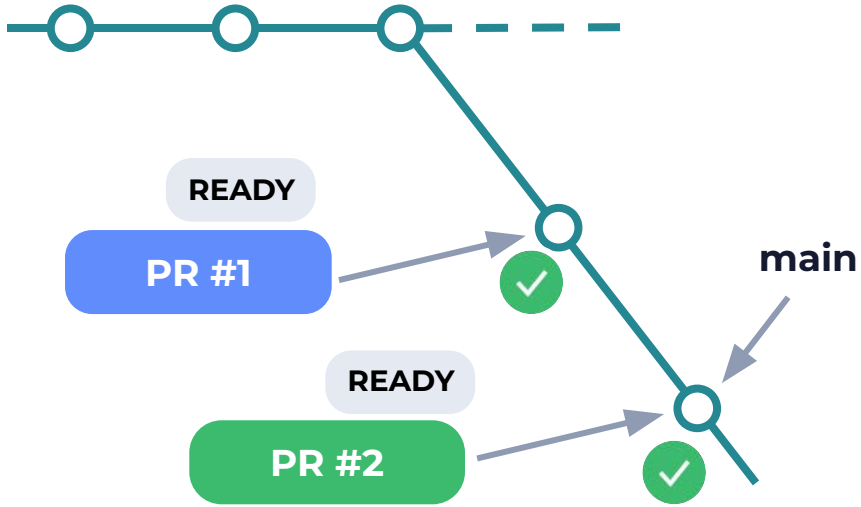
Reimagining merges - optimistic queues



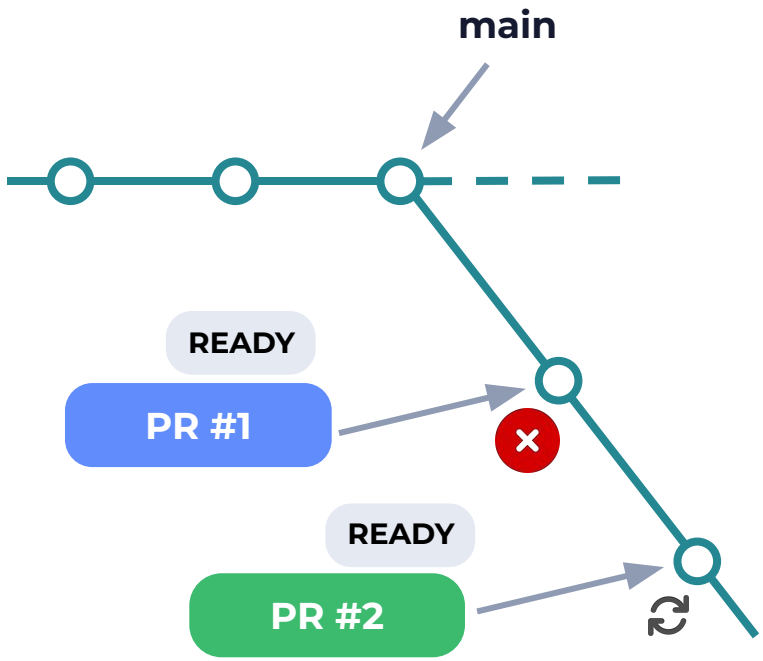
Reimagining merges - optimistic queues



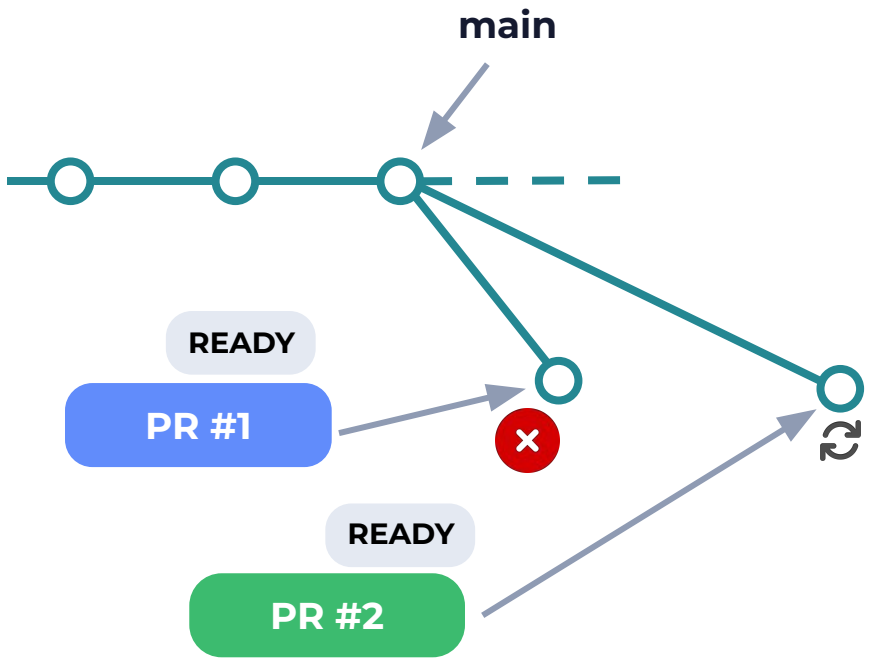
Reimagining merges - optimistic queues



Reimagining merges - Failure



Reimagining merges - Failure



Optimistic queue - Performance Review

Best case

CI time = 30 mins

PRs per day = 100

Total merge time = < **1 hour**

Total CI runs = **100**

Median case

CI time = 30 mins

PRs per day = 100

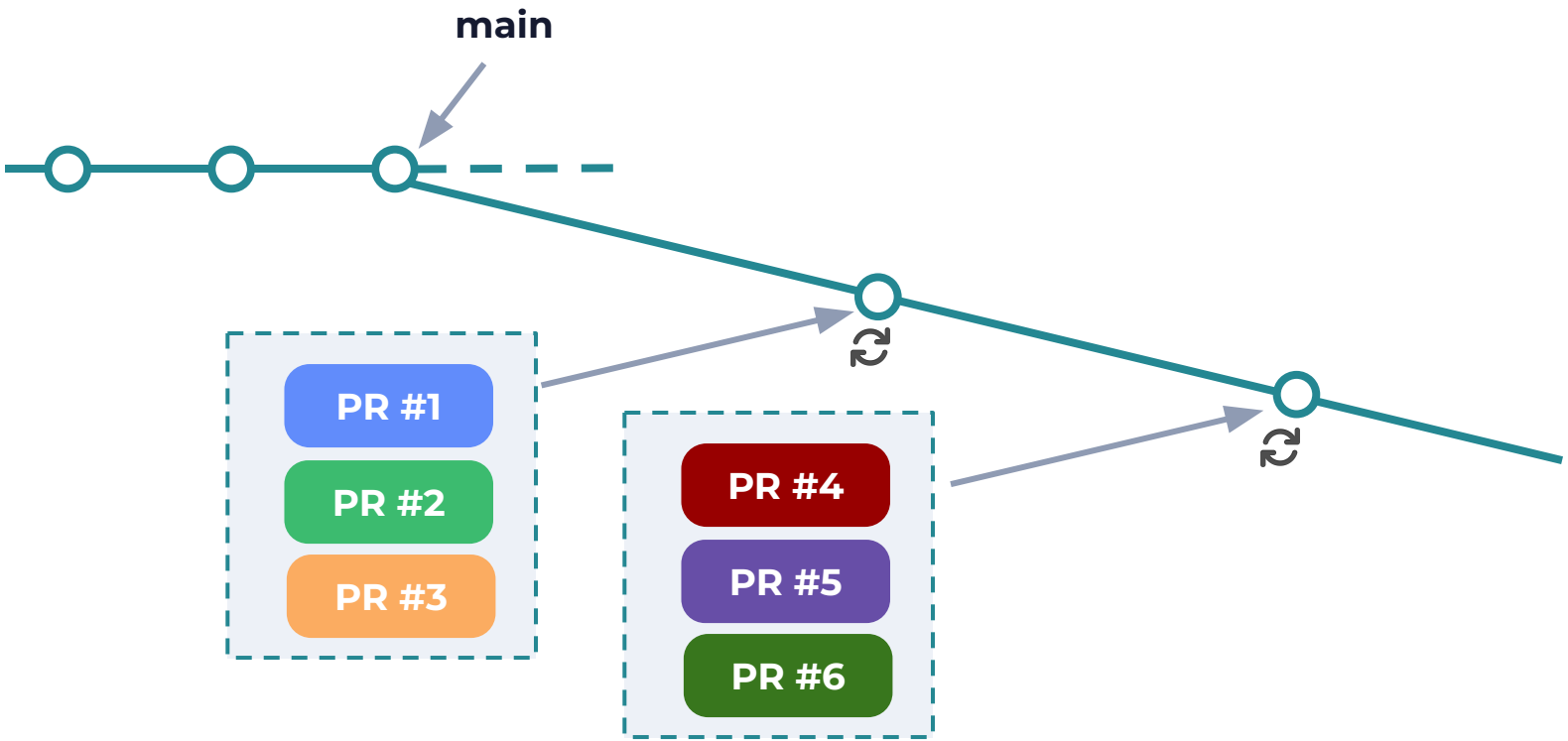
Failure rate = 10%

Total merge time = ~**6 hours**

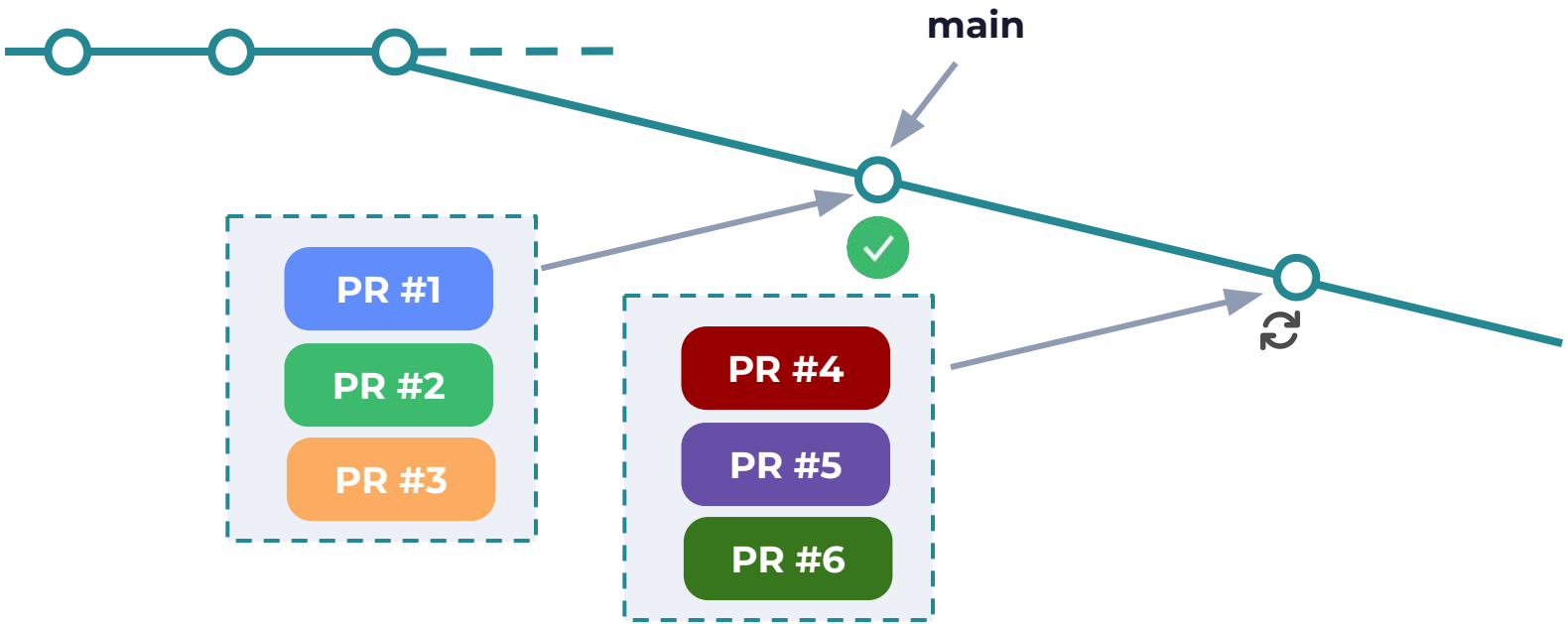
Total CI runs = **~150**

Can we still do better?

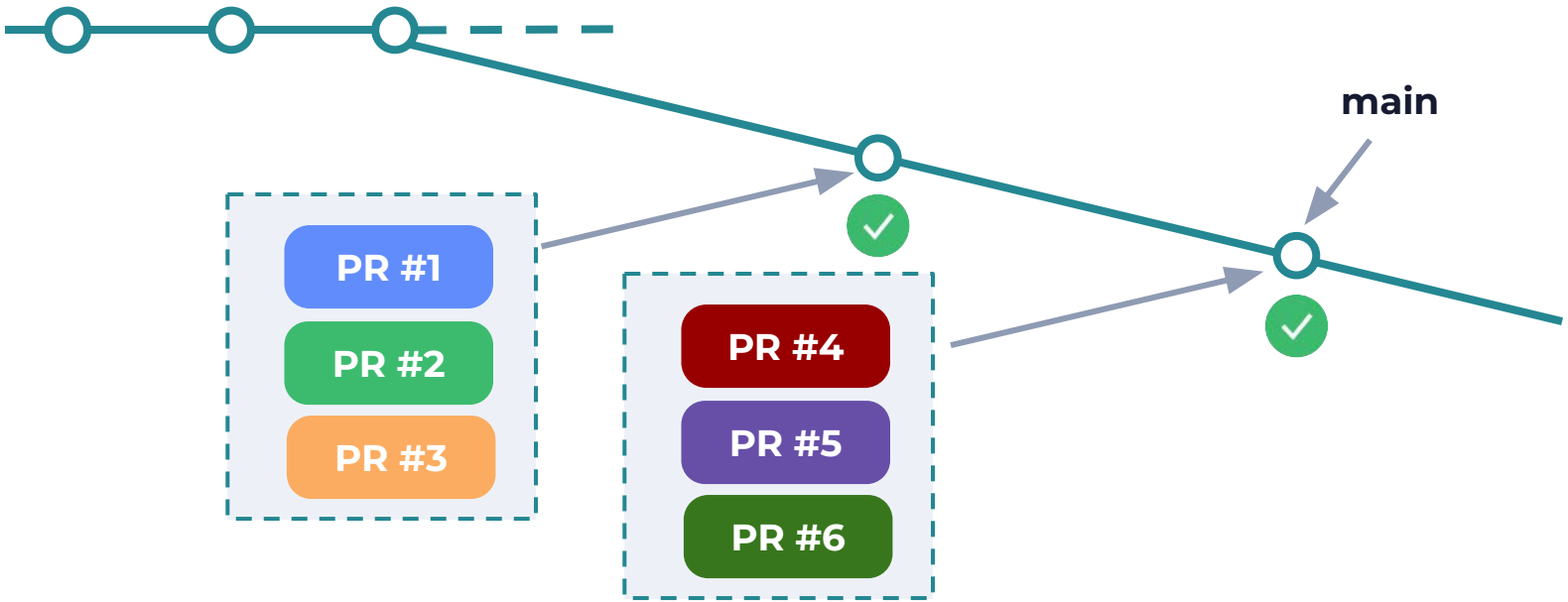
Batching optimistic queues



Batching optimistic queues



Batching optimistic queues



Optimistic queue (batching) - Performance Review

Best case

CI time = 30 mins

PRs per day = 100

Batch size = 4

Total merge time = < **1 hour**

Total CI runs = **25**

Median case

CI time = 30 mins

PRs per day = 100

Batch size = 4

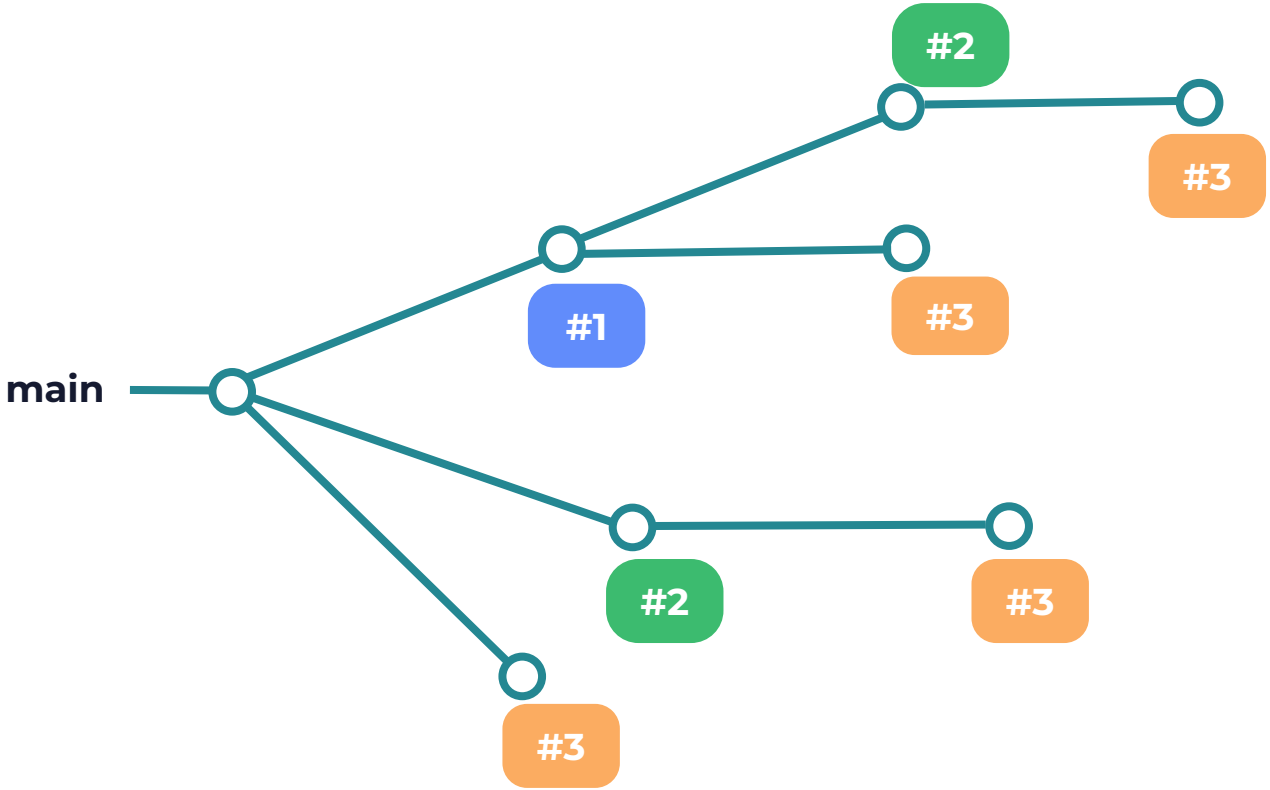
Failure rate = 10%

Total merge time = ~**4 hours**

Total CI runs = **~48**

Can we still do better?

Using predictive modeling



PR #1

PR #2

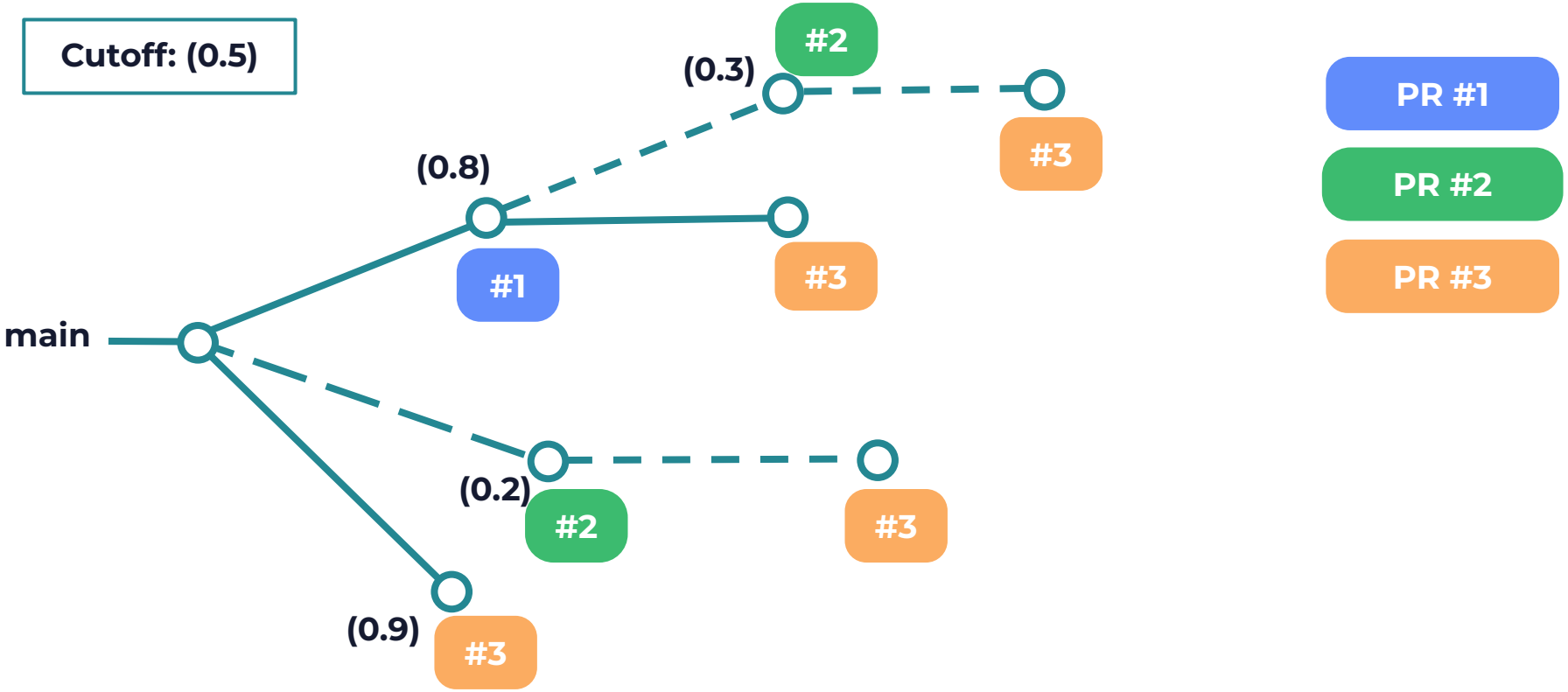
PR #3

Using predictive modeling

Optimization based on

- Lines of code
- Types of files modified
- Test added / removed
- Number of dependencies

Using predictive modeling



Can we still do better?

Multi-queues

Multi-queues - Using affected targets

order of queuing

PR #	Affected targets			
1	A			
2		B	C	
3				D
4		B		
5			C	
6	A			
7			C	
8	A			
9	A			

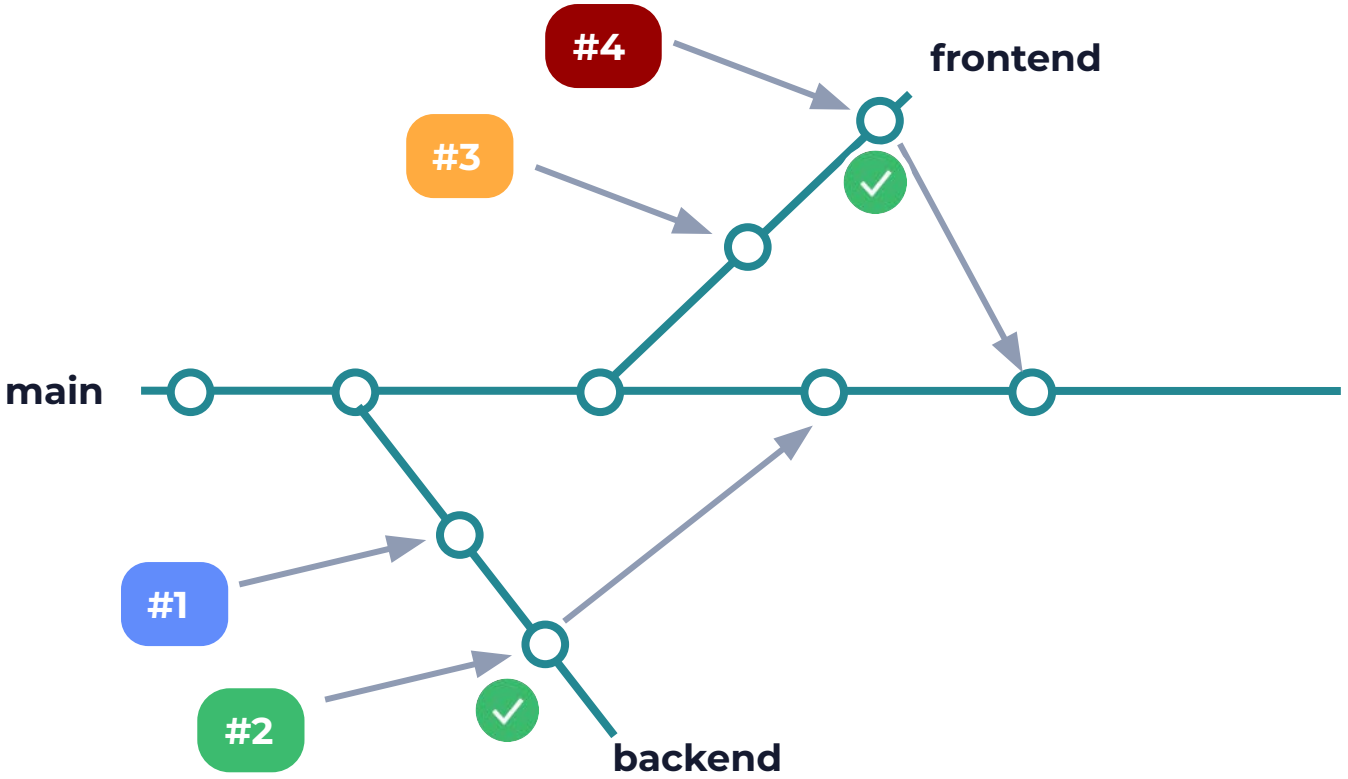
Q1 -> 1, 6, 8, 9

Q2 -> 2,4

Q3 -> 2, 5,6

Q4 -> 3

Disjoint multi-queues - Using affected targets



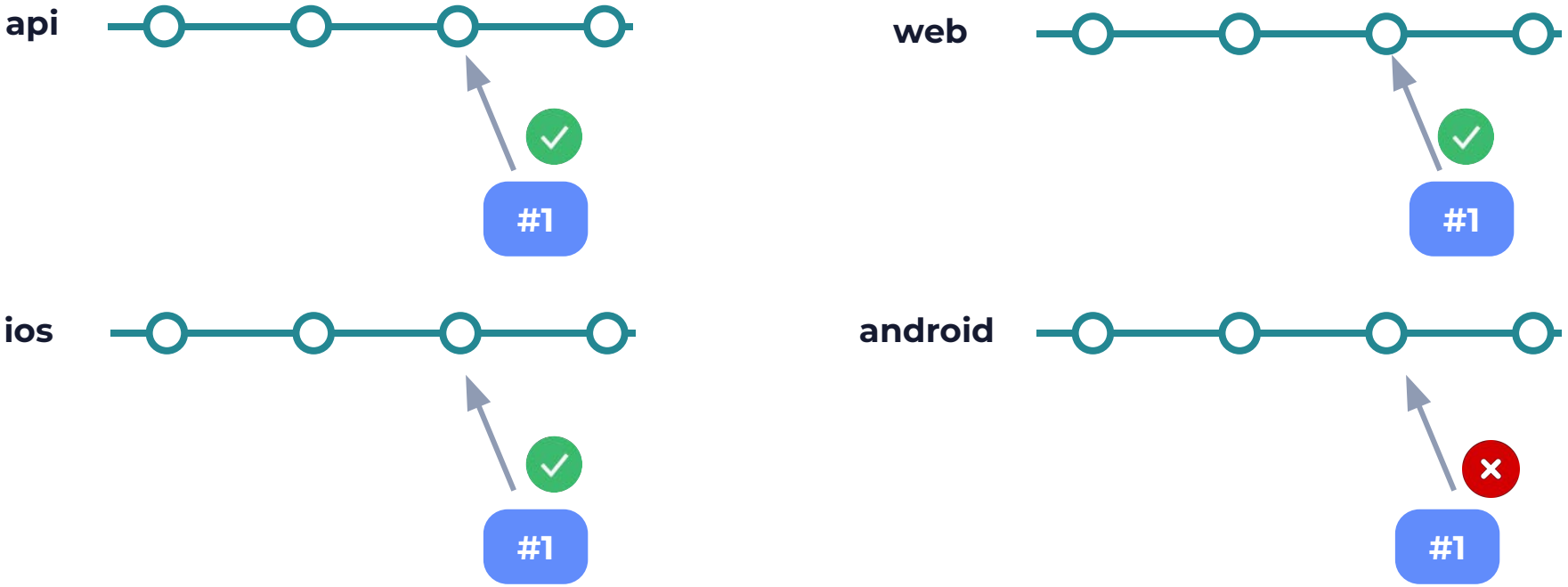
Further optimizations

- **Reordering changes** - high priority, lower failure risk
- **Fail fast** - Reordering test execution
- **Split test execution** - Pre-merge and post-merge testing

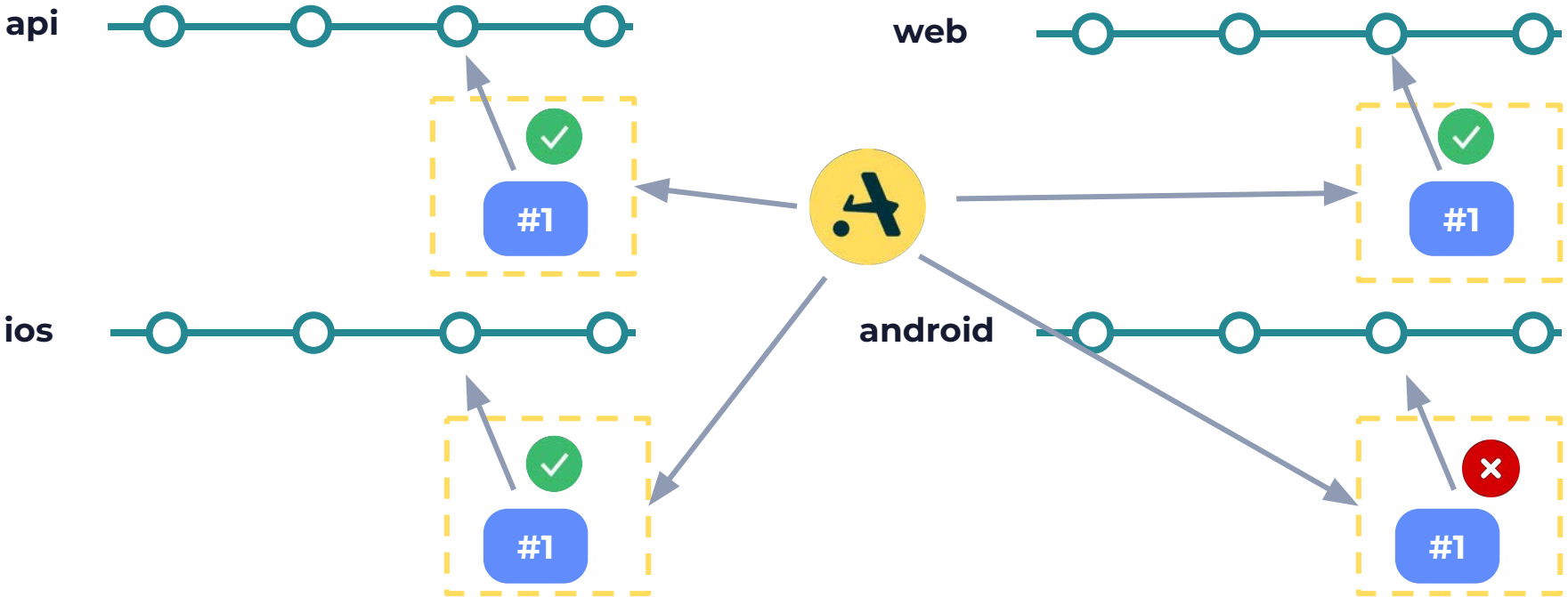
Other works

Polyrepo

Merging challenges in Polyrepo

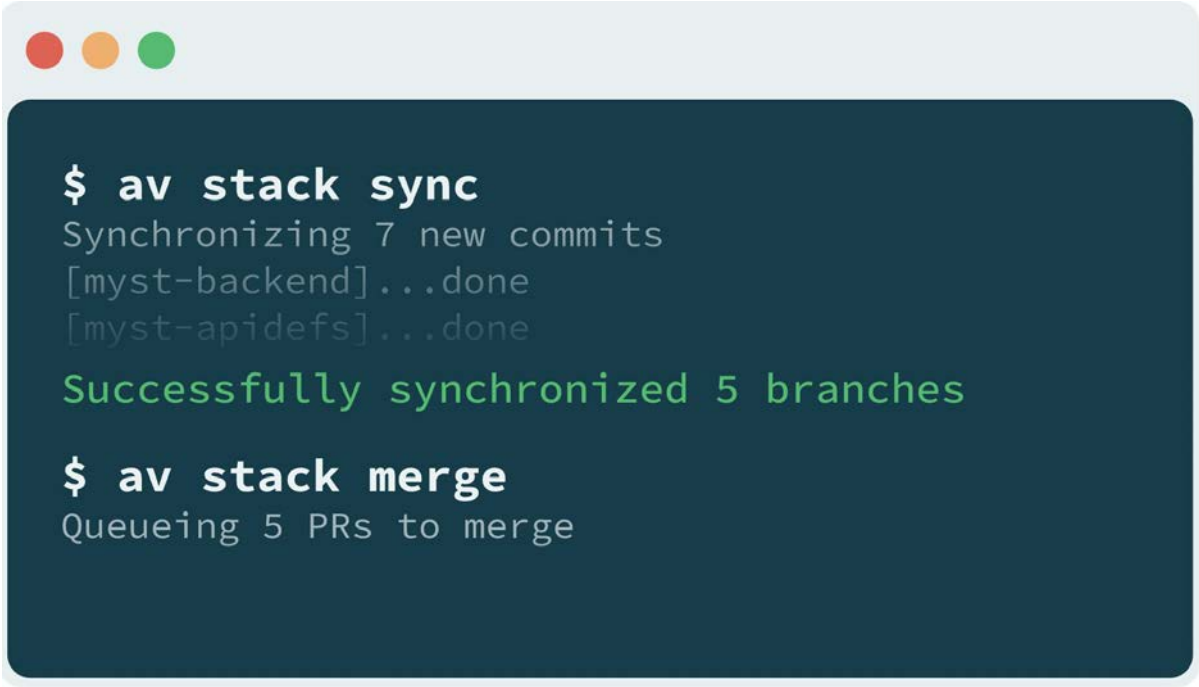


Merging challenges in Polyrepo



Flaky test management

Managing Stacked PRs



References

- Keeping master green at scale - Uber
- Bors
- Evergreen - Airbnb's merge queue
- Merging code in high-velocity repositories - LinkedIn

Questions?

ankit@aviator.co
@ankitxg

We are hiring: **aviator.co/jobs**