Is your Gatekeeper Locking out Quality?

Amy Phillips

Amy Phillips

About me

Songkick

About Songkick - business, users, technology

A story of moving to fast, frequent releases

The problem with releases

Releases are scary.

Often infrequent, leads to more anxiety, less familiarity

Mitigate by testing. And making sure testers do a good job, and that they're the ones agreeing to release the software.

No one likes bad releases so we add safe guards. Usually these safe guards are sign offs. We create gatekeepers to keep us safe.

A typical release

Code

Release candidate

Bug Report

Developer

Build Manager

Tester

Ops Engineer

What is Quality?

Definition

What is Testing?

Definition

The problem with gatekeepers

In ancient times gatekeepers, also called doorkeepers, served at various places, such as city gates, temple gates, even at the gateways or doorways of homes. Gatekeepers of city gates were appointed to see that the gates were closed at night, and they acted as watchmen at the gate.

http://wol.jw.org/en/wol/d/r1/lp-e/1200001618#h=2

Gatekeepers protect the city by keeping the bad people out.

Not a great role model for an aspiring DevOps team!

What are we trying to achieve with releases?

Releases when we want them. Without the unexpected surprises of downtime, bugs, unhappier users.

Several years ago at Songkick we were in exactly this place.

I was the gatekeeper. My job was mostly to tell developers that they couldn't integrate code. We couldn't release fast enough to keep up with the amount of code that was being produced. Pretty much everyone was frustrated.

The Team

About the team, our culture and environment

Not a simple problem to solve

There were many things holding us back:

test environments didn't match production
Silos
Difficult to debug issues
Slow to realise there was an issue in production
Developers didn't have environments to code and test.
Confidence
…and so on.

Fix the right problem

Releasing less frequently and doing more testing was simply not going to overcome these issues. Applying a gatekeeper role had delayed our ability to ship quality releases more than we had realised.

Step 1 - identify the single biggest (perceived) bottleneck and remove it.
Step 2 - identify the next biggest (perceived) bottleneck and remove it.
Repeat until you run out of bottlenecks.

At Songkick that meant fixing our developer environments. Removing downtime from releases to test environments, improved production monitoring, and most of all trust.

Trust

Testers learning to trust developers and automated tests
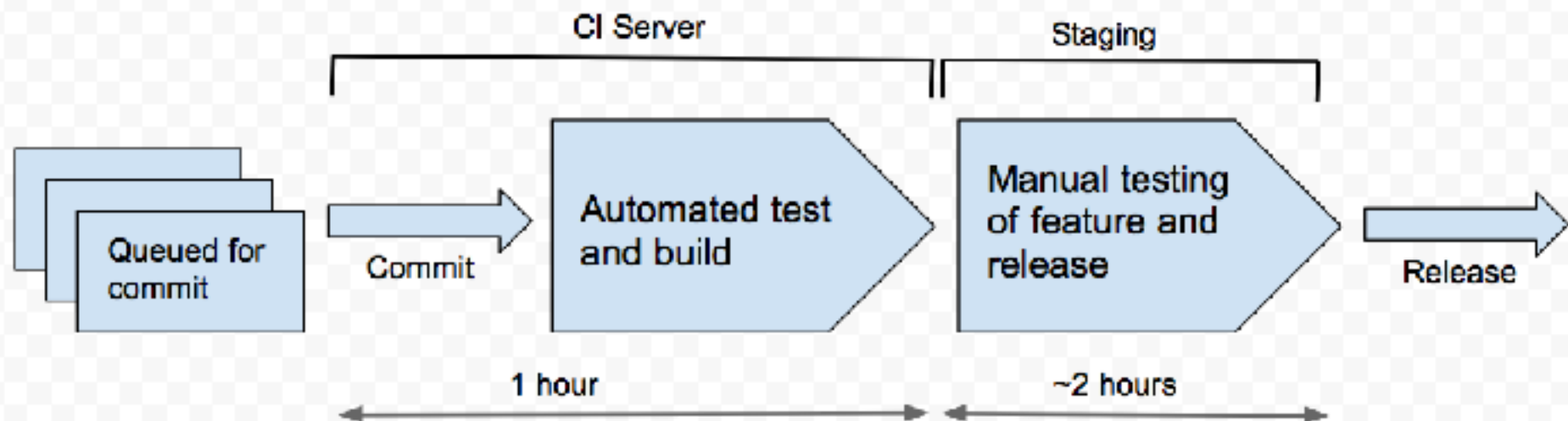Developers learning to trust testers and automated tests
Operations Engineers learning to trust monitoring and developers
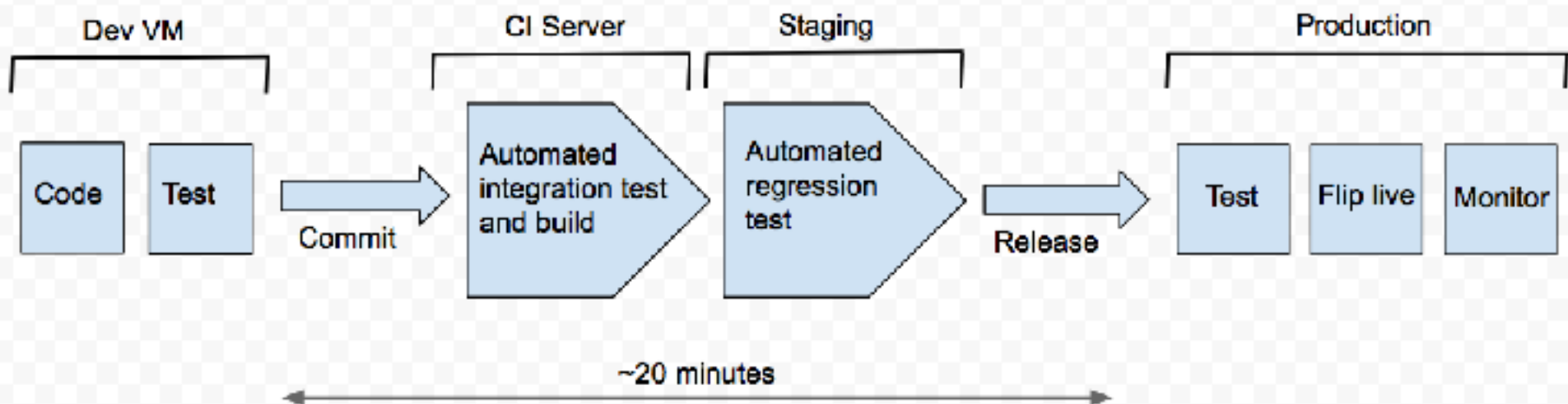Business departments learning to trust the development teams
Managers trusting workers.
Workers trusting managers.

The old release process

The current release process

The result

Quality at Songkick relies on users having the features they need, and want, developers having code and tools to do their job, testers being involved in a meaningful way, business owners being included in requirements, product managers having the team, and freedom to build the product that they want to build. None of these things are achieved by a gatekeeper holding back releases. All are achieved by creating a team of informed, trusted individuals who are able, and motivated to release frequently.

Thanks!