Why I Prefer Good Testing over Excellent Testing

Tina Fletcher - @fletchertinam

StarCon - January 2017

~~with enough time, we'll find every bug!~~

But...

Do testing, and also prepare for failure

Excellent Testing:

- Think of as many test cases as you can
- Try to run them all
- Keep testing until you run out of time

Good Testing:

- Think of as many test cases as you can
- Run the ones that will tell you the most about how well your product is working
- Spend remaining time planning failure handling

D2L™

brightspace

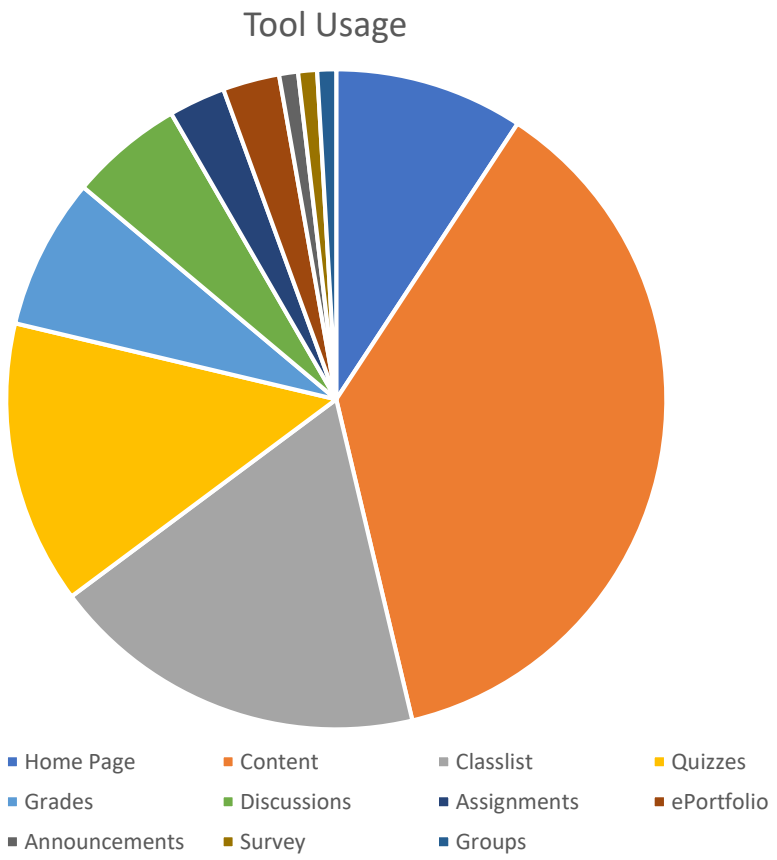
Defects I've Missed Lately

(and wouldn't have found with all the time in the world)

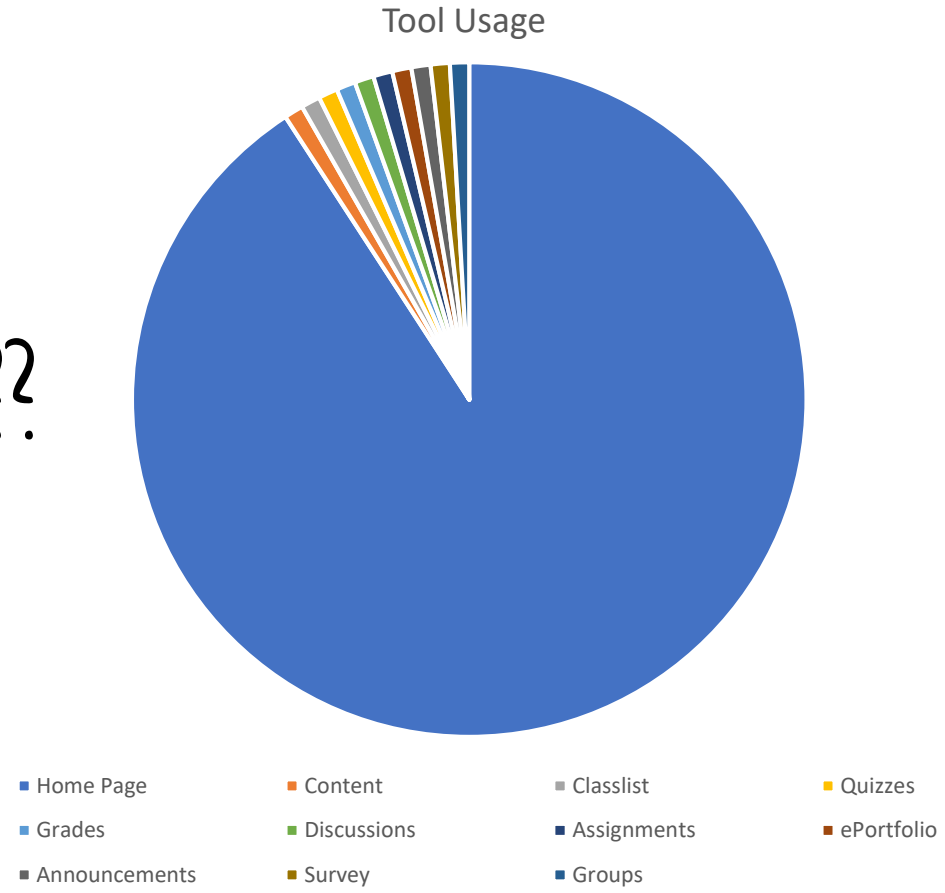
Story 1:

Monitoring user REALLY loves the home page

Story 1: Monitoring user REALLY loves the home page



why??



Each school has an automated "monitoring" user that hits the home page tool once <u>every 5 minutes</u> to make sure the site is up.

Story 1: Monitoring user REALLY loves the home page

LESSON:

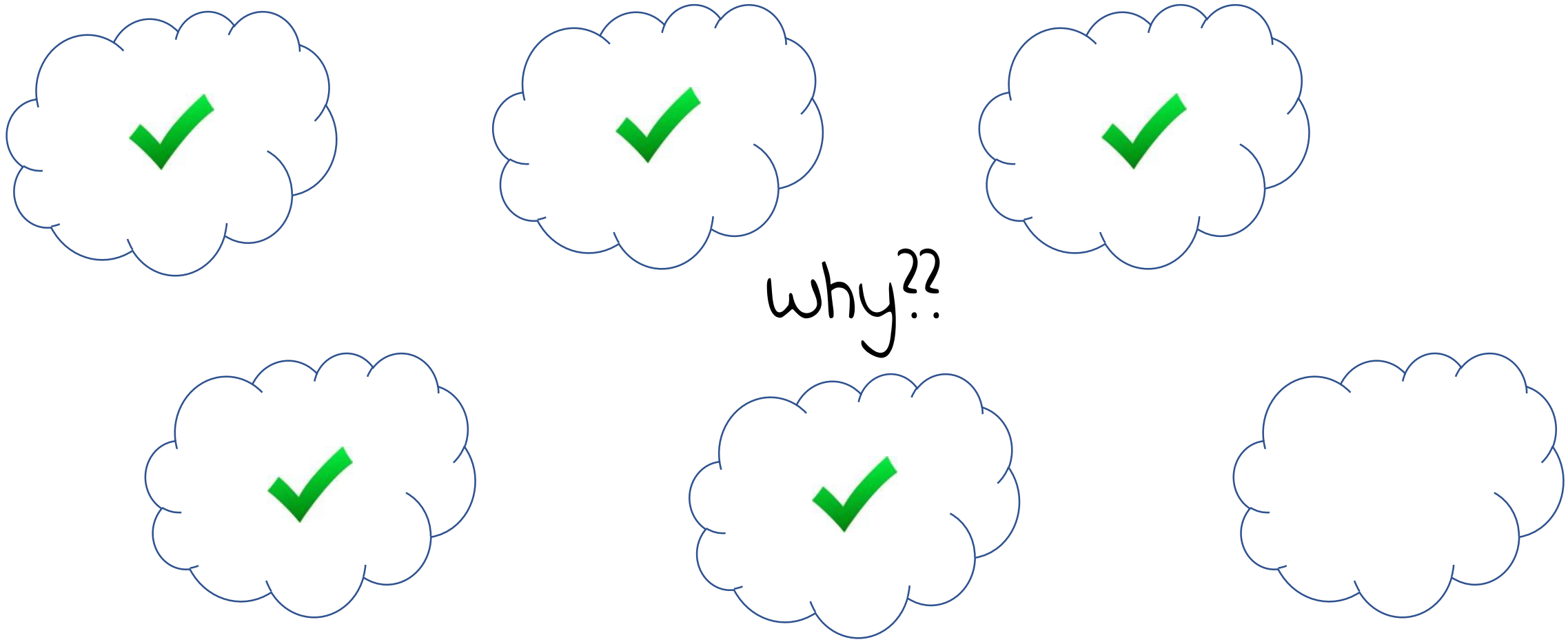
You don't know what you don't know.

(and you won't be testing for it)

Story 2:

The port that was closed for no reason

Story 2: The port that was closed for no reason



Port X was closed. But it was open when we last checked.

Story 2: The port that was closed for no reason

LESSON:

what's true today might not be true tomorrow.

Story 3:

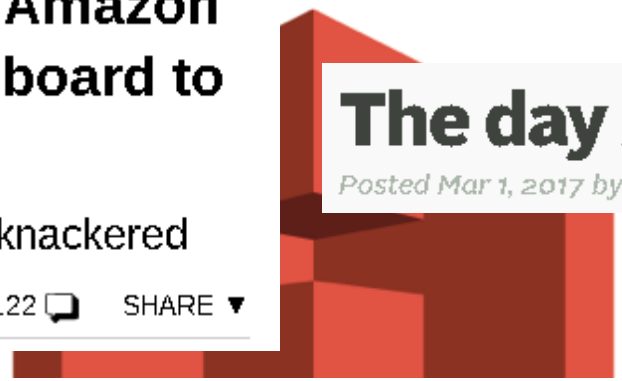
Yeah, but S3 will never go down

Story 3: Yeah, but S3 will never go down

AWS's S3 outage was so bad Amazon couldn't get into its own dashboard to warn the world

Websites, apps, security cams, IoT gear knackered

By Shaun Nichols in San Francisco 1 Mar 2017 at 03:00 122 SHARE



The day Amazon S3 storage stood still

Posted Mar 1, 2017 by Ron Miller (@ron_miller)

FEB 28, 2017 @ 04:01 PM 29,059

The Little Black Book of Billionaire Secrets

Amazon S3 Outage Has Broken A Large Chunk Of The Internet

How a typo took down S3, the backbone of the internet

Hello, operator

By Casey Newton | @CaseyNewton | Mar 2, 2017, 1:24pm EST

S3

Amazon AWS S3 outage is breaking things for a lot of websites and apps

Posted Feb 28, 2017 by Darrell Etherington (@etherington)

Story 3: Yeah, but S3 will never go down

LESSON:

Things that are out of your control can impact you.

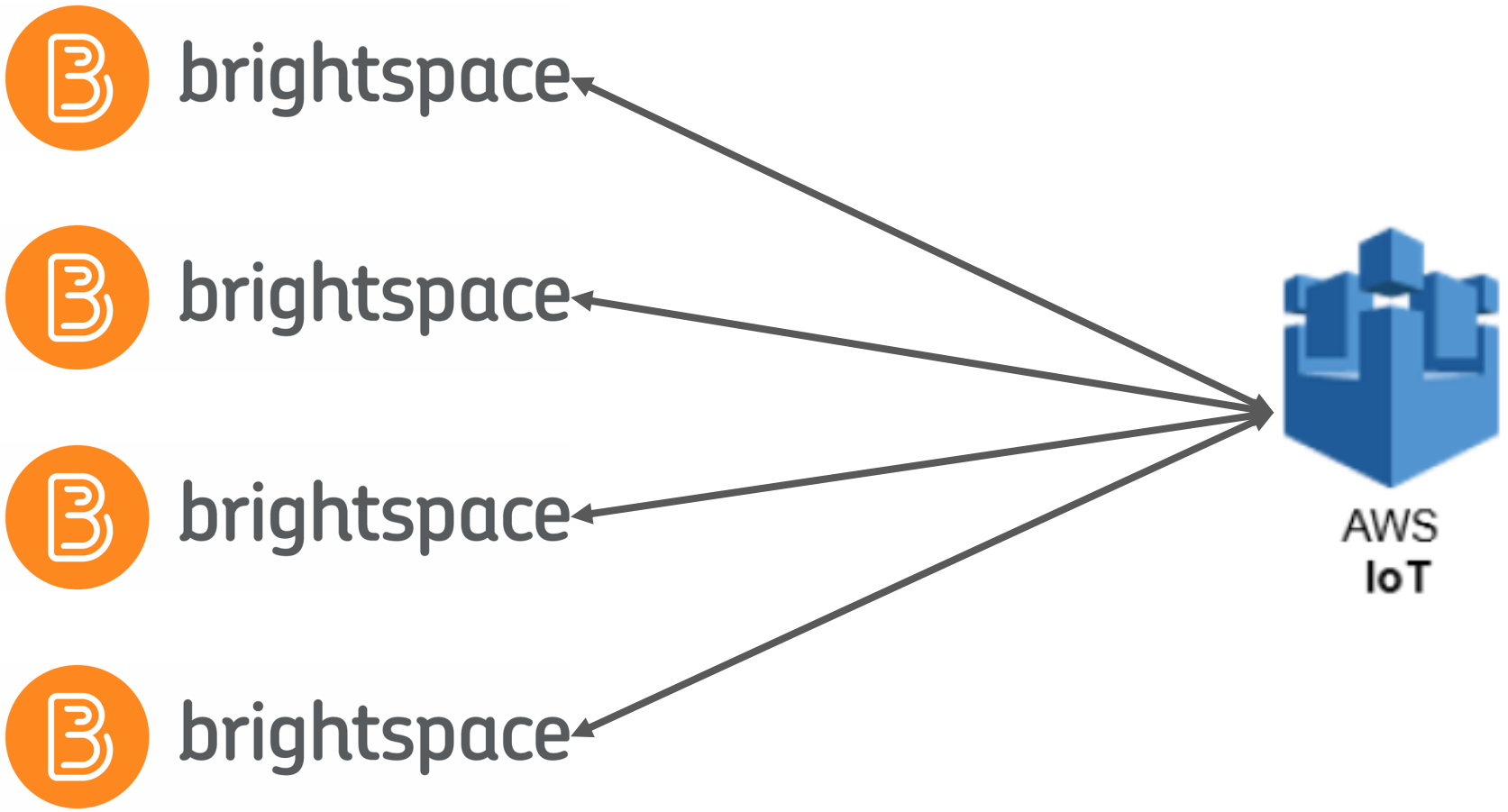
(and nothing is invincible)

Times I Have Been Saved By Solid Monitoring and Roll-Back Strategies

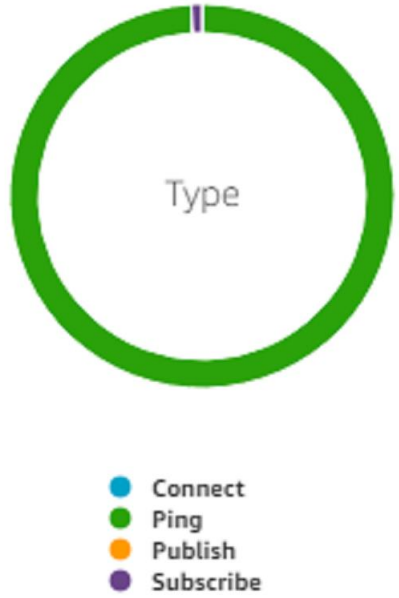
Story 4:

The off by 10,000 error

Story 4: The off by 10,000 error



Story 4: The off by 10,000 error



Every instance of the software was pinging the message broker once every one second.

Story 4: The off by 10,000 error

LESSON:

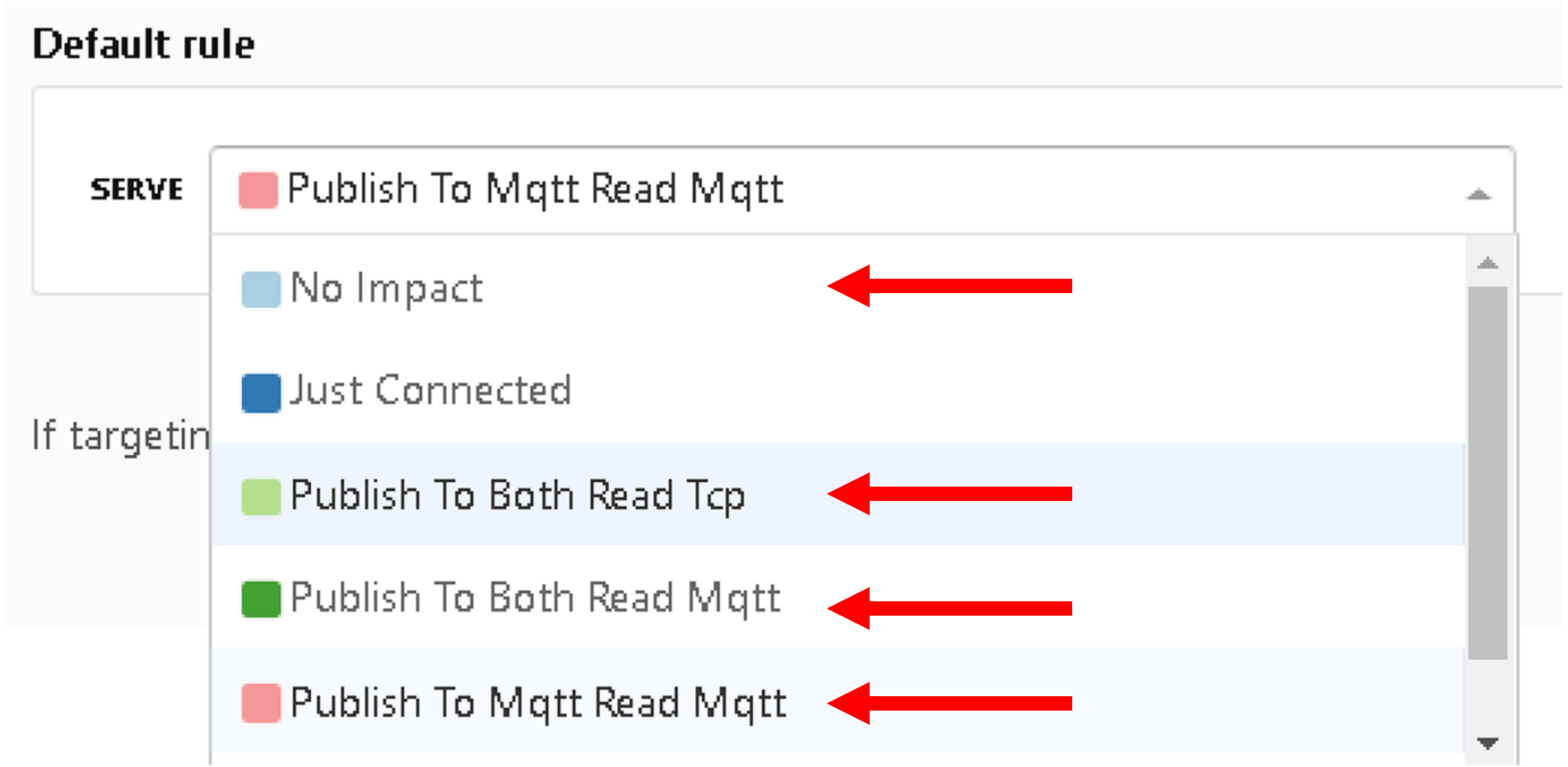
If your thing is working fine, it doesn't mean there are no problems.

Story 5:

The calmest production failure ever

Story 5: The calmest production failure ever

Story 5: The calmest production failure ever



Story 5: The calmest production failure ever

LESSON:

Safe roll-out and roll-back strategies allow you the luxury of thinking clearly during "emergencies".

Conclusion:

How do I make "good testing" work for me?

1. Do good testing

- Test until you feel pretty confident
- Skip tests that are not "real" enough
- Skip tests that cost more than they're worth
- Skip tests where failure would not result in a bug fix

2. Plan for failure

- Is it possible to roll out our changes gradually?

- What information is needed to detect that something is going wrong?

- How will we be notified when things are going wrong?

2. Plan for failure

- who is responsible for responding to error notifications?

- what troubleshooting steps will we take when we get a notification?

- what communications should occur while we are dealing with issues?

2. Plan for failure

- Are there cases where fixes or solutions could be executed automatically?

- what's the fastest way we could roll back to an earlier version?

- what are our contingency plans when there are problems with external dependencies?

Remember:

Be ready to handle failure; you won't find all the bugs anyway

Thanks!

@fletchertinam