



Turning the Quality of Your Deployment Pipeline Into a Team Task

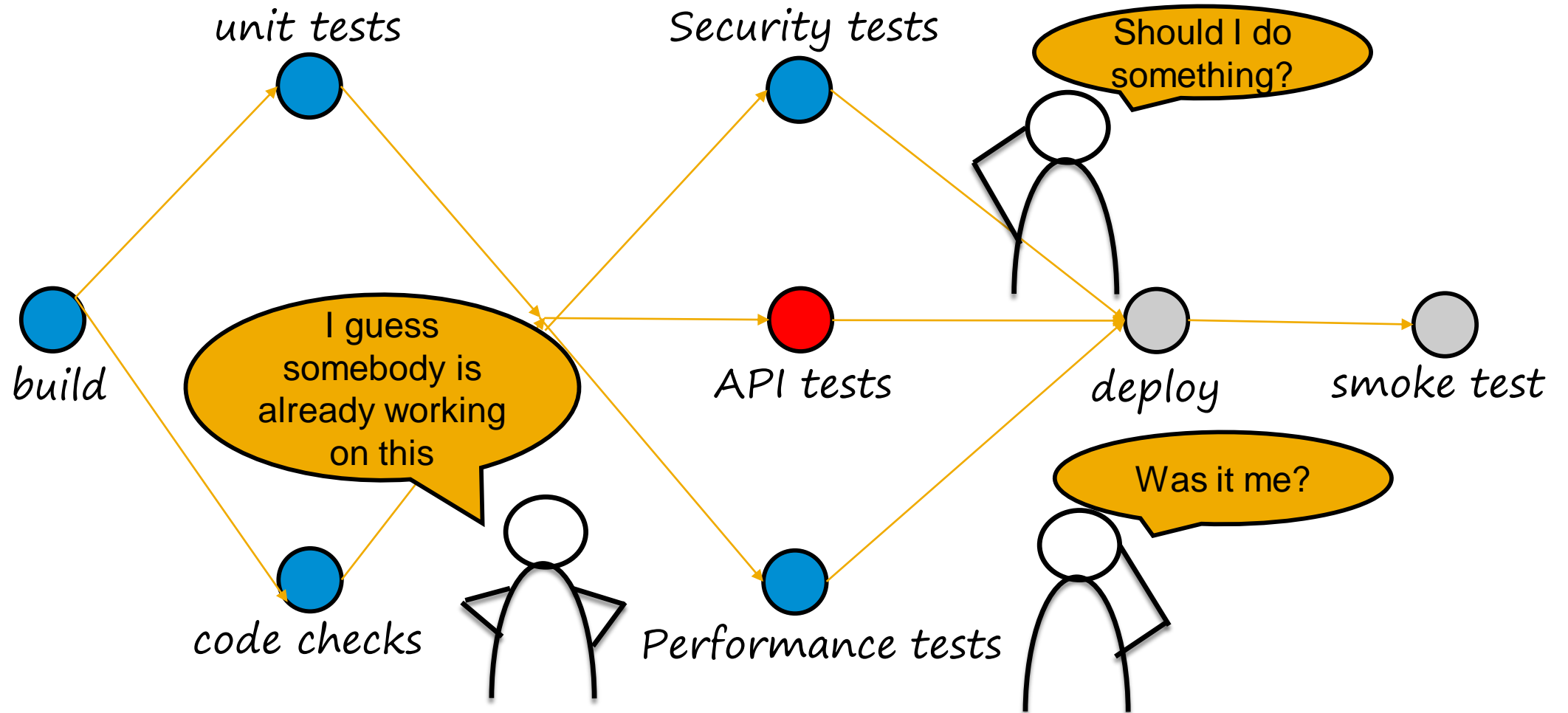
Areti Panou, SAP
October 3rd 2019

PUBLIC



THE BEST RUN 

Who is fixing the failing pipeline steps?



The bystander effect

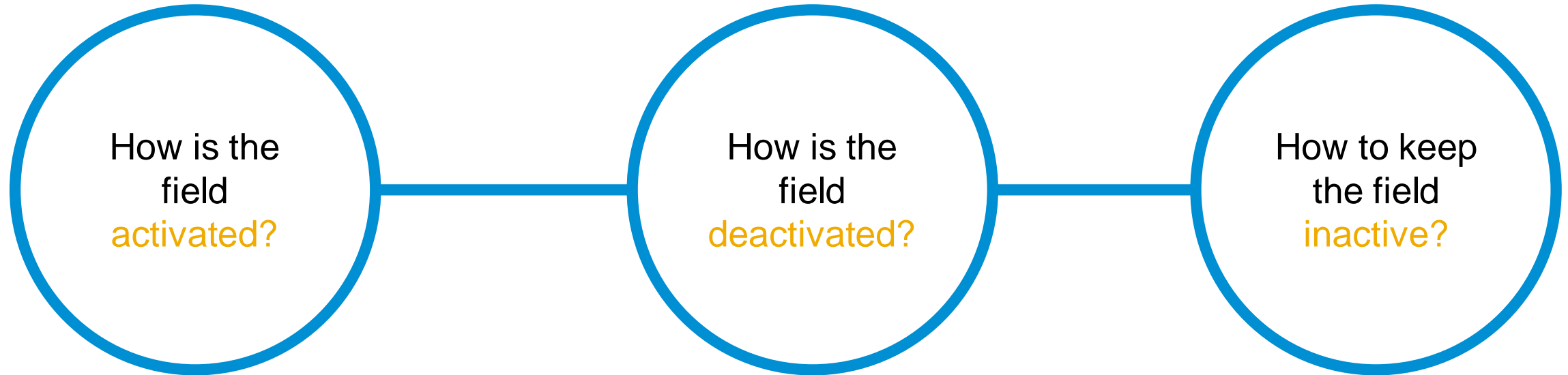


“The Somebody Else's Problem field... relies on people's natural predisposition not to see anything they don't want to, weren't expecting, or can't explain”

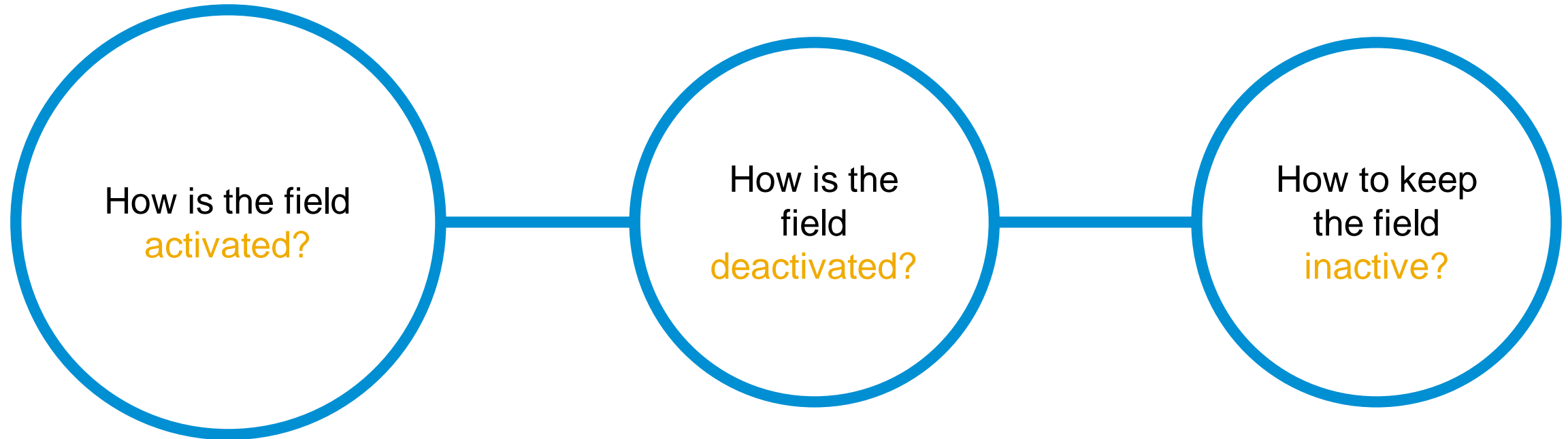
Douglas Adams

Life, the Universe and Everything

Addressing the “Somebody Else’s Problem” (SEP) field



Addressing the “Somebody Else’s Problem” (SEP) field



QAs were the messengers making
any failure your problem

@unremarkableQA



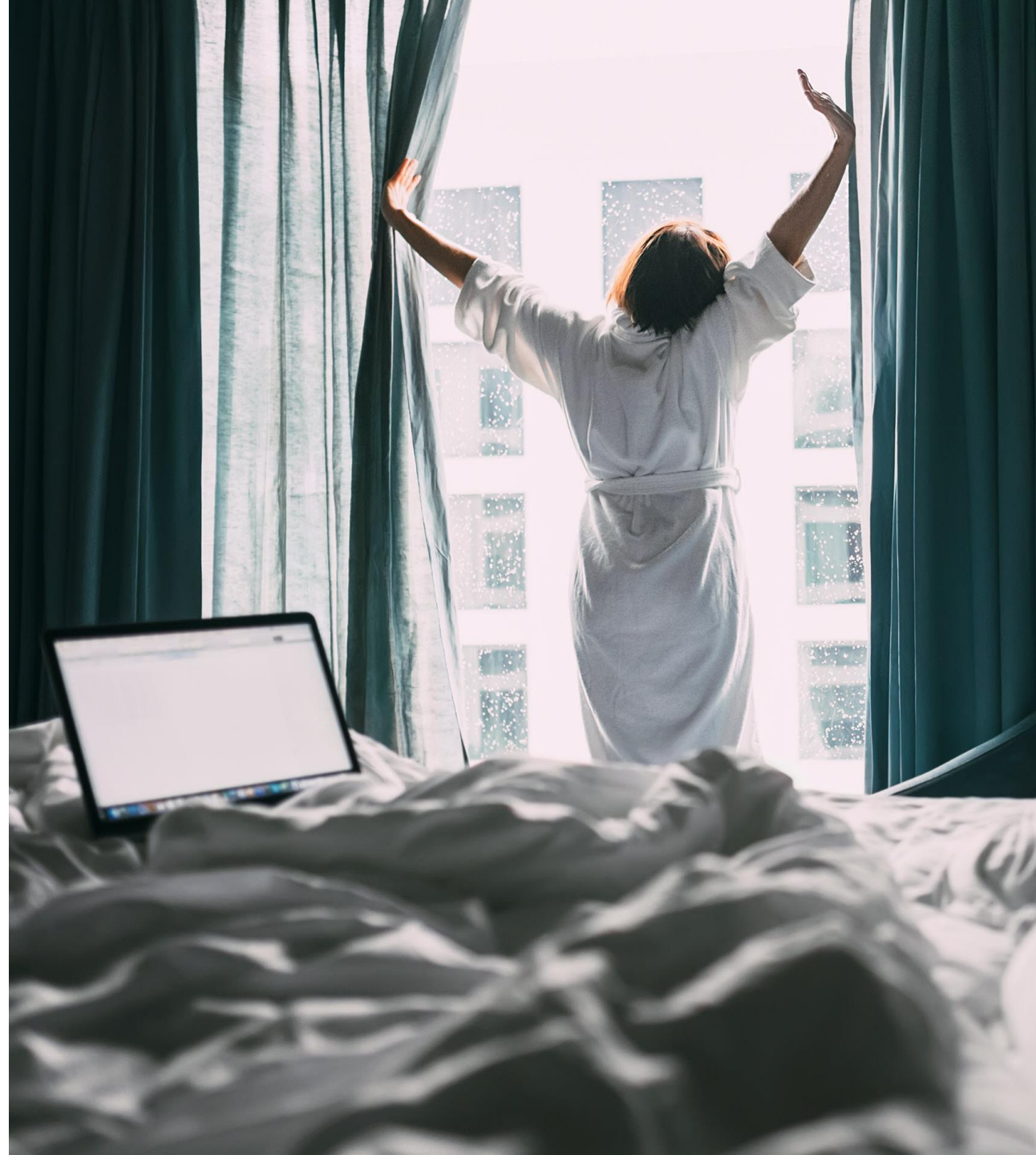
QAs didn't trust developers to make failures their problem

@unremarkableQA

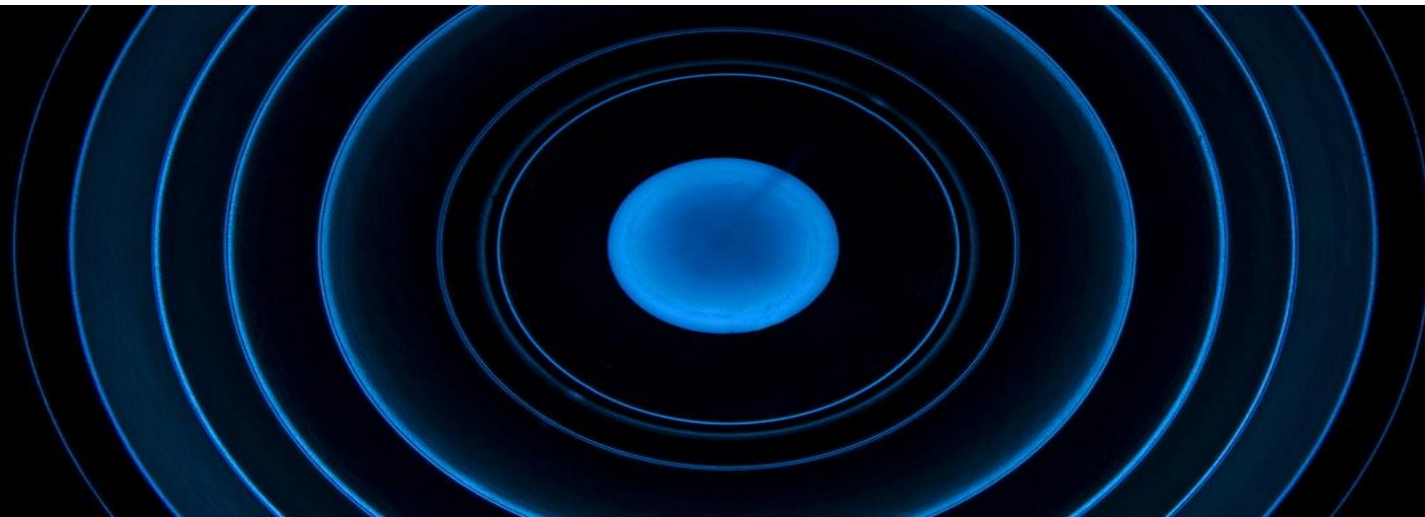
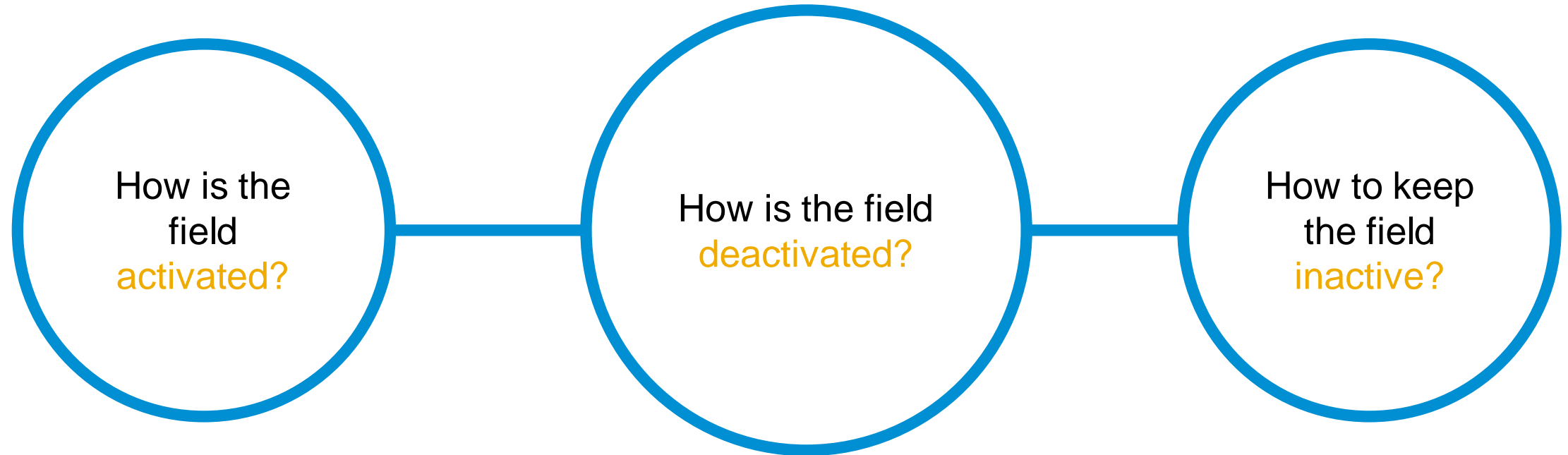


Unless personally assigned,
developers believed that pipeline step
failures were somebody else's
problem

@unremarkableQA



Addressing the “Somebody Else’s Problem” (SEP) field



How is the field **deactivated**?

“Keep only those things that speak to the heart, and discard items that no longer spark joy.”

Marie Kondo

Remove
things
**you don't
want to
see**

Remove
things
**you don't
expect to
see**

Remove
things
**you can't
explain**



How is the field **deactivated**?

“Keep only those things that speak to the heart, and discard items that no longer spark joy.”

Marie Kondo

Remove things
**you don't
want to see**

Remove
things
**you don't
expect to
see**

Remove
things
**you can't
explain**



Things you don't want to see



Flaky steps



Long running steps



Complicated-to-fix steps

“Place any non-deterministic test in a quarantined area. (But fix quarantined tests quickly.)”

Martin Fowler (@martinfowler)

[Eradicating Non-Determinism in Tests \(2011\)](#)

Things you don't want to see



Flaky steps

Quarantine, stabilise or
remove



Long running steps



Complicated-to-fix steps

Things you don't want to see



Flaky steps

Quarantine, stabilise or
remove



Long running steps

Parallelise to save time



Complicated-to-fix steps

**“A 30 minute code change
took 2 weeks to get the
acceptance tests working”**

Sarah Wells (@sarahjwells)

[Quality for ‘Cloud Natives’: what changes when your systems are complexed and distributed \(2019\)](#)

“Introduce synthetic monitoring. This replaced our acceptance tests”

Sarah Wells (@sarahjwells)

[Quality for ‘Cloud Natives’: what changes when your systems are complexed and distributed \(2019\)](#)

Things you don't want to see



Flaky steps

Quarantine, stabilise or
remove



Long running steps

Parallelise to save time



Complicated-to-fix steps

Simplify or replace

How is the field **deactivated**?

“Keep only those things that speak to the heart, and discard items that no longer spark joy.”

Marie Kondo

Remove things
**you don't
want to
see**

Remove things
**you don't
expect to see**

Remove things
**you can't
explain**



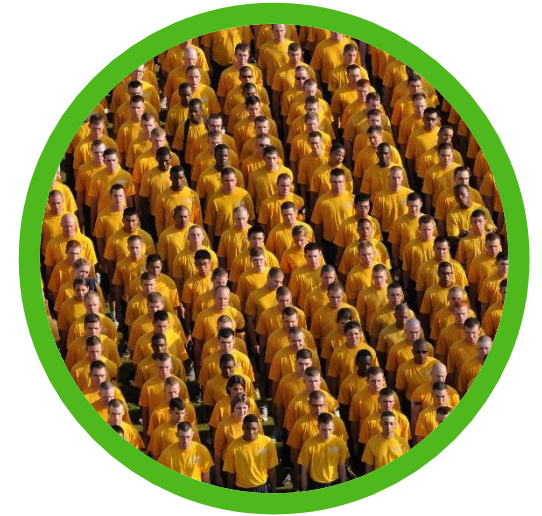
Things you don't expect to see



Unresponsive external services



New unannounced steps



Being the n-th person notified of the failed step

Things you don't expect to see

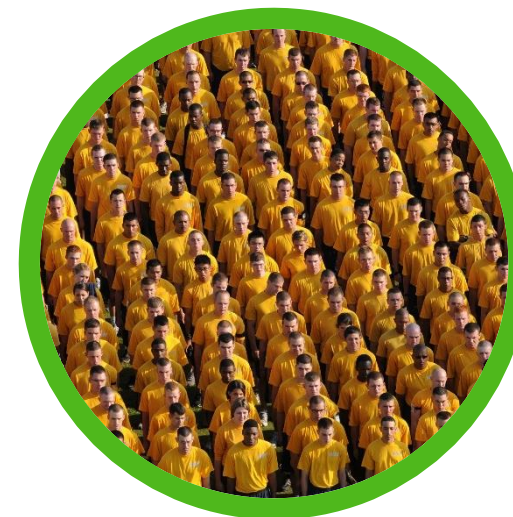


Unresponsive external services

Ping external services before starting the pipeline



New unannounced steps



Being the n-th person notified of the failed step

Things you don't expect to see



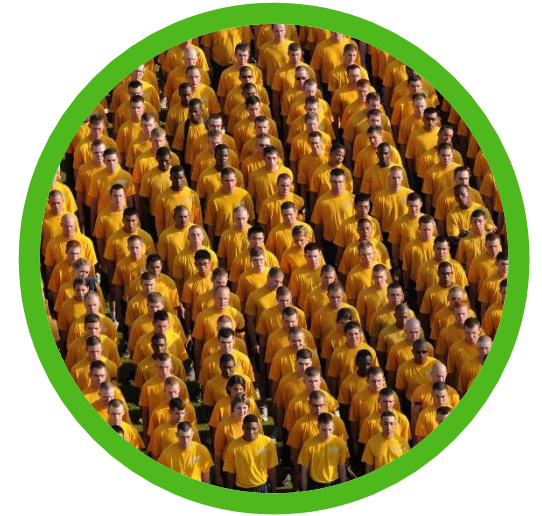
Unresponsive external services

Ping external services before starting the pipeline



New unannounced steps

Establish a new steps ritual



Being the n-th person notified of the failed step

Make transparent who is already working on a fix

	Build	Date ↑	Failure Duration	Status	Description
	Backend_Site_Generation_DEV #1129	17 hr	4 days 17 hr	unclaimed	
	Test_Smoketest_DEV #5019	1 day 5 hr	3 days 5 hr	john.doe@acme.com	
	Test_Acceptance_DEV #4064	1 day 7 hr	1 day 7 hr	unclaimed	

Icon: [S](#) [M](#) [L](#)

[Legend](#)  [RSS for all](#)  [RSS for failures](#)  [RSS for just latest builds](#)

[Claim – Jenkins plugin](#)

Things you don't expect to see



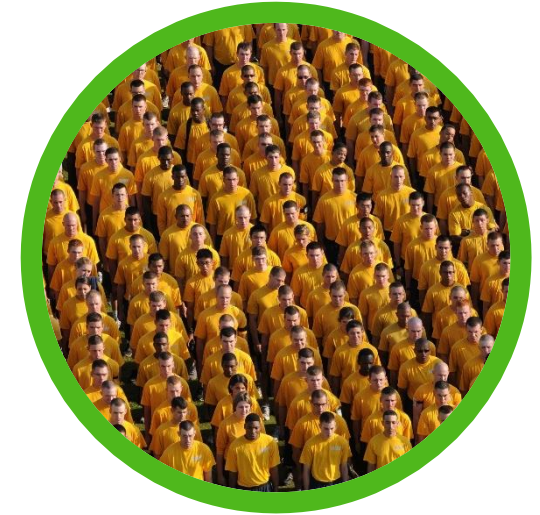
Unresponsive external services

Ping external services before starting the pipeline



New unannounced steps

Establish a new steps ritual



Being the n-th person notified of the failed step

Create public-awareness of who is resolving the failure

How is the field **deactivated**?

“Keep only those things that speak to the heart, and discard items that no longer spark joy.”

Marie Kondo

Remove things
**you don't
want to
see**

Remove things
**you don't
expect to
see**

Remove things
**you can't
explain**



Things you can't explain



Steps with unclear purpose




Steps with unclear failure implications







Steps with unclear fix deadlines


A good example of a bad step description

Project Test_Backend_Scripts_DEV

 [Recent Changes](#)

 Disk Usage

 Directory icon Job	597 MB
 Directory icon All builds	656 KB
 Directory icon Locked builds	-
 Directory icon All workspaces	596 MB
 Directory icon Slave workspaces	596 MB
 Directory icon Non-slave workspaces	-

 [Latest Test Result \(no failures\)](#)

Things you can't explain



Steps with unclear purpose

Make the value and information of the step visible

Steps with unclear failure implications

Steps with unclear fix deadlines

Things you can't explain



Steps with unclear purpose

Make the value and information of the step visible

Steps with unclear failure implications

Make the impact and the reasons of the failure transparent

Steps with unclear fix deadlines

Things you can't explain



Steps with unclear purpose

Make the value and information of the step visible

Steps with unclear failure implications

Make the impact and the reasons of the failure transparent

Steps with unclear fix deadlines

Set a time limit for the fixes

A better example of a step description

Name	Security_tests
Description	Checks that APIs are secure

VS

Name	Sec_API_Perm_tests
Description	Checks that only roles with the right permissions can access the APIs
Runs	On commit
Severity	Super high – Legal problems if broken
Threshold	1 broken test = no deployment
Wiki	//acme/our-wiki/api-perm-tests
Contacts	jane.doe@acme.com (CET)

Things you can't explain



Steps with unclear purpose

Make the value and information of the step visible

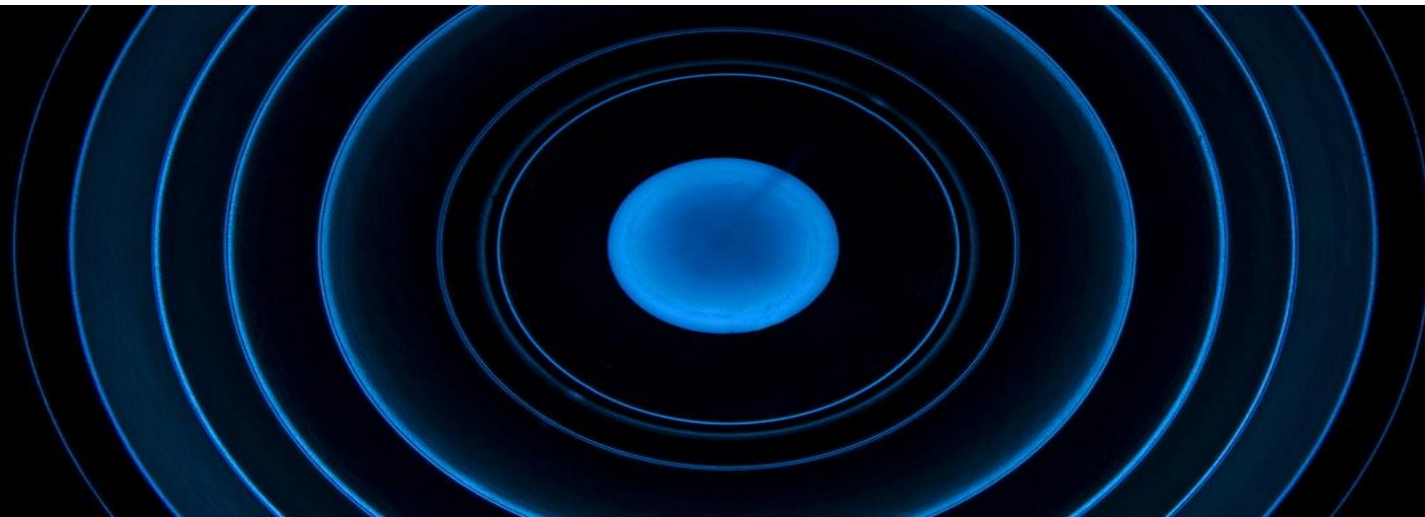
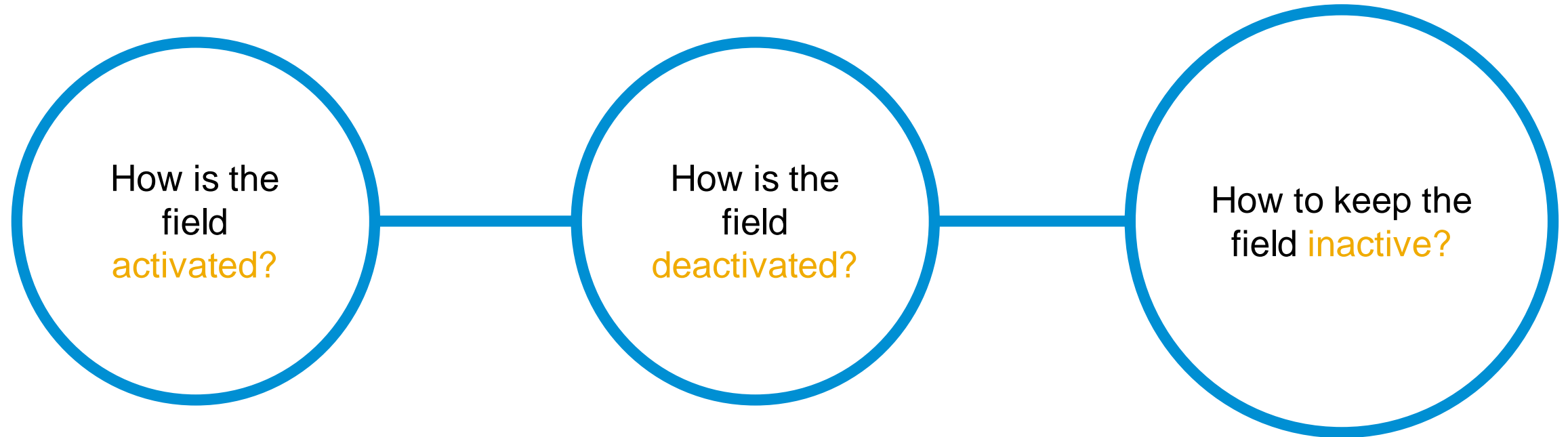
Steps with unclear failure implications

Make the impact and the reasons of the failure transparent

Steps with unclear fix deadlines

Set a time limit for the fixes

Addressing the “Somebody Else’s Problem” (SEP) field



Automatically assign someone as the owner of a failed step using a deterministic rule.



The owner of a failed step should be responsible for the solution, not necessarily the fix.

@unremarkableQA



Assign an owner of the quality of the pipeline rather than the pipeline steps



Educate and enable the whole team
to become owner of the quality of the
pipeline

@unremarkableQA

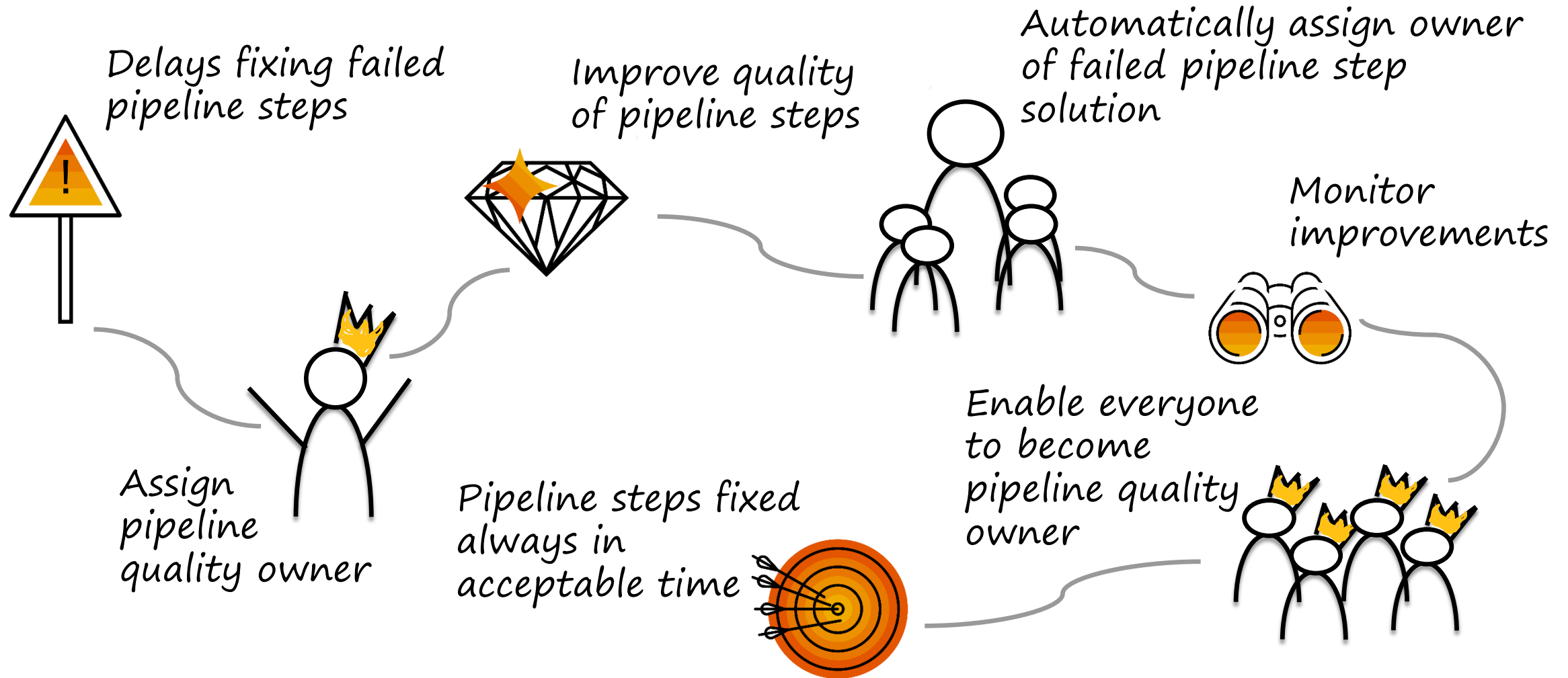


Be prepared for a rough start or
maybe not...

@unremarkableQA



Turning the quality of your deployment pipeline into a team task



Thank you.

Contact information:

Areti Panou

Cloud Quality Coach



@unremarkableQA



unremarkabletester.com



areti.panou@sap.com