

3 amazing opportunities in platform engineering which totally aren't screw ups because this is America

Steve Smith

Global VP of Technology at Scale



Hello!

Global VP of Technology at Scale

Helping organisations to build capabilities, to accelerate business outcomes at scale - many teams, many services

Platform leadership experiences

- 5 teams/20 services -> 8 teams/70 services in 3 yrs
- 10 teams/50 services -> 60 teams/600 services in 2 yrs
- 1 team/1 service -> 40 teams, 120 services in 2.5 yrs

Books

[Measuring Continuous Delivery](#) and [Build Quality In](#)



Steve Smith



3 ways you're screwing up platform engineering



What is platform engineering

What is platform engineering

Not dead, just resting! How To Win at Maintenance Mode



Not dead, just resting!
How to win at
maintenance mode

Steve Smith
Head of Scale



Get Together
Go Faster



DOES-US | OCTOBER 3-5, 2023
#ITREVOLUTION #DOES23 @ITREVDOES



What is platform engineering

Paving a Secure Road at John Lewis & Partners

DEVOPS
ENTERPRISE
SUMMIT



Simon Skelton
Platform & Operations Manager
John Lewis & Partners

Chris Rutter
Principal Consultant
Equal Experts

Paving a Secure Road at
John Lewis & Partners

jlj JOHN LEWIS
PARTNERSHIP

JOHN LEWIS | WAITROSE

DOES23 Amsterdam | 15-17 May 2023
#DOES23 @ITREVDOES

Get Together
Go Faster



3 ways you're screwing up platform engineering



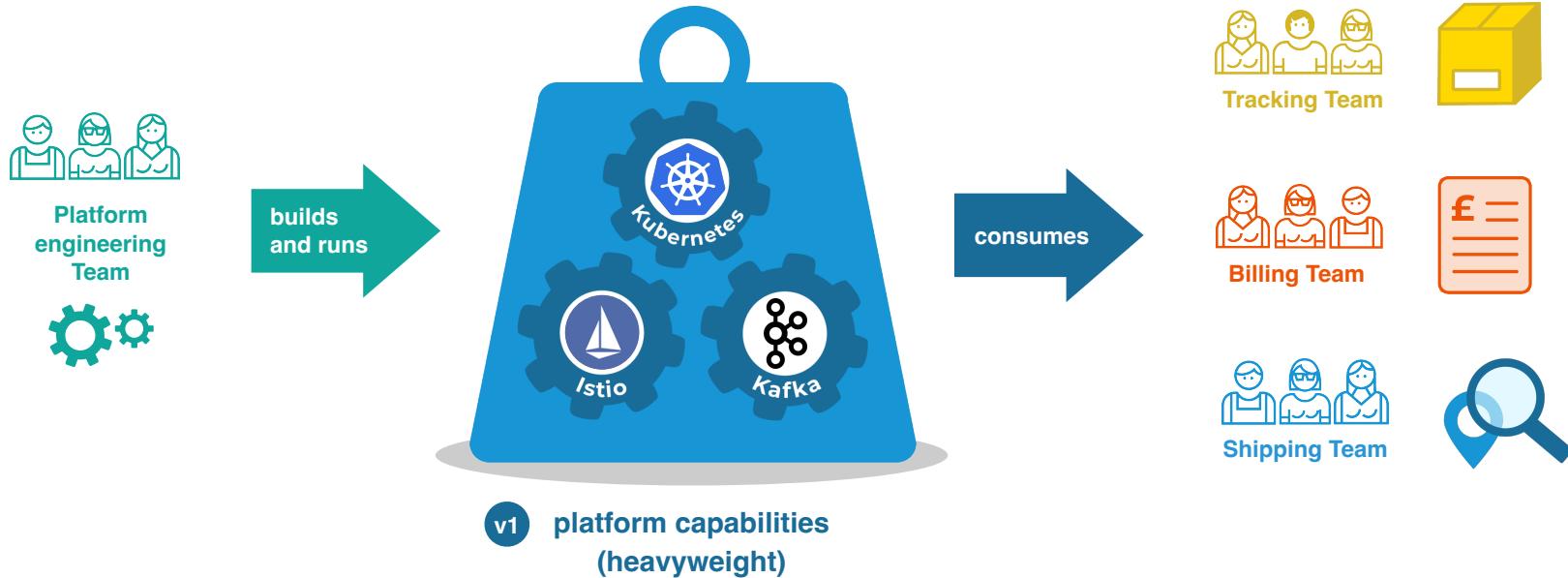
#1 Power tools

#2 Technology anarchy

#3 Ticketing hell

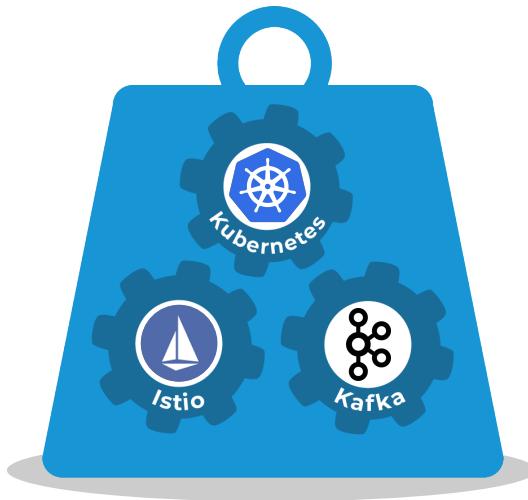
Screw up #1 - Power Tools

Kubernetes, Kafka, Istio or other power tools underpin your platform

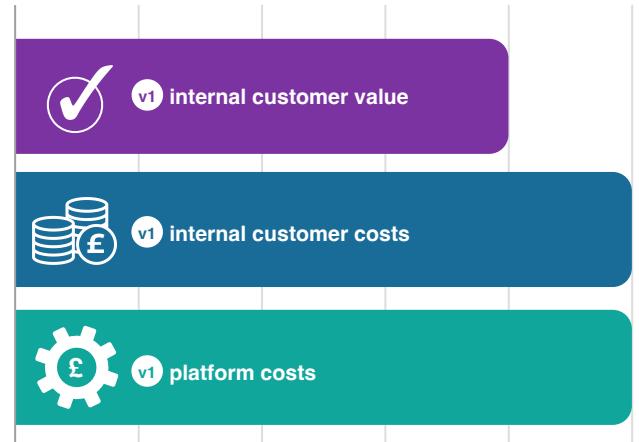


Screw up #1 - Power Tools

Kubernetes, Kafka, Istio or other power tools underpin your platform



v1 platform capabilities
(heavyweight)



Screw up #1 - Power Tools



“Some platform engineers wanted Kubernetes even though the delivery teams said they didn’t want it. Arguments went on for months, and it was really unproductive”

Platform lead, American
broadcaster



“The platform team wouldn’t upgrade Istio until they saw a Dutch company publish an upgrade blog. Teams didn’t want to use the platform, because innovation was too slow”

Tech lead, German
ecommerce

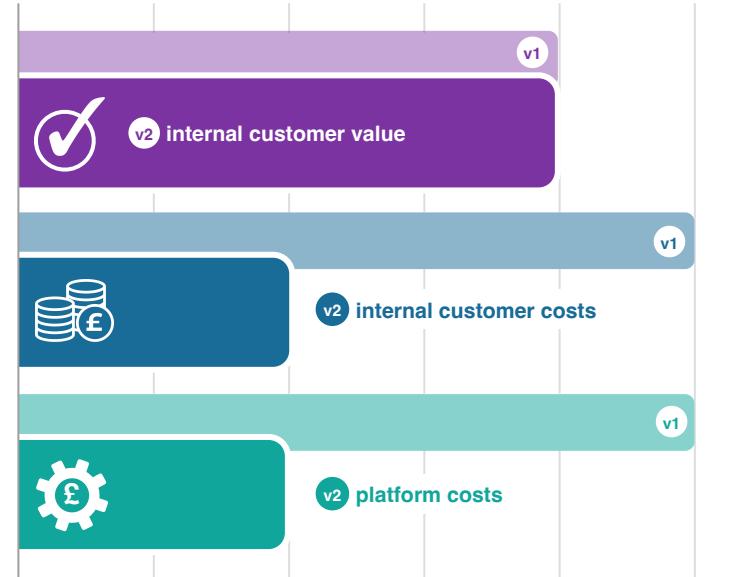
Screw up #1 - Power Tools

Lift and shift workloads from heavyweight to lightweight capabilities

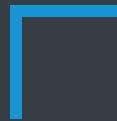


Screw up #1 - Power Tools

Lift and shift teams from heavyweight to lightweight capabilities



3 ways you're screwing up platform engineering



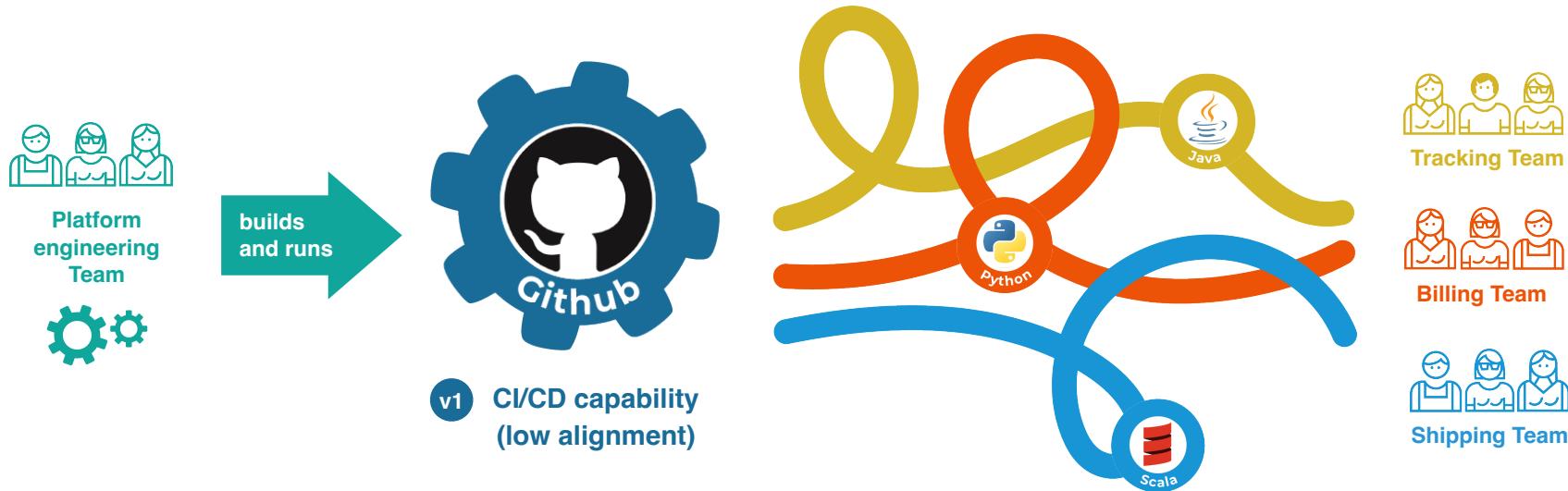
#1 Power tools

#2 Technology anarchy

#3 Ticketing hell

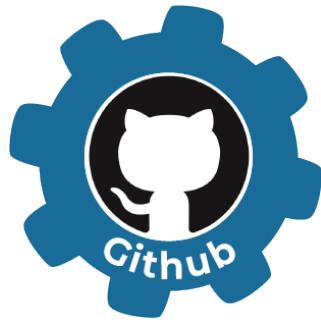
Screw up #2 - Technology anarchy

Your platform enables autonomy but doesn't offer technical alignment

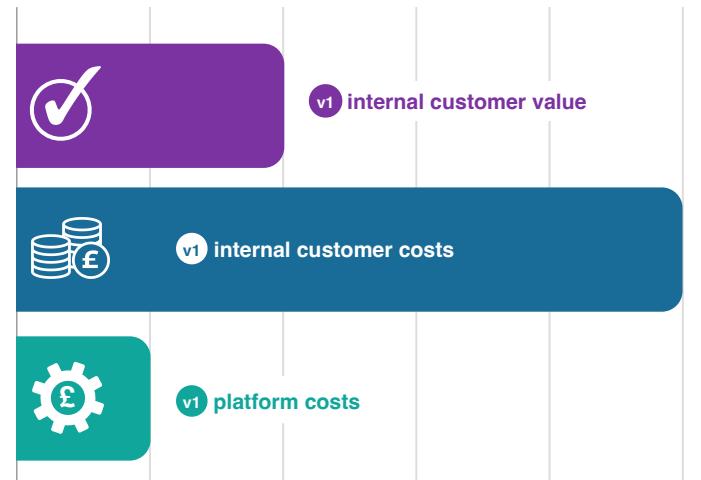
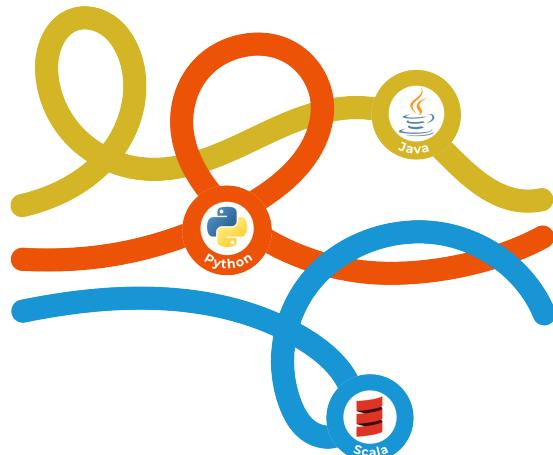


Screw up #2 - Technology anarchy

Your platform enables autonomy but doesn't offer technical alignment



v1 CI/CD capability
(low alignment)



Screw up #2 - Technology anarchy

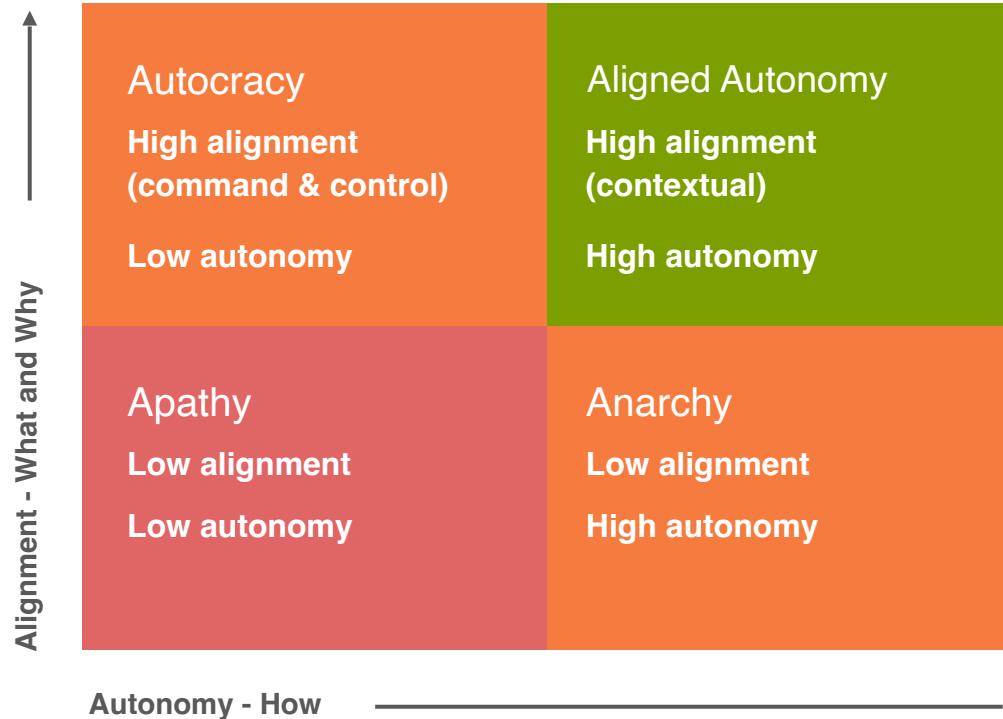
“We had 20 teams doing You Build It You Run It. But no product work was happening, because teams did GCP themselves, so that was 20 GCP workstreams each week”

Delivery manager,
British telco

“Our 10 teams migrated their messaging services from RabbitMQ to Kafka. But with no platform team, they had to do 10 different migrations themselves, and it took months”

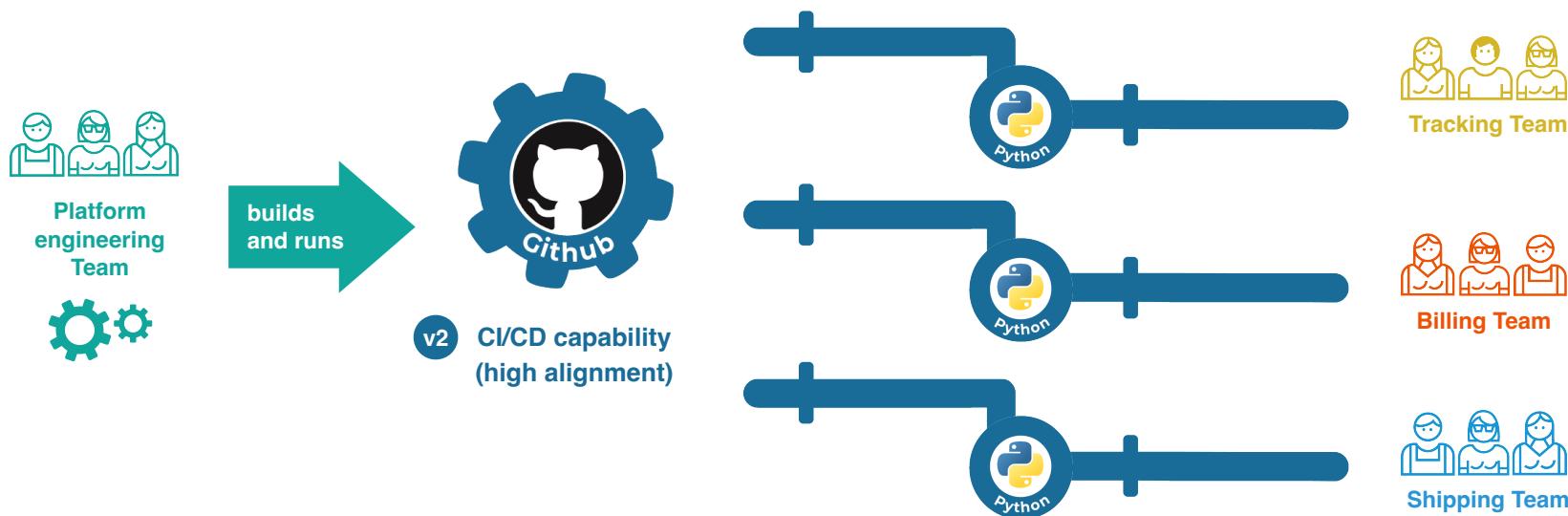
Head of engineering,
American retailer

Screw up #2 - Technology anarchy



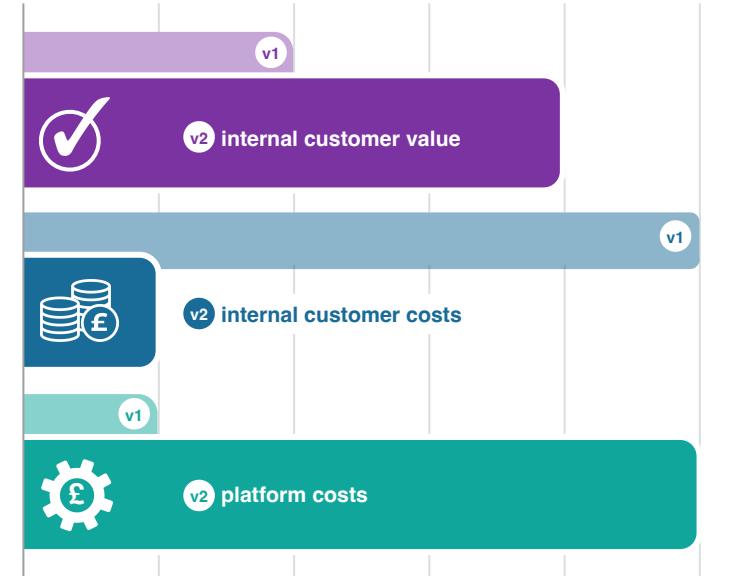
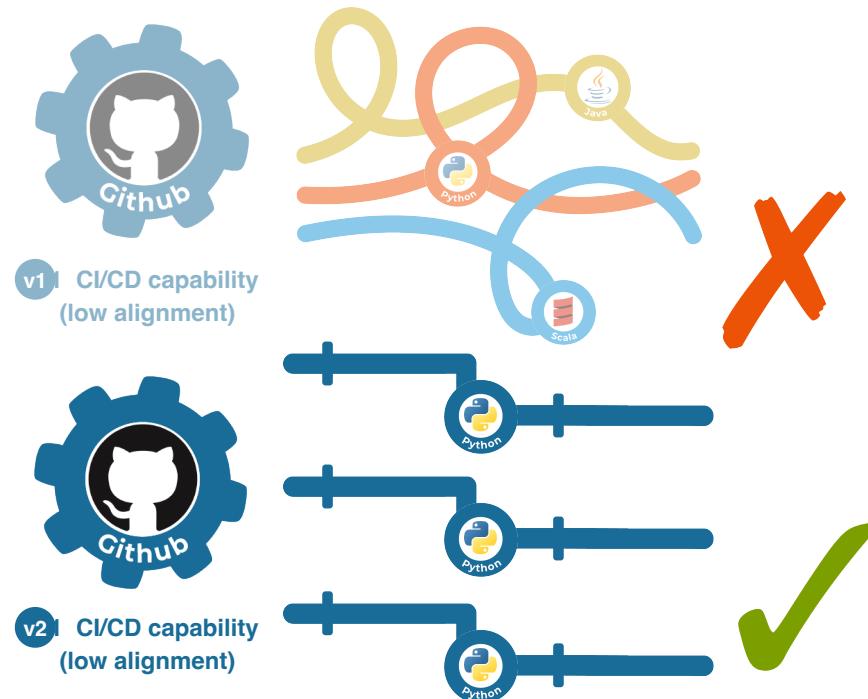
Screw up #2 - Technology anarchy

Build technical alignment into platform capabilities, as paved roads



Screw up #2 - Technology anarchy

Build technical alignment into platform capabilities, as paved roads



3 ways you're screwing up platform engineering



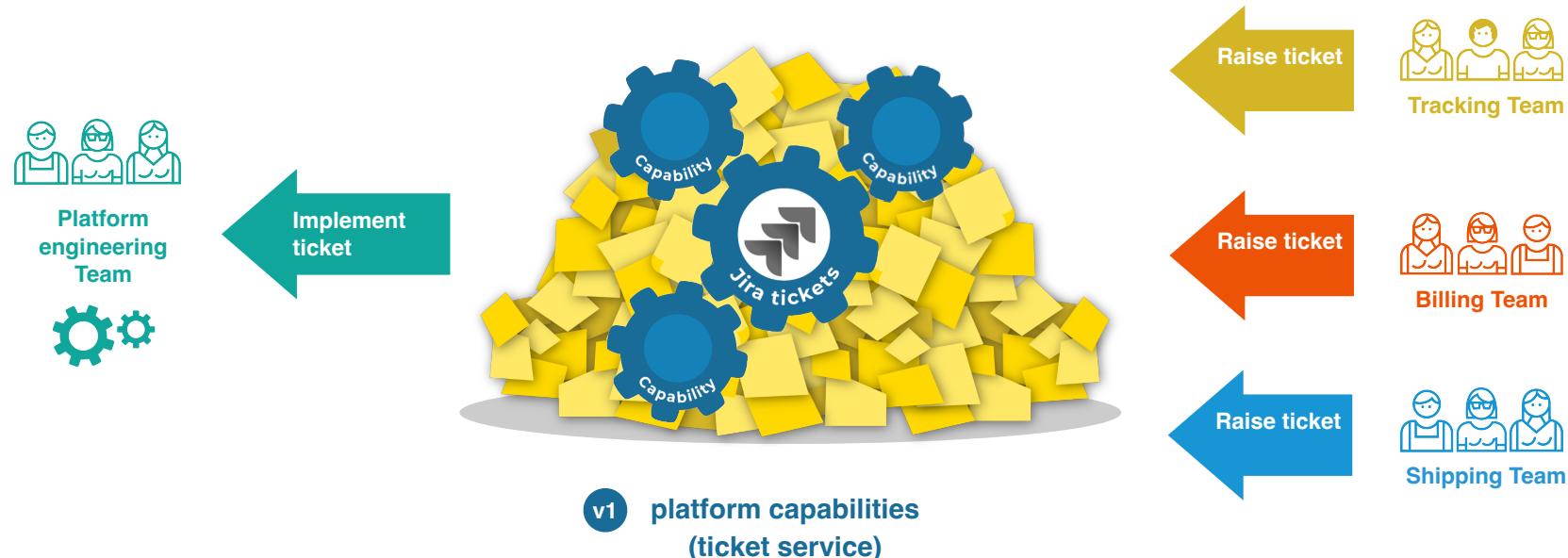
#1 Power tools

#2 Technology anarchy

#3 Ticketing hell

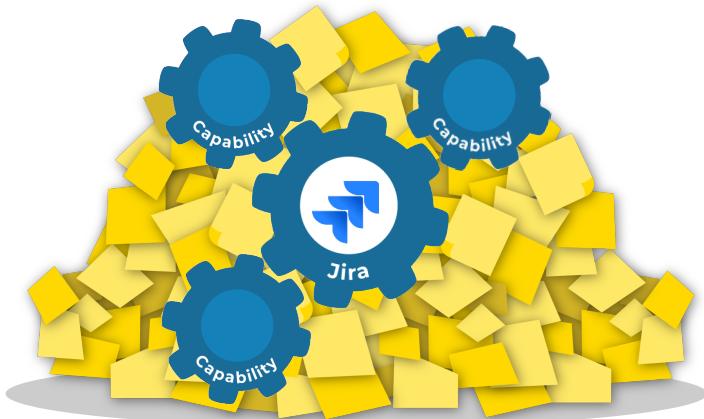
Screw up #3 - Ticketing hell

Your platform is an Ops service desk, team requests are handled as tickets

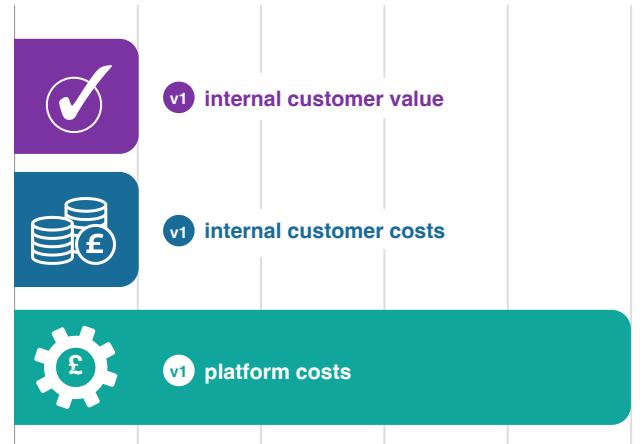


Screw up #3 - Ticketing hell

Your platform is an Ops service desk, team requests are handled as tickets



v1 platform capabilities
(ticket service)



Screw up #3 - Ticketing hell



“It was 11 weeks before my new starter platform ticket was completed. I couldn't do anything, and felt stressed, but my team told me it was normal”

Tech lead

Dutch bank

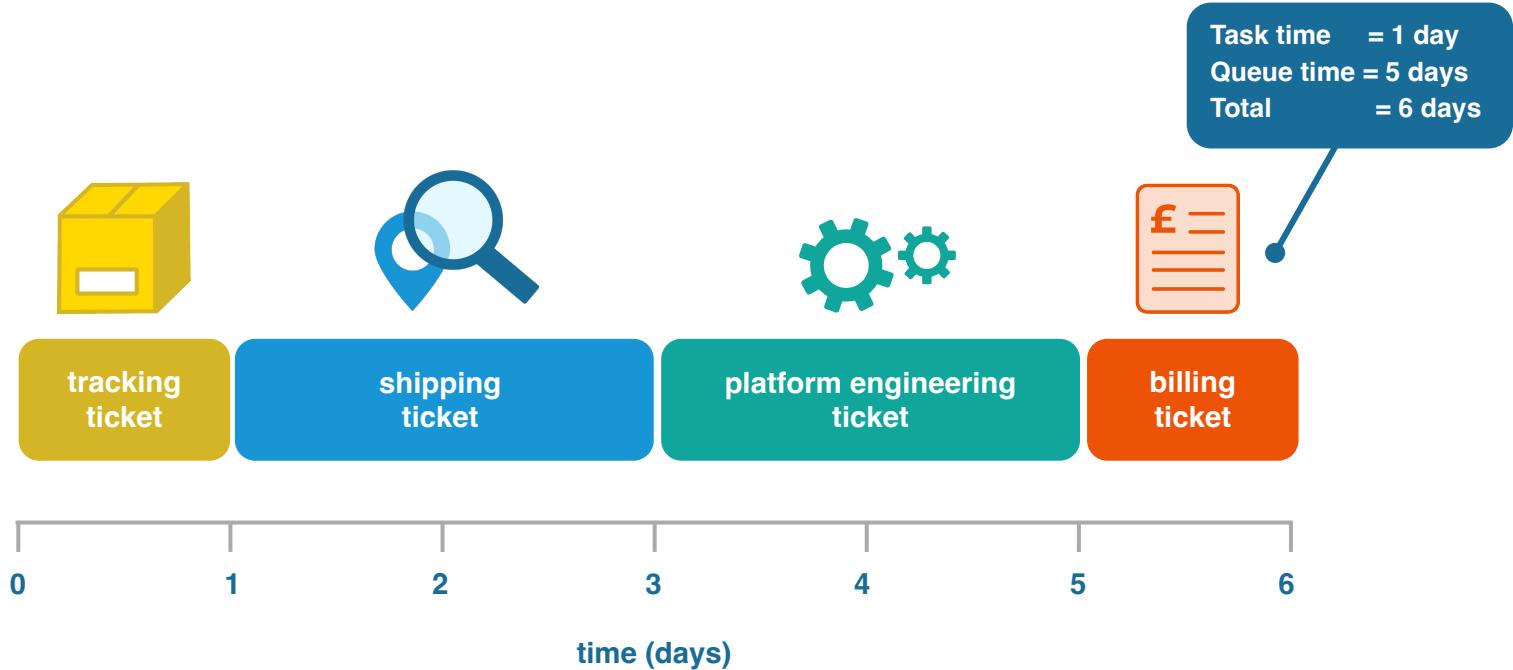


“I have to create a ticket for a dev deploy, another for a test deploy, another for prod, teams create so much demand that our platform team can't ever keep up”

Delivery manager

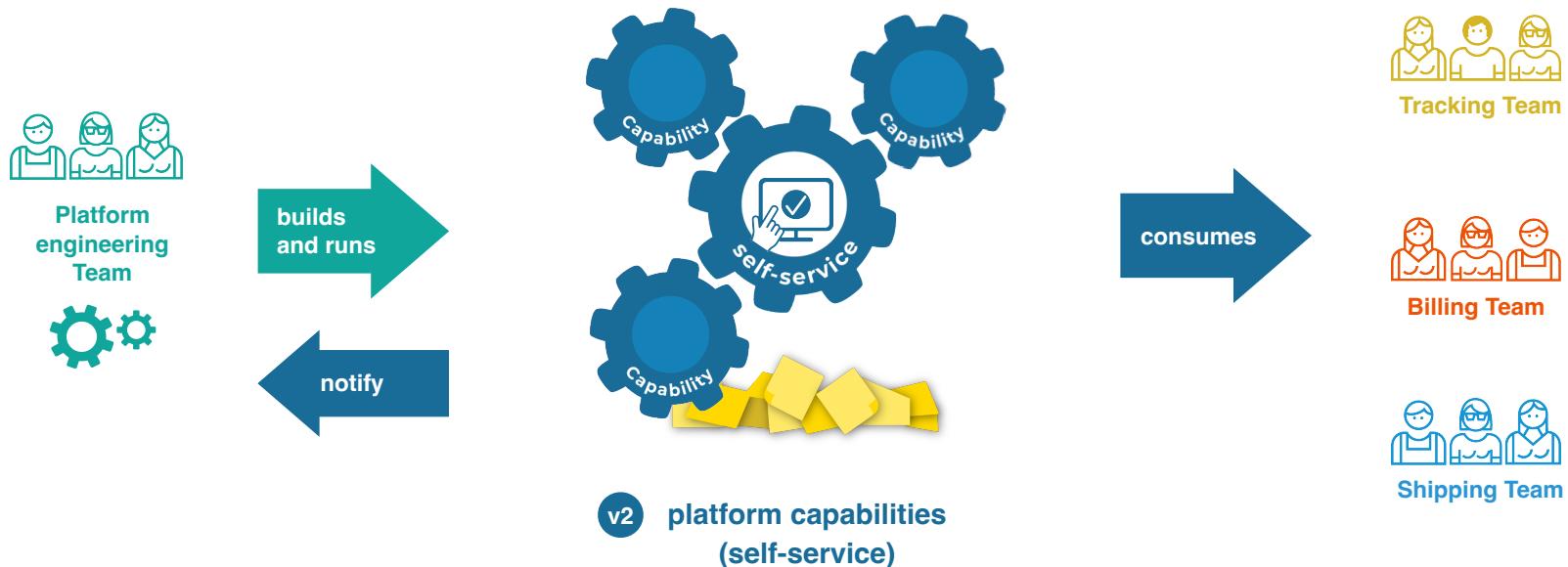
Australian telco

Screw up #3 - Ticketing hell



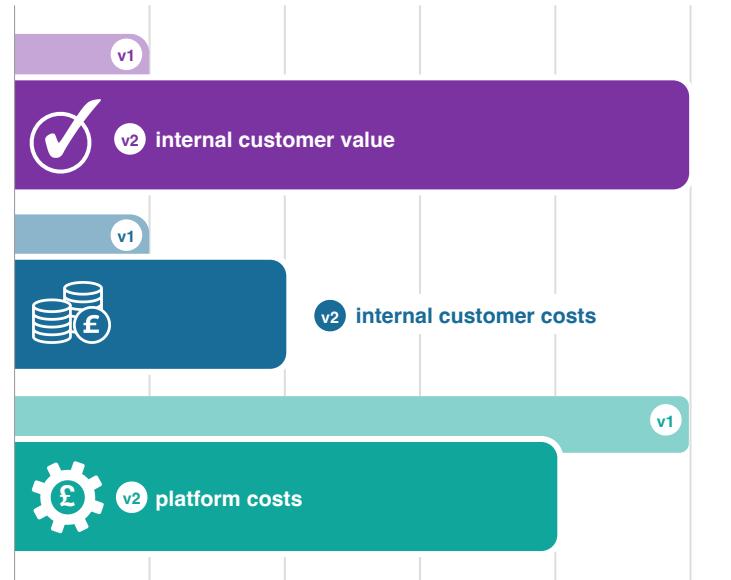
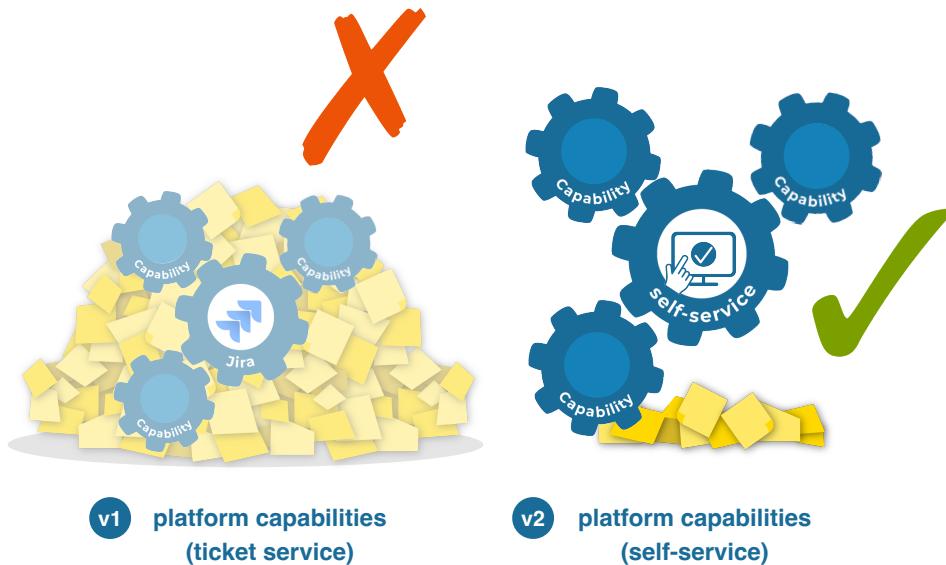
Screw up #3 - Ticketing hell

Turn high demand tasks into self-service paved roads



Screw up #3 - Ticketing hell

Turn high demand tasks into self-service paved roads



3 ways you're screwing up platform engineering

Takeaways

Takeaways

Platform engineering needs a **product mindset**

- it **isn't** a project
- it **isn't** gatekeeping
- is **isn't** build it and they will come

Increase internal customer value, reduce internal customer costs, manage platform costs



Help I'm looking for

- Have you seen power tools, technology anarchy, or ticketing hell in platform engineering?
- What were their leading indicators?
- Were you able to escape?



3 amazing opportunities in platform engineering which totally aren't screw ups because this is America

Steve Smith

Global VP of Technology at Scale

