Understanding why people resist using your platform

Coté - February 6thth, 2024

We all know that

Changing organizations fails 70% of the time.

Sources: ”Mind-sets matter in transformations,” McKinsey, 2019, many other sources; “How Studying Organizational Change Lost Its Way,” Journal of Change Management, Mark Hughes, 2022.

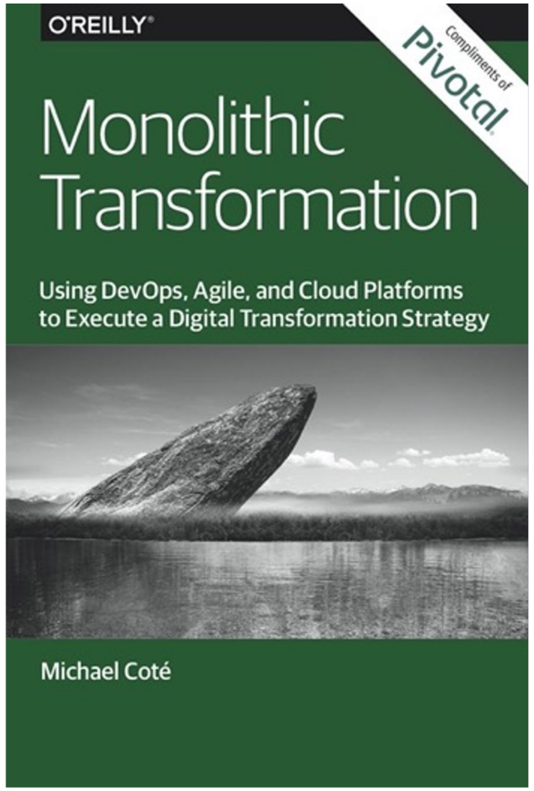
Actually,

We have no idea how frequently changing organizations succeeds or fails.

Sources: "Mind-sets matter in transformations," McKinsey, 2019, many other sources; "How Studying Organizational Change Lost Its Way," Journal of Change Management, Mark Hughes, 2022.

Coté

https://newsletter.cote.io/ | cote@broadcom.com

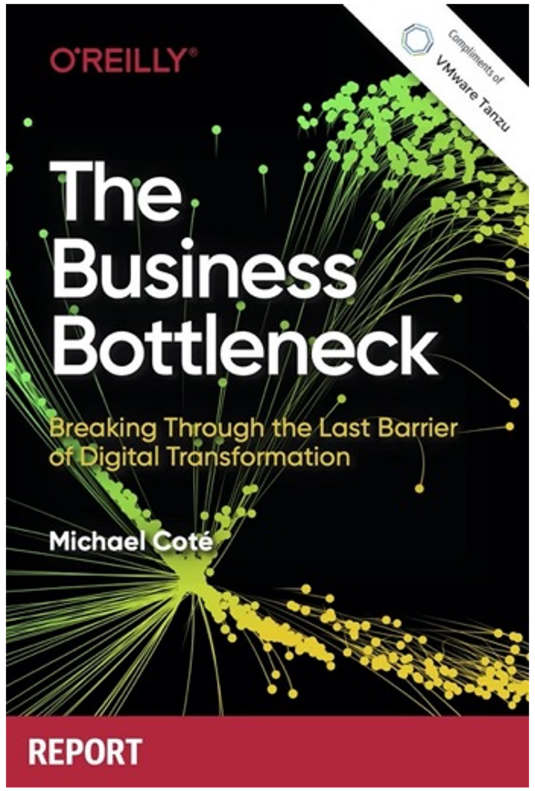

O'REILLY

Compliments of Pivotal

Monolithic Transformation

Using DevOps, Agile, and Cloud Platforms to Execute a Digital Transformation Strategy

Michael Coté


O'REILLY

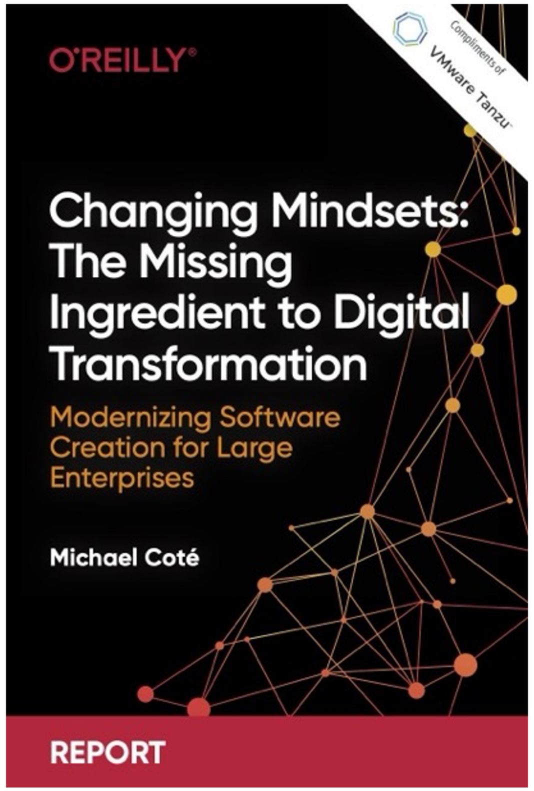
Compliments of VMware Tanzu

The Business Bottleneck

Breaking Through the Last Barrier of Digital Transformation

Michael Coté

REPORT


O'REILLY

Compliments of VMware Tanzu

Changing Mindsets: The Missing Ingredient to Digital Transformation

Modernizing Software Creation for Large Enterprises

Michael Coté

REPORT

Tanzu Talk

SOFTWARE DEFINED TALK

RedMonk DELL 451 Research Pivotal vmware Tanzu by Broadcom

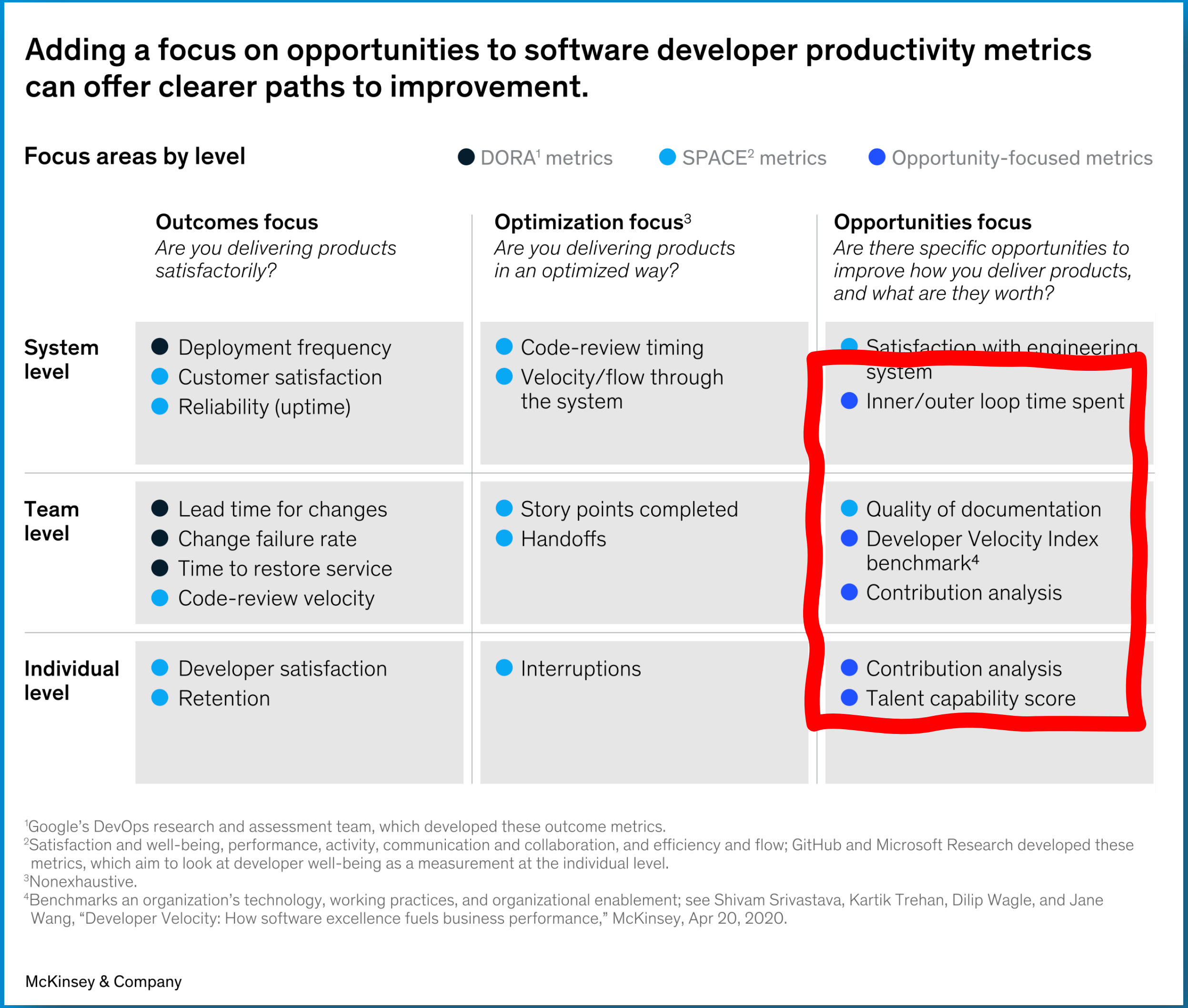
The Wonderful World of Management

"People"

“This is a 1 ½ CIO Job.”

Sources: “Fortune 500 C-Suite Snapshot: Profiles in Functional Leadership,” SpencerStuart, 2023 (analysis done as of June 30, 2023).

Management & workers often have different incentives & motivations



Adding a focus on opportunities to software developer productivity metrics can offer clearer paths to improvement.

Focus areas by level ● DORA[1] metrics ◍ SPACE[2] metrics 🔵 Opportunity-focused metrics

	Outcomes focus *Are you delivering products satisfactorily?*	**Optimization focus[3]** *Are you delivering products in an optimized way?*	**Opportunities focus** *Are there specific opportunities to improve how you deliver products, and what are they worth?*
System level	● Deployment frequency ◍ Customer satisfaction ◍ Reliability (uptime)	◍ Code-review timing ◍ Velocity/flow through the system	◍ Satisfaction with engineering system 🔵 Inner/outer loop time spent
Team level	● Lead time for changes ● Change failure rate ● Time to restore service ◍ Code-review velocity	◍ Story points completed ◍ Handoffs	◍ Quality of documentation 🔵 Developer Velocity Index benchmark[4] 🔵 Contribution analysis
Individual level	◍ Developer satisfaction ◍ Retention	◍ Interruptions	🔵 Contribution analysis 🔵 Talent capability score

[1]Google's DevOps research and assessment team, which developed these outcome metrics.
[2]Satisfaction and well-being, performance, activity, communication and collaboration, and efficiency and flow; GitHub and Microsoft Research developed these metrics, which aim to look at developer well-being as a measurement at the individual level.
[3]Nonexhaustive.
[4]Benchmarks an organization's technology, working practices, and organizational enablement; see Shivam Srivastava, Kartik Trehan, Dilip Wagle, and Jane Wang, "Developer Velocity: How software excellence fuels business performance," McKinsey, Apr 20, 2020.

McKinsey & Company

Sources: "Great Attrition' or 'Great Attraction'? The choice is yours," Aaron De Smet, Bonnie Dowling, Marino Mugayar-Baldocchi, Bill Schaninger, McKinsey, Sep 2021; "Yes, you can measure software developer productivity," Chandra Gnanasambandam, Martin Harrysson, Alharith Hussin, Jason Keovichit, and Shivam Srivastava, McKinsey, August, 2023. "The SPACE of Developer Productivity," March, 2021 . See also further commentary from Coté.

Happiness is not a business outcome

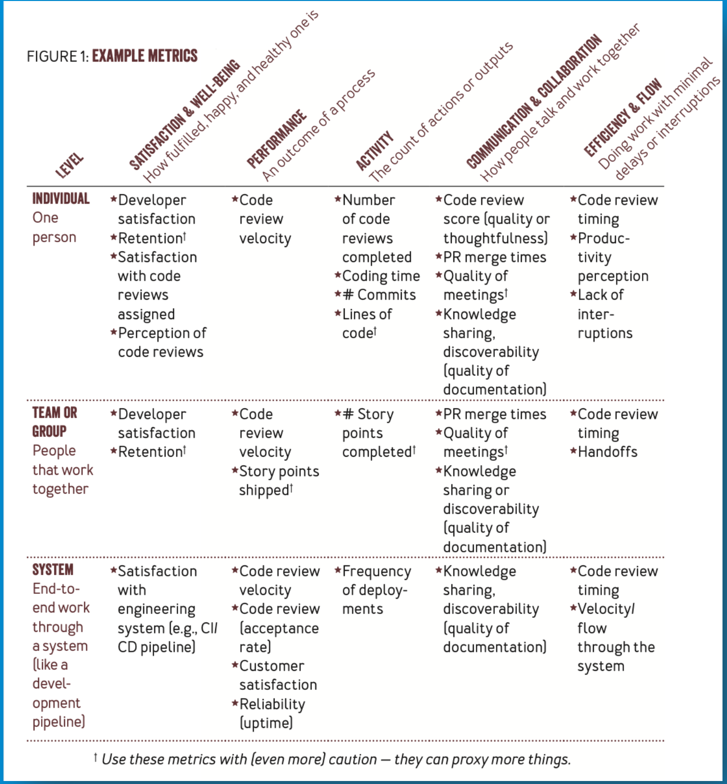


FIGURE 1: **EXAMPLE METRICS**

LEVEL	SATISFACTION & WELL-BEING How fulfilled, happy, and healthy one is	PERFORMANCE An outcome of a process	ACTIVITY The count of actions or outputs	COMMUNICATION & COLLABORATION How people talk and work together	EFFICIENCY & FLOW Doing work with minimal delays or interruptions
INDIVIDUAL One person	★Developer satisfaction ★Retention[†] ★Satisfaction with code reviews assigned ★Perception of code reviews	★Code review velocity	★Number of code reviews completed ★Coding time ★# Commits ★Lines of code[†]	★Code review score (quality or thoughtfulness) ★PR merge times ★Quality of meetings[†] ★Knowledge sharing, discoverability (quality of documentation)	★Code review timing ★Produc-tivity perception ★Lack of inter-ruptions
TEAM OR GROUP People that work together	★Developer satisfaction ★Retention[†]	★Code review velocity ★Story points shipped[†]	★# Story points completed[†]	★PR merge times ★Quality of meetings[†] ★Knowledge sharing or discoverability (quality of documentation)	★Code review timing ★Handoffs
SYSTEM End-to-end work through a system (like a devel-opment pipeline)	★Satisfaction with engineering system (e.g., CI/CD pipeline)	★Code review velocity ★Code review (acceptance rate) ★Customer satisfaction ★Reliability (uptime)	★Frequency of deploy-ments	★Knowledge sharing, discoverability (quality of documentation)	★Code review timing ★Velocity/flow through the system

[†] *Use these metrics with (even more) caution — they can proxy more things.*

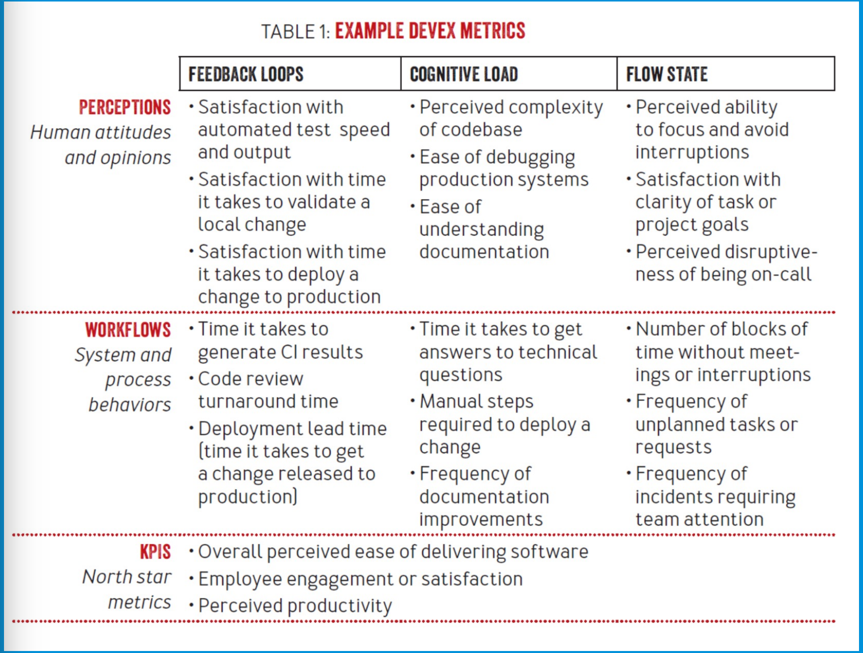
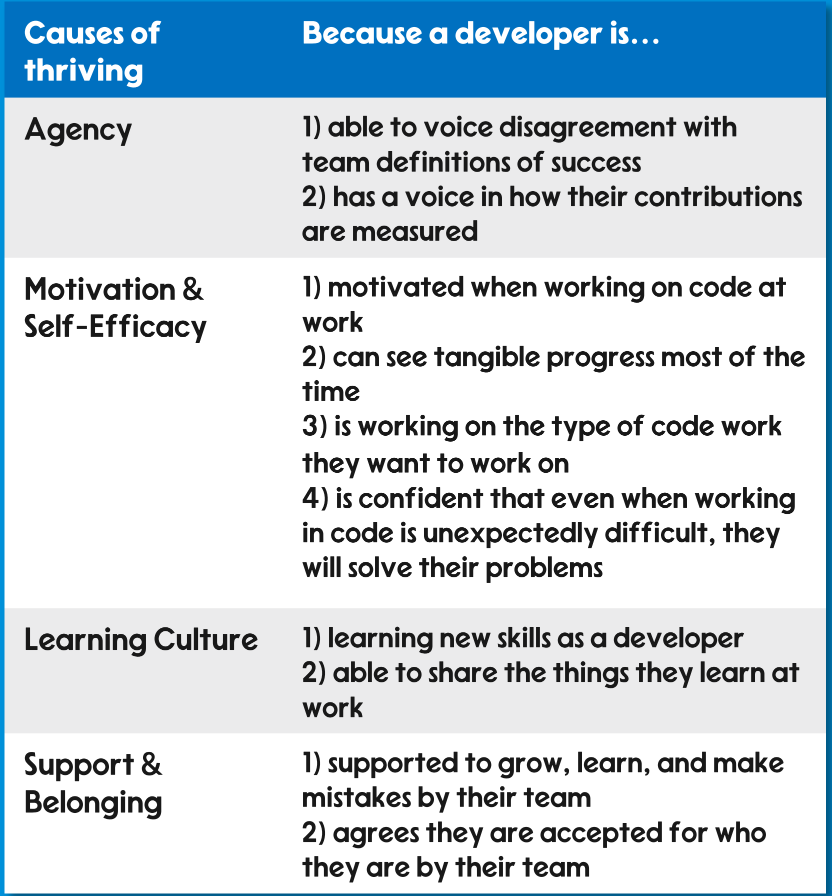


TABLE 1: **EXAMPLE DEVEX METRICS**

	FEEDBACK LOOPS	COGNITIVE LOAD	FLOW STATE
PERCEPTIONS *Human attitudes and opinions*	• Satisfaction with automated test speed and output • Satisfaction with time it takes to validate a local change • Satisfaction with time it takes to deploy a change to production	• Perceived complexity of codebase • Ease of debugging production systems • Ease of understanding documentation	• Perceived ability to focus and avoid interruptions • Satisfaction with clarity of task or project goals • Perceived disruptive-ness of being on-call
WORKFLOWS *System and process behaviors*	• Time it takes to generate CI results • Code review turnaround time • Deployment lead time (time it takes to get a change released to production)	• Time it takes to get answers to technical questions • Manual steps required to deploy a change • Frequency of documentation improvements	• Number of blocks of time without meet-ings or interruptions • Frequency of unplanned tasks or requests • Frequency of incidents requiring team attention
KPIS *North star metrics*	• Overall perceived ease of delivering software • Employee engagement or satisfaction • Perceived productivity		



Causes of thriving	Because a developer is…
Agency	1) able to voice disagreement with team definitions of success 2) has a voice in how their contributions are measured
Motivation & Self-Efficacy	1) motivated when working on code at work 2) can see tangible progress most of the time 3) is working on the type of code work they want to work on 4) is confident that even when working in code is unexpectedly difficult, they will solve their problems
Learning Culture	1) learning new skills as a developer 2) able to share the things they learn at work
Support & Belonging	1) supported to grow, learn, and make mistakes by their team 2) agrees they are accepted for who they are by their team

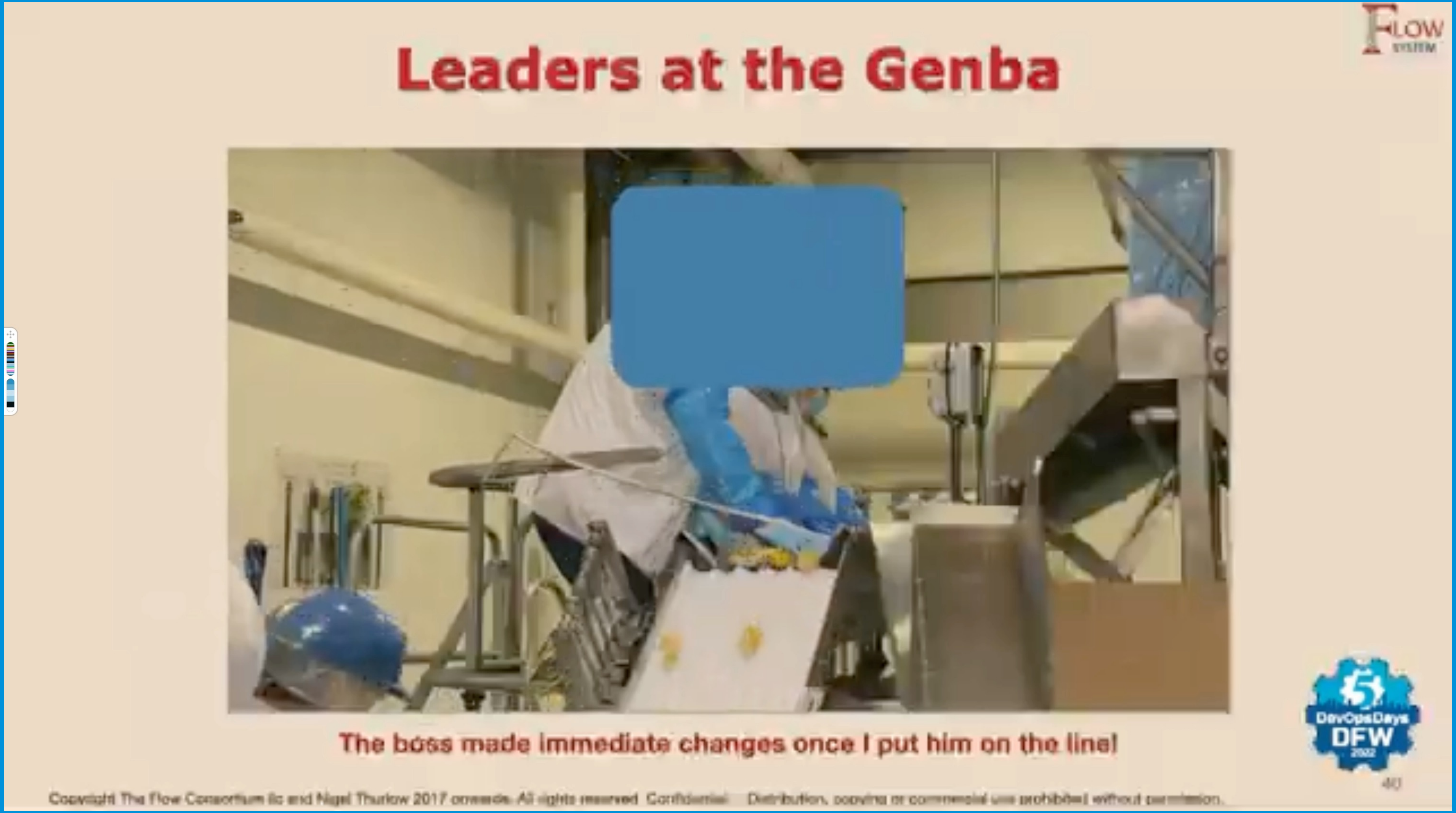
Source: "The SPACE of Developer Productivity," Nicole Forsgren, Margaret-Anne Storey, Chandra Maddila, Thomas Zimmermann, Brian Houck, and Jenna Butler, Jan-Feb, 2021; "DevEx: What Actually Drives Productivity," Abi Noda, Margaret-Anne Storey, Nicole Forsgren, Microsoft Research, Michaela Greiler, May 2023. Developer Thriving framework and studies, Cat Hicks et. al., March, 2023.

Management vs. workers often have different urgency & motivation to change

Exec's View	Work the Same	Transform!
Compensation	$	$$$$
Risk	HIGH	HIGH
Outcome	💣	👍

Staff's View	Work the Same	Transform!
Compensation	$	$\boxed{\$}$
Risk	$\boxed{\text{LOW}}$	HIGH
Outcome	👍	🤷‍♀️

The people who do the work (should) decide how to change the work



Leaders at the Genba

The boss made immediate changes once I put him on the line!

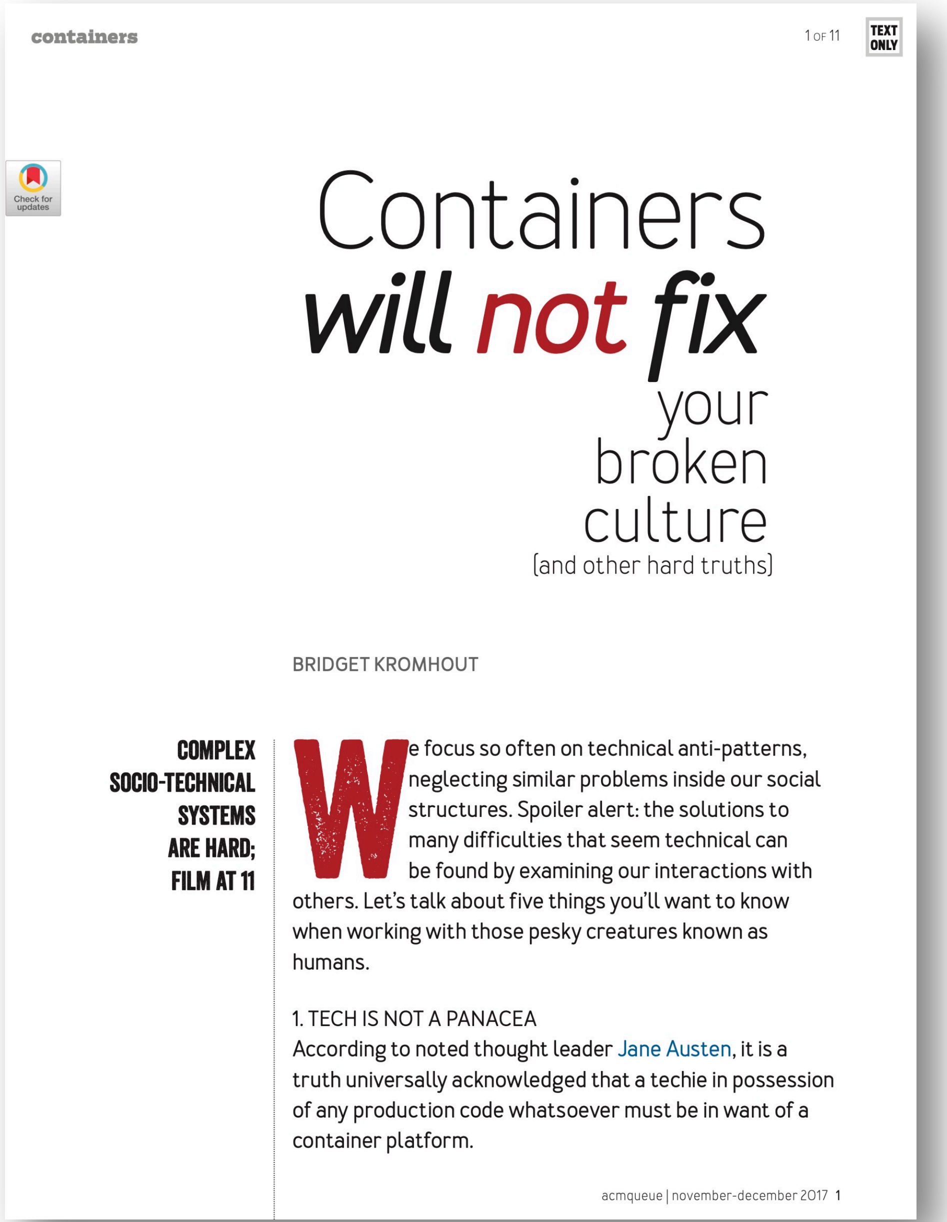
40

Sources: "DevOps is Enterprise Wide," Nigel Thurlow, DevOpsDays Dallas 2022.

“We believe that we need to reimagine banking to make banking simple, seamless and invisible to allow our customers to live more bank less.

increase developer productivity by building an abstraction layer to simplify Kubernetes complexity”.

Siew Choo Soh, DBS Bank



TEXT ONLY

Check for updates

Containers *will not fix* your broken culture (and other hard truths)

BRIDGET KROMHOUT

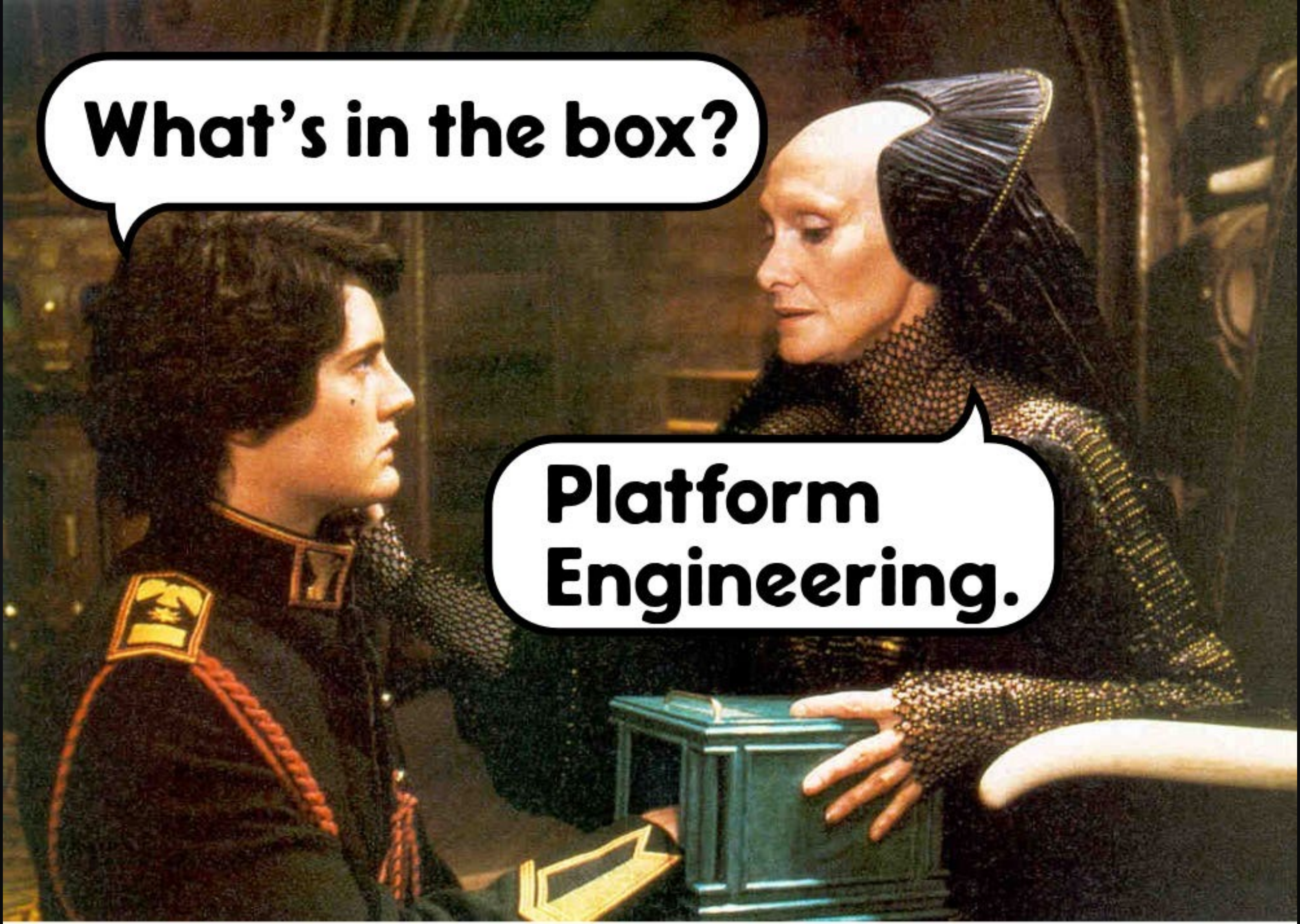
COMPLEX SOCIO-TECHNICAL SYSTEMS ARE HARD; FILM AT 11

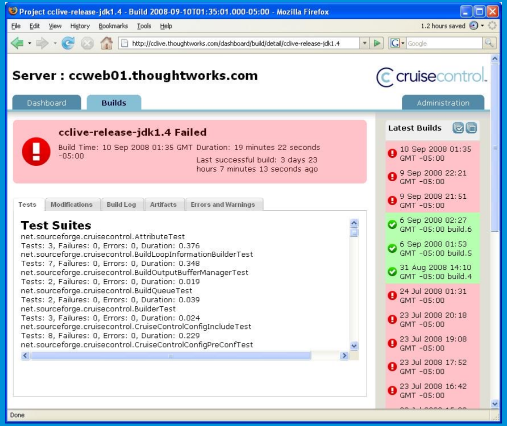
We focus so often on technical anti-patterns, neglecting similar problems inside our social structures. Spoiler alert: the solutions to many difficulties that seem technical can be found by examining our interactions with others. Let's talk about five things you'll want to know when working with those pesky creatures known as humans.

1. TECH IS NOT A PANACEA

According to noted thought leader Jane Austen, it is a truth universally acknowledged that a techie in possession of any production code whatsoever must be in want of a container platform.

Source: "Containers Will Not Fix Your Broken Culture," Bridget Kromhout, Dec 2017.


What's in the box?
Platform Engineering.


Project cclive-release-jdk1.4 - Build 2008-09-10T01:35:01.000-05:00 - Mozilla Firefox
File Edit View History Bookmarks Tools Help
1.2 hours saved
http://cclive.thoughtworks.com/dashboard/build/detail/cclive-release-jdk1.4
Google
cruisecontrol
Dashboard
Builds
Administration
Tests
Modifications
Build Log
Artifacts
Errors and Warnings
Done

Server : ccweb01.thoughtworks.com

cclive-release-jdk1.4 Failed

Build Time: 10 Sep 2008 01:35 GMT -05:00 Duration: 19 minutes 22 seconds
Last successful build: 3 days 23 hours 7 minutes 13 seconds ago

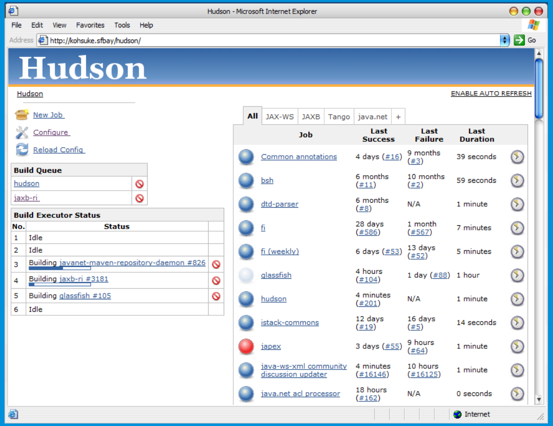
Test Suites

```
net.sourceforge.cruisecontrol.AttributeTest
Tests: 3, Failures: 0, Errors: 0, Duration: 0.376
net.sourceforge.cruisecontrol.BuildLoopInformationBuilderTest
Tests: 7, Failures: 0, Errors: 0, Duration: 0.348
net.sourceforge.cruisecontrol.BuildOutputBufferManagerTest
Tests: 2, Failures: 0, Errors: 0, Duration: 0.019
net.sourceforge.cruisecontrol.BuildQueueTest
Tests: 2, Failures: 0, Errors: 0, Duration: 0.039
net.sourceforge.cruisecontrol.BuilderTest
Tests: 3, Failures: 0, Errors: 0, Duration: 0.024
net.sourceforge.cruisecontrol.CruiseControlConfigIncludeTest
Tests: 8, Failures: 0, Errors: 0, Duration: 0.229
net.sourceforge.cruisecontrol.CruiseControlConfigPreConfTest
```

Latest Builds

- 10 Sep 2008 01:35 GMT -05:00
- 9 Sep 2008 22:21 GMT -05:00
- 9 Sep 2008 21:51 GMT -05:00
- 6 Sep 2008 02:27 GMT -05:00 build.6
- 6 Sep 2008 01:53 GMT -05:00 build.5
- 31 Aug 2008 14:10 GMT -05:00 build.4
- 24 Jul 2008 01:31 GMT -05:00
- 23 Jul 2008 20:18 GMT -05:00
- 23 Jul 2008 19:08 GMT -05:00
- 23 Jul 2008 17:52 GMT -05:00
- 23 Jul 2008 16:42 GMT -05:00

2001


Hudson - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://kohsuke.sfbay/hudson/
Go
Hudson
Hudson
ENABLE AUTO REFRESH
New Job
Configure
Reload Config
All
JAX-WS
JAXB
Tango
java.net
+
Internet

Build Queue

hudson
jaxb-ri

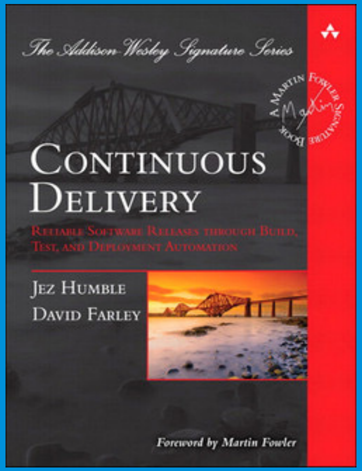
Build Executor Status

No.	Status
1	Idle
2	Idle
3	Building javanet-maven-repository-daemon #826
4	Building jaxb-ri #3181
5	Building glassfish #105
6	Idle

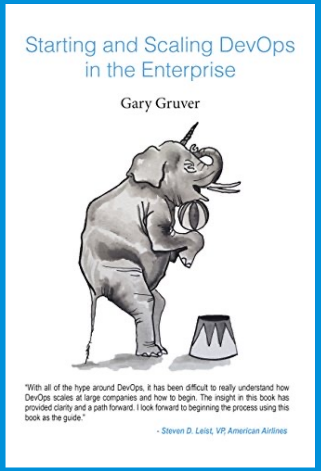
Job	Last Success	Last Failure	Last Duration
Common annotations	4 days (#16)	9 months (#3)	39 seconds
bsh	6 months (#11)	10 months (#2)	59 seconds
dtd-parser	6 months (#8)	N/A	1 minute
fi	28 days (#586)	1 month (#567)	7 minutes
fi (weekly)	6 days (#53)	13 days (#52)	5 minutes
glassfish	4 hours (#104)	1 day (#88)	1 hour
hudson	4 minutes (#201)	N/A	1 minute
istack-commons	12 days (#19)	16 days (#5)	14 seconds
japex	3 days (#55)	9 hours (#64)	1 minute
java-ws-xml community discussion updater	4 minutes (#16146)	10 hours (#16125)	1 minute
java.net acl processor	18 hours (#162)	N/A	0 seconds

2005

2011


The Addison Wesley Signature Series
A Martin Fowler Signature Book
Continuous Delivery
Reliable Software Releases through Build, Test, and Deployment Automation
Jez Humble
David Farley
Foreword by Martin Fowler

2010

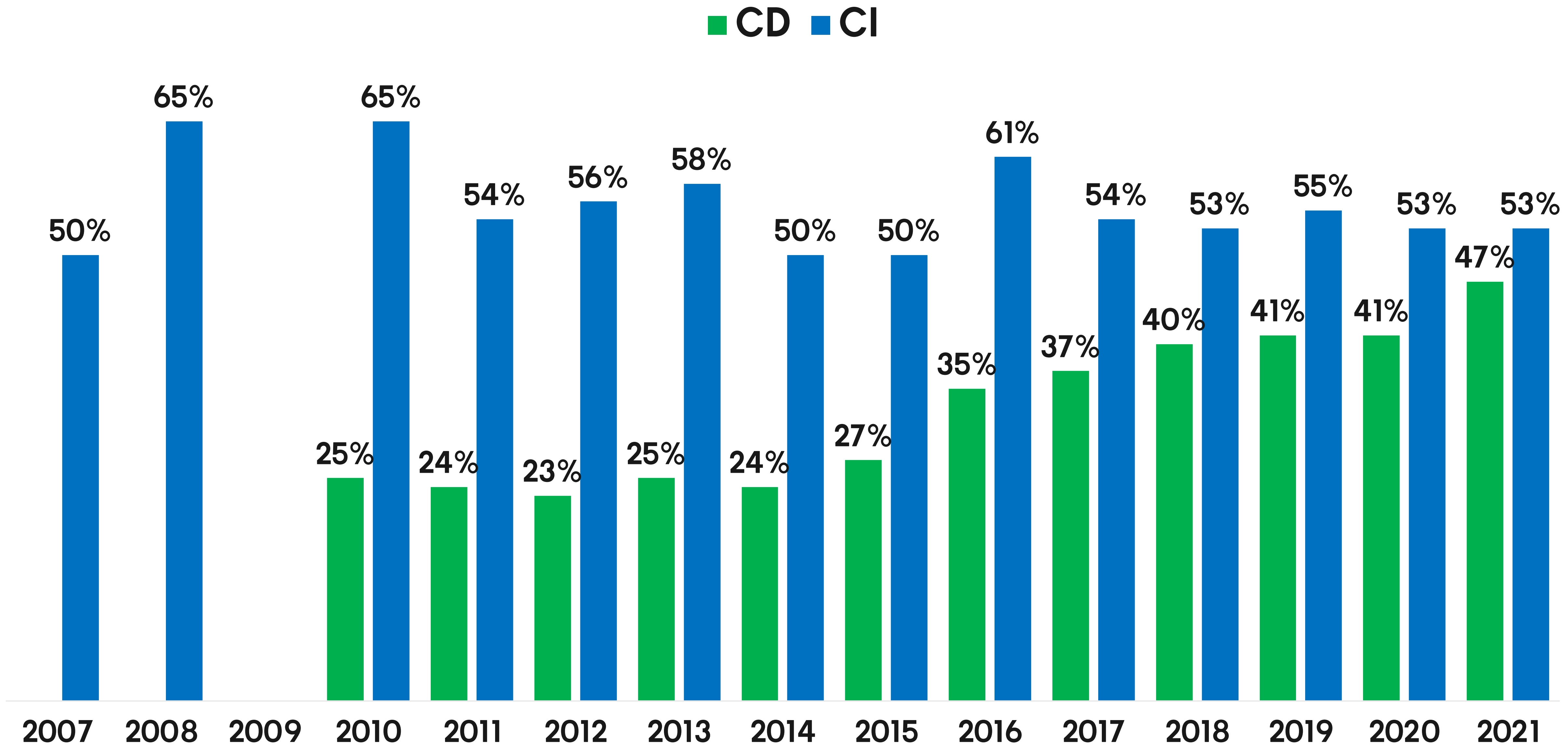

Starting and Scaling DevOps in the Enterprise
Gary Gruver
"With all of the hype around DevOps, it has been difficult to really understand how DevOps scales at large companies and how to begin. The insight in this book has provided clarity and a path forward. I look forward to beginning the process using this book as the guide."
- Steven D. Leist, VP, American Airlines

2016

"Printer firmware? Hold my beer."

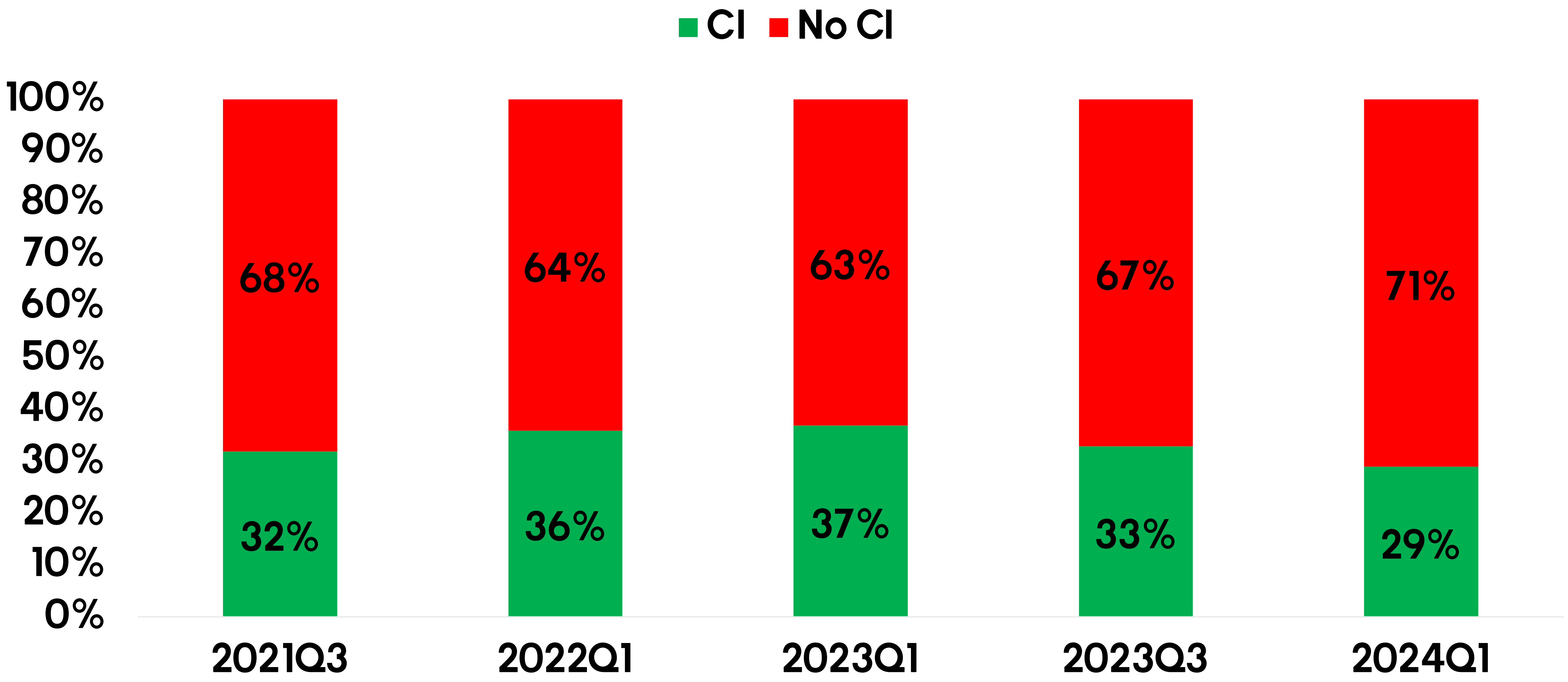
Note: originally attributed to Grady Booch in 1991, also XP principal in 1998. Sources: book listings, Sourceforge(!), Wikipedia on Sep 1st, 2023.

CI and CD usage, 2007 to 2021



Sources: State of Agile Surveys, VersionOne/CollabNet/digital.ai.

CI Usage, 2021 to 2024



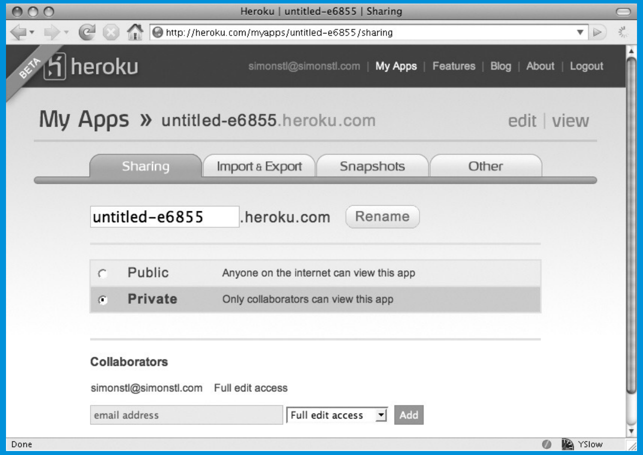
Question: Which of the following technologies have you used as part of your development activities in the last 12 months? Source: CD Foundation Surveys (Slashdata).

Source: "State of CI/CD Report 2024: The Evolution of Software Delivery Performance," CD Foundation and SlashData, April, 2024, and previous studies.

The Eternal Recurrence of (Platforms, PaaS, DevOps, Cloud Native, Golden Paths, Platform Engineering, …)

(╯°□°) ╯□ ┻━━┻)

(╯°□°) ╯□ ┻━━┻)



Heroku | untitled-e6855 | Sharing

http://heroku.com/myapps/untitled-e6855/sharing

heroku — simonstl@simonstl.com | My Apps | Features | Blog | About | Logout

My Apps » untitled-e6855.heroku.com — edit | view

Sharing | Import & Export | Snapshots | Other

untitled-e6855 .heroku.com Rename

- Public — Anyone on the internet can view this app
- [✓ Private] — Only collaborators can view this app

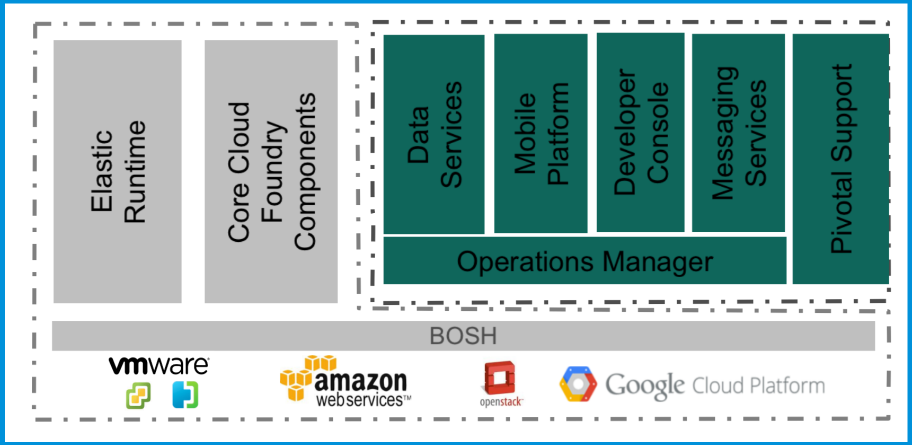
Collaborators

simonstl@simonstl.com Full edit access

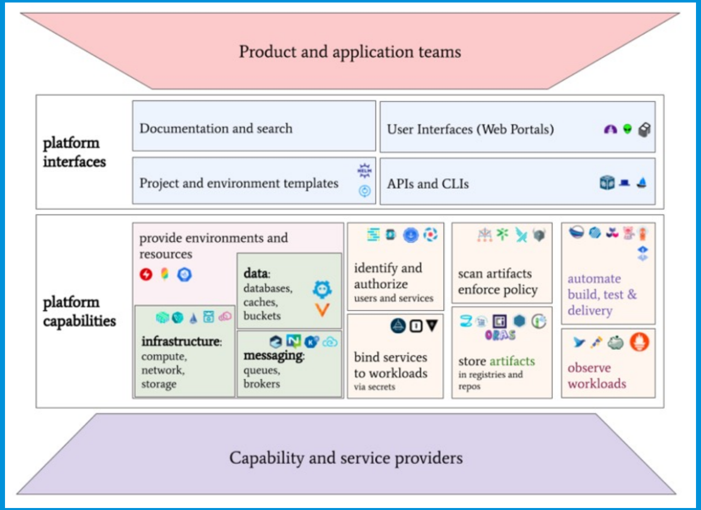
email address — Full edit access — Add

Done

2007

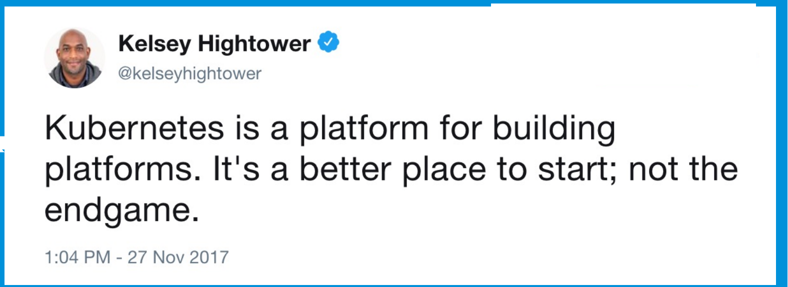


2011 to 2015



Product and application teams
platform interfaces
Documentation and search
User Interfaces (Web Portals)
Project and environment templates
APIs and CLIs
platform capabilities
provide environments and resources
data: databases, caches, buckets
infrastructure: compute, network, storage
messaging: queues, brokers
identify and authorize users and services
bind services to workloads via secrets
scan artifacts enforce policy
store artifacts in registries and repos
automate build, test & delivery
observe workloads
Capability and service providers

2023 & Beyond



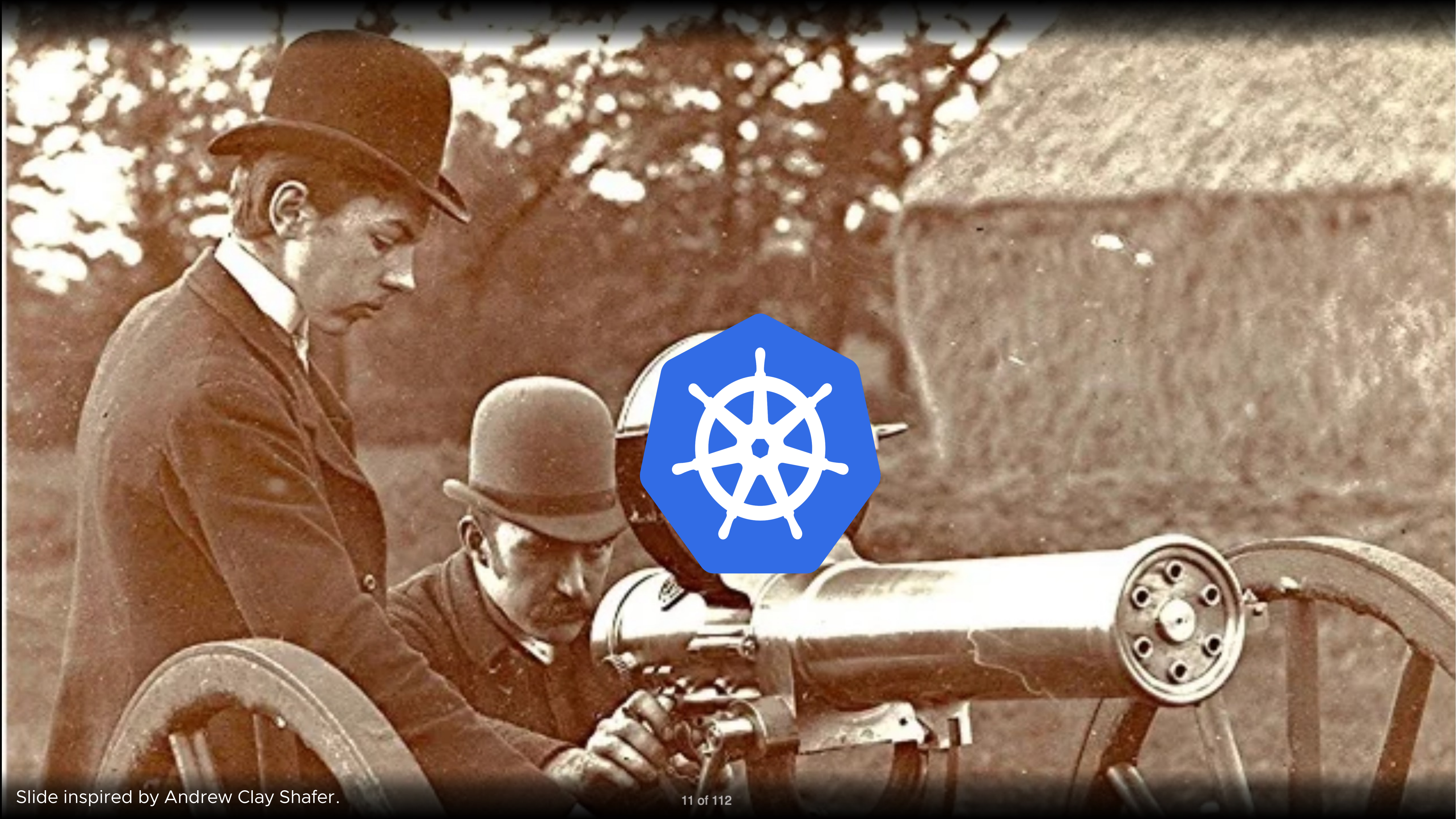
Kelsey Hightower @kelseyhightower

Kubernetes is a platform for building platforms. It's a better place to start; not the endgame.

1:04 PM - 27 Nov 2017

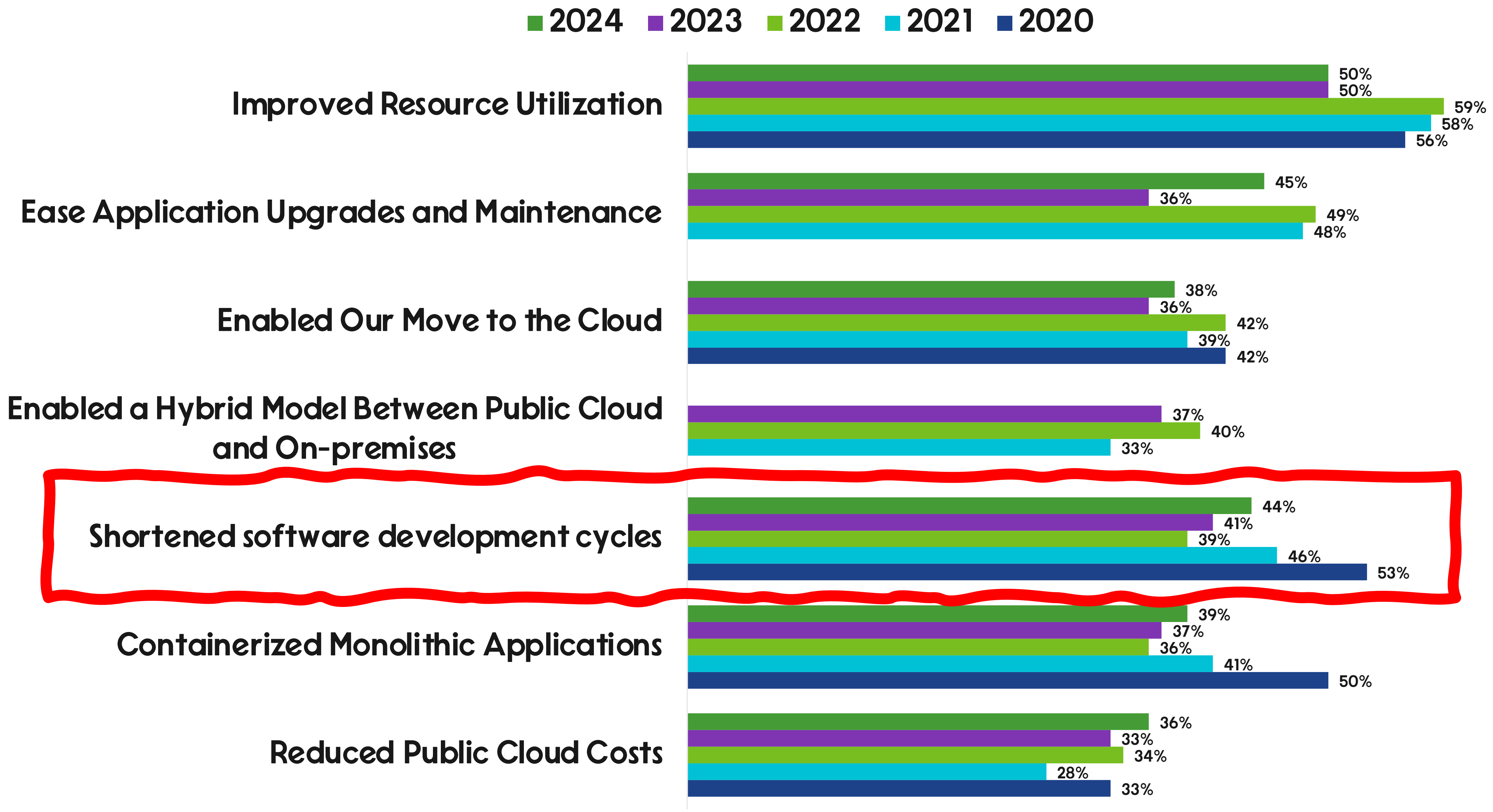
Not pictured:

OO, Small Talk, RUP, CORBA, J2EE/.Net, SOA & WS-*, RAD, Low Code, Public Clouds

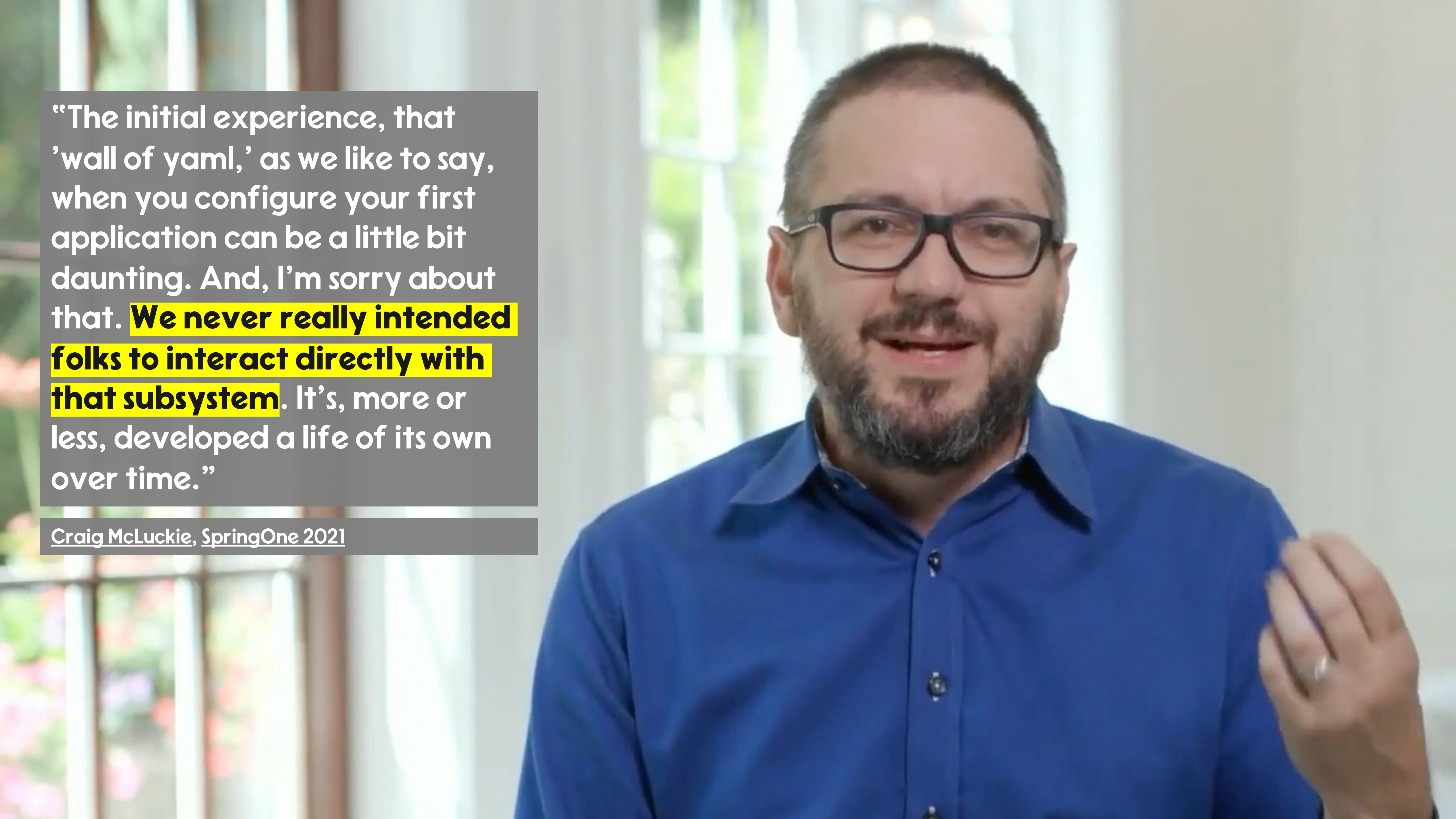


Slide inspired by Andrew Clay Shafer.

What benefits has your organization realized from operating Kubernetes?



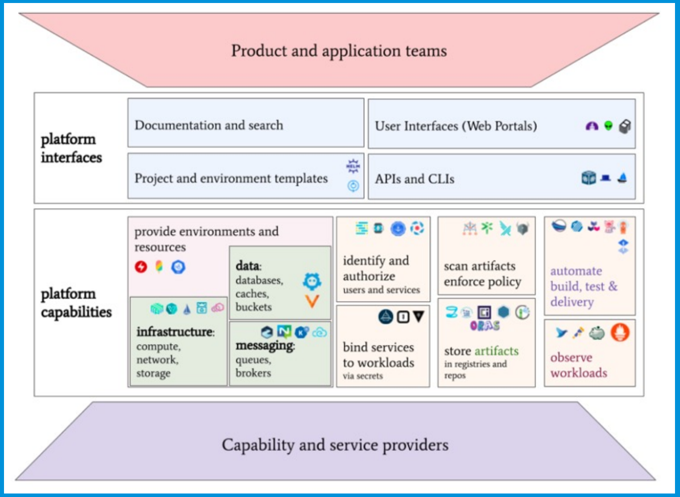
Source: State of Cloud Native Platform 2024, various State of Kubernetes. More: "Exploring the State of Cloud Native App Platforms and VMware Tanzu," July, 2024.



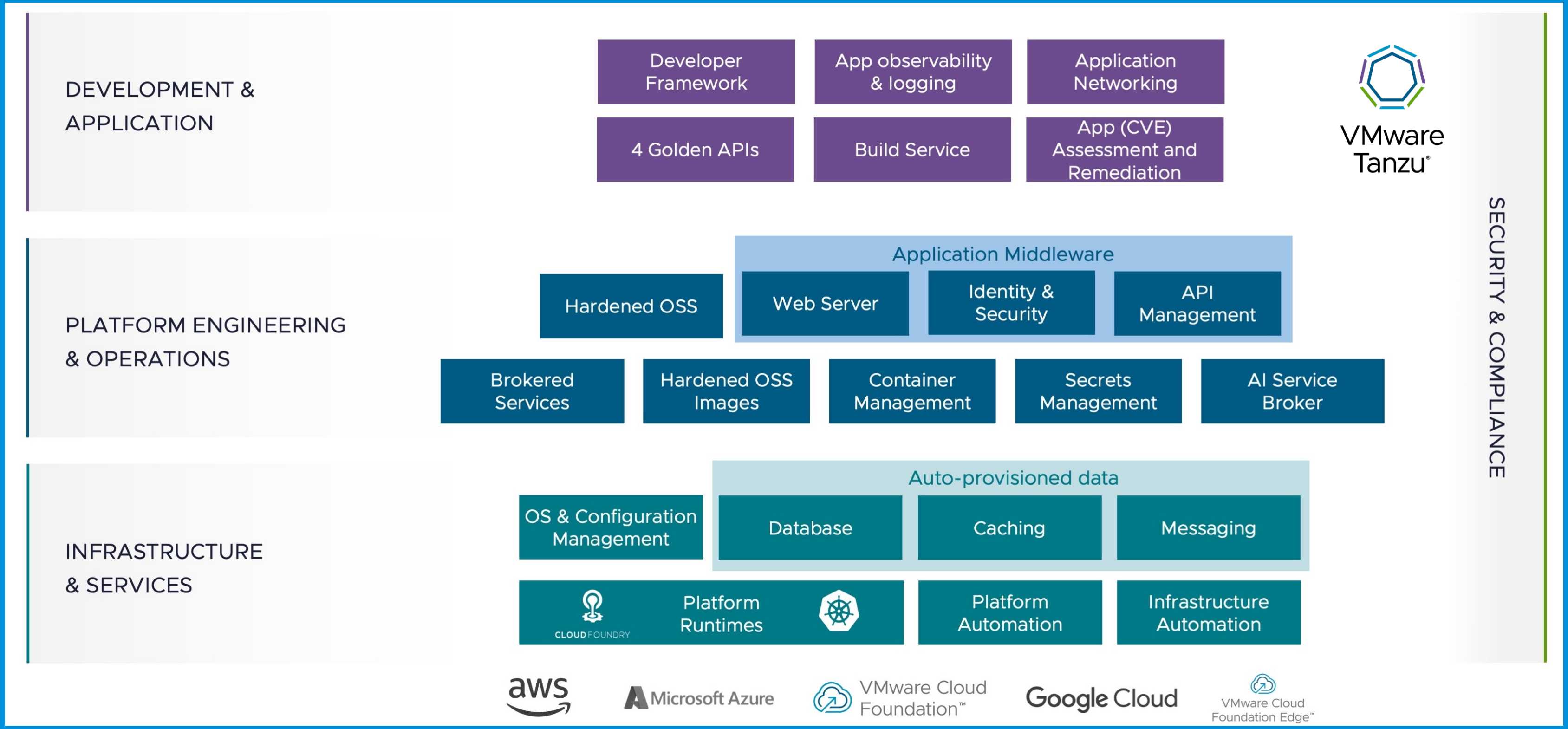
"The initial experience, that 'wall of yaml,' as we like to say, when you configure your first application can be a little bit daunting. And, I'm sorry about that. **We never really intended folks to interact directly with that subsystem**. It's, more or less, developed a life of its own over time."

Craig McLuckie, SpringOne 2021

Don’t build platforms, build apps



Product and application teams
platform interfaces
Documentation and search
User Interfaces (Web Portals)
Project and environment templates
APIs and CLIs
platform capabilities
provide environments and resources
data: databases, caches, buckets
infrastructure: compute, network, storage
messaging: queues, brokers
identify and authorize users and services
bind services to workloads via secrets
scan artifacts enforce policy
store artifacts in registries and repos
ORAS
automate build, test & delivery
observe workloads
Capability and service providers



Sources: “CNCF Platforms White Paper,” March 2023; VMware Tanzu.

Don’t build platforms.

Build apps.

Thanks!

Slides & stuff

📥 https://newsletter.cote.io/

📚 https://cote.io/fearchange/

🏢 cote@broadcom.com