# Kubernetes is not for developers…?

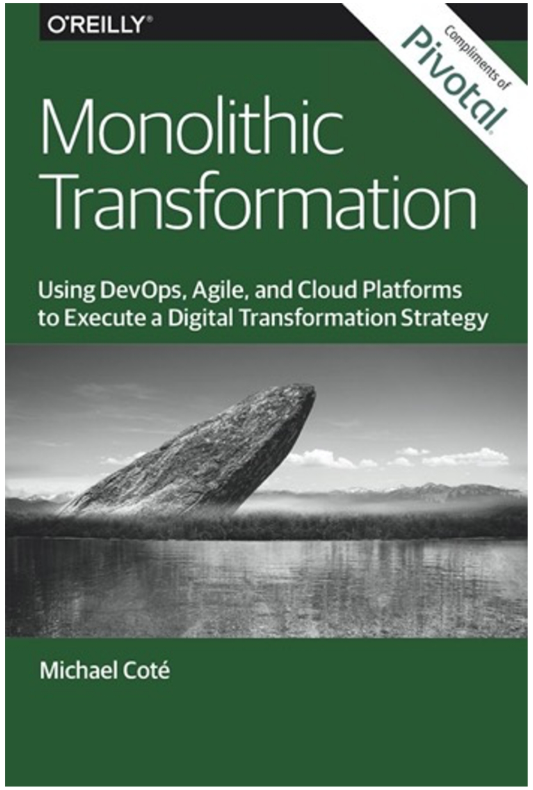
@cote – 11th November 2021

# “It depends” 🙄

is the right answer,
but it’s not the *fun* answer.

Sources: “Kubernetes: Whose Idea Was This?” panel, DevOps Loop, Oct, 2021.

# I AM COTÉ - ENGAGE WITH MY BRAND!



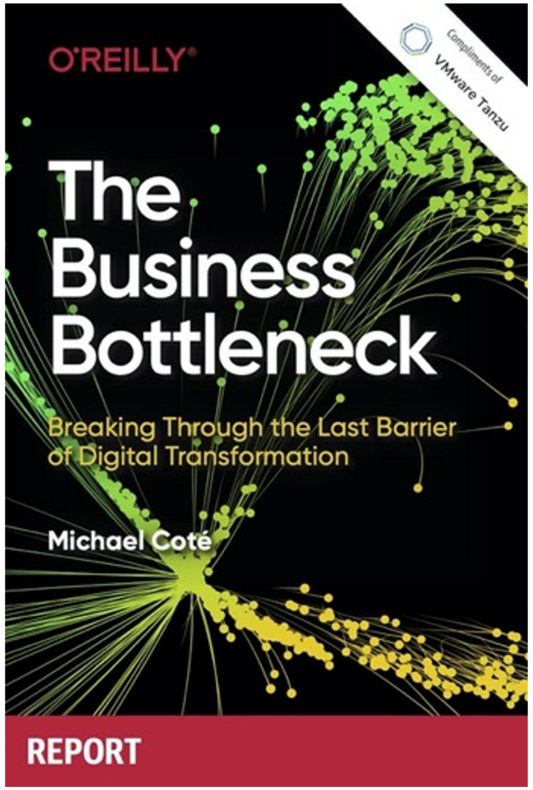
O'REILLY®

Compliments of Pivotal

Monolithic Transformation

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

Michael Coté



O'REILLY®

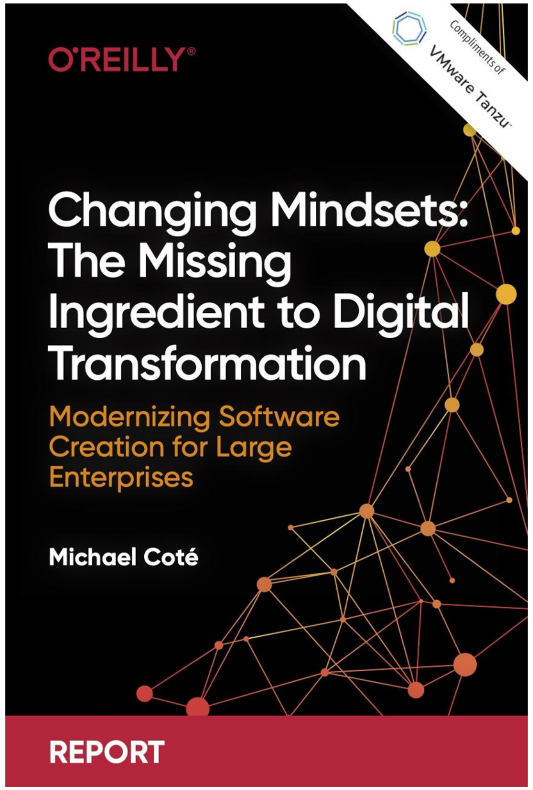
Compliments of VMware Tanzu

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

@cote – cotem@vmware.com – http://cote.io

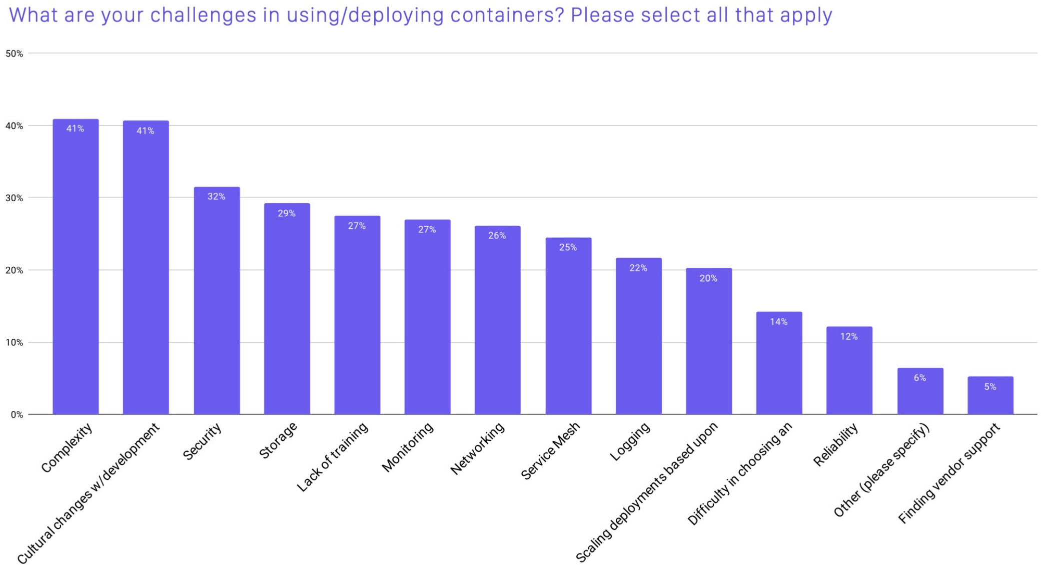
# Conclusions (to be pondered)

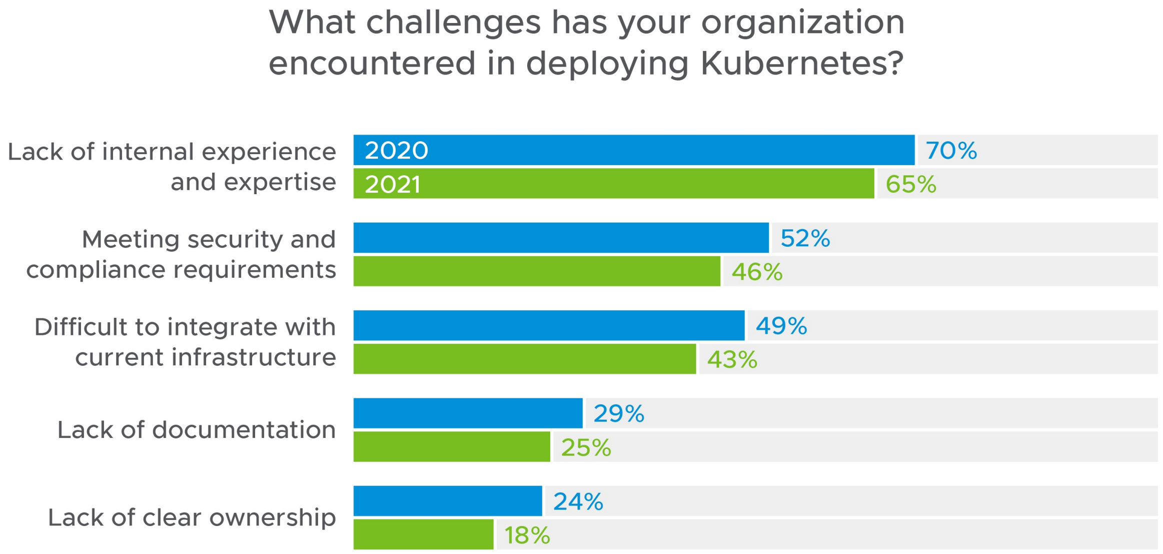
- The larger your organization, the further developers should be from kubernetes to maximize app value, developer productivity, and stability.

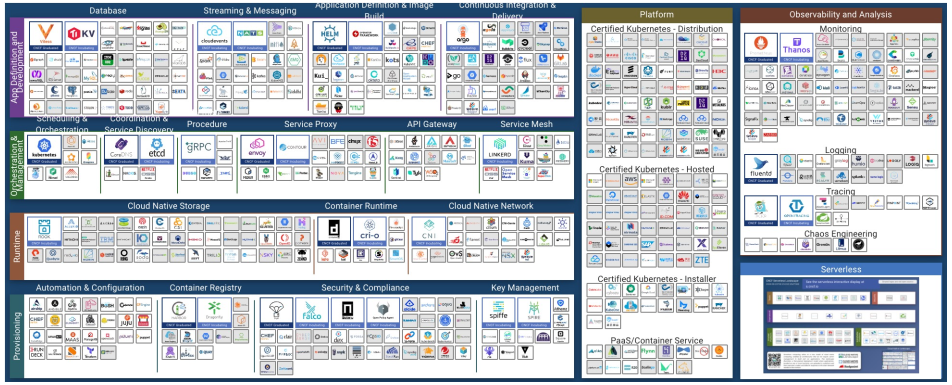
- Variation at the infrastructure and tools layer is often “waste” at app-centric organizations.

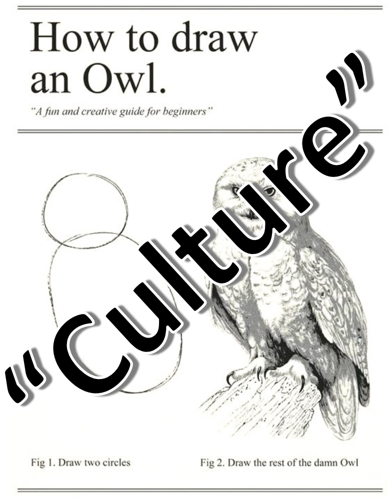
- Developers enjoy DIY too much. Instead, rely on a platform as a product team to spend time wisely and be strategic

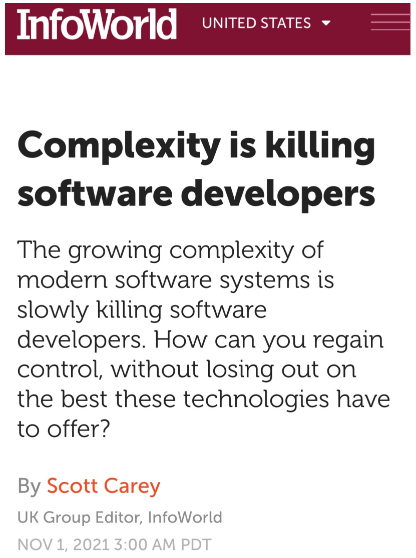
# So many complaints


What are your challenges in using/deploying containers? Please select all that apply


What challenges has your organization encountered in deploying Kubernetes?


App Definition and Development
Database
Streaming & Messaging
Application Definition & Image Build
Continuous Integration & Delivery
Platform
Certified Kubernetes - Distribution
Observability and Analysis
Monitoring
Orchestration & Management
Scheduling & Orchestration
Coordination & Service Discovery
Procedure
Service Proxy
API Gateway
Service Mesh
Certified Kubernetes - Hosted
Logging
Tracing
Chaos Engineering
Runtime
Cloud Native Storage
Container Runtime
Cloud Native Network
Certified Kubernetes - Installer
Serverless
Provisioning
Automation & Configuration
Container Registry
Security & Compliance
Key Management
PaaS/Container Service


How to draw an Owl.
"A fun and creative guide for beginners"
Fig 1. Draw two circles
Fig 2. Draw the rest of the damn Owl
„Culture"



## Complexity is killing software developers

The growing complexity of modern software systems is slowly killing software developers. How can you regain control, without losing out on the best these technologies have to offer?

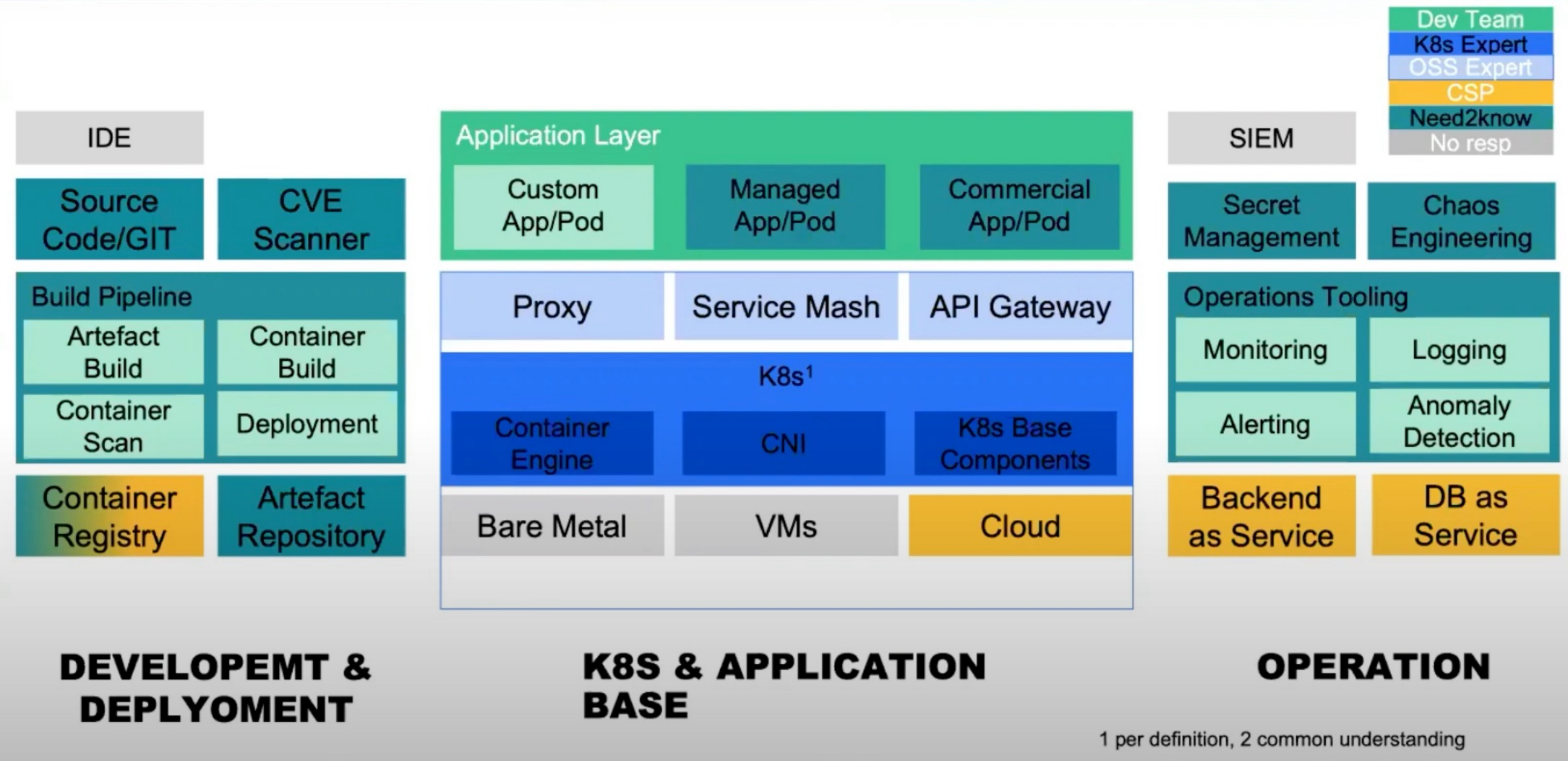
By Scott Carey

UK Group Editor, InfoWorld

NOV 1, 2021 3:00 AM PDT

Sources: CNCF Survey 2020, May and June 2020; "State of Kubernetes 2021" report, VMware; InfoWorld article.

“A platform for building platforms.”



Dev Team
K8s Expert
OSS Expert
CSP
Need2know
No resp

IDE

Source Code/GIT

CVE Scanner

Build Pipeline

Artefact Build

Container Build

Container Scan

Deployment

Container Registry

Artefact Repository

Application Layer

Custom App/Pod

Managed App/Pod

Commercial App/Pod

Proxy

Service Mash

API Gateway

K8s[1]

Container Engine

CNI

K8s Base Components

Bare Metal

VMs

Cloud

SIEM

Secret Management

Chaos Engineering

Operations Tooling

Monitoring

Logging

Alerting

Anomaly Detection

Backend as Service

DB as Service

**DEVELOPEMT & DEPLYOMENT**

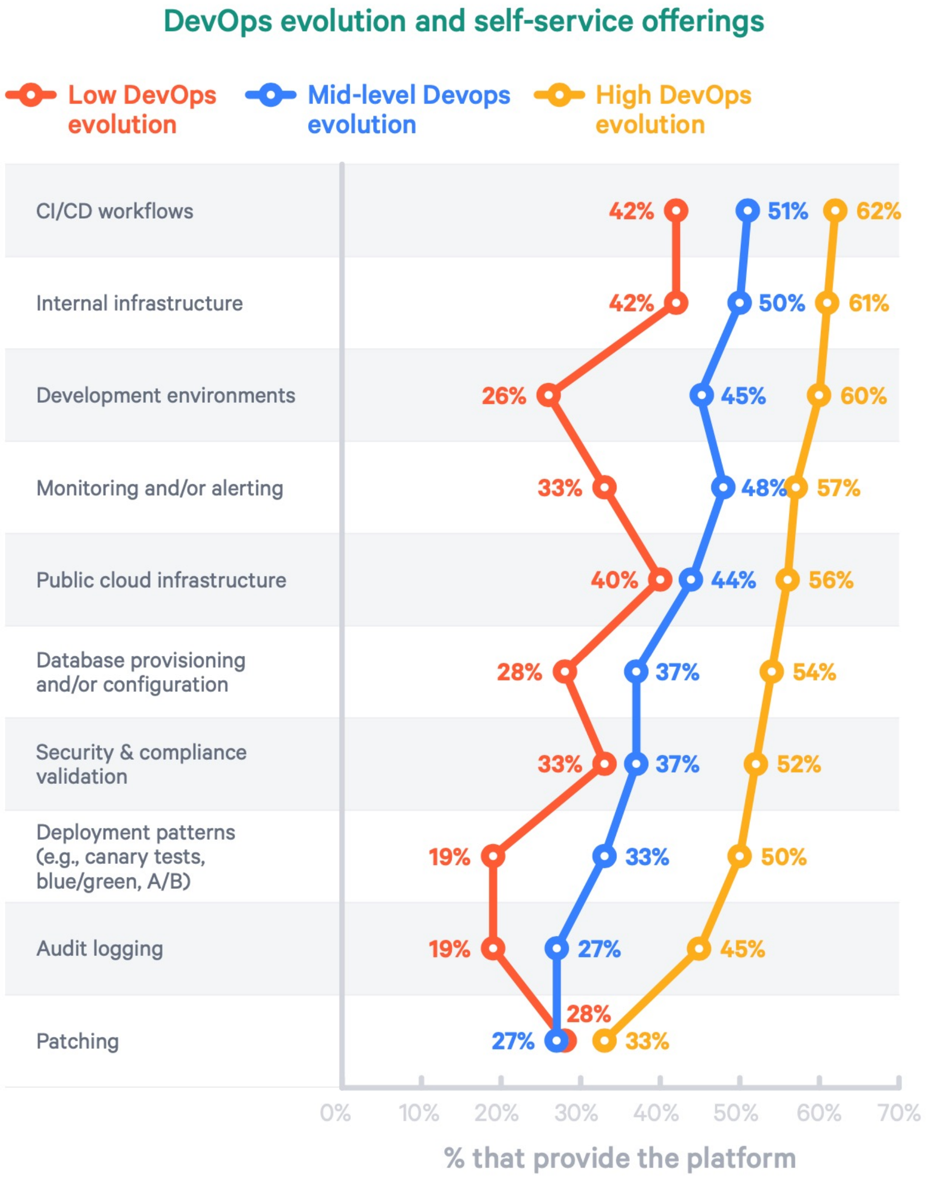
**K8S & APPLICATION BASE**

**OPERATION**

1 per definition, 2 common understanding

Source: "A Cloud Native Foundation For Developer/Application Platforms," Max Körbächer, CNCF DevX Day, Oct, 2021.

A platform is a runtime environment, with services/middleware, developer tools, and **self-service access**

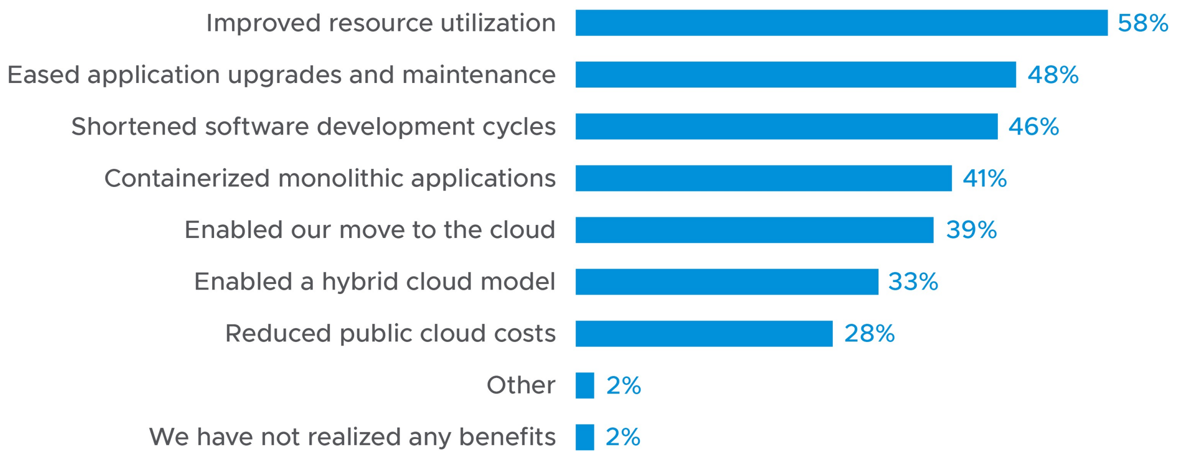


Sources: "State of DevOps Report 2020," Puppet and CircleCI edition; "What I Talk About When I Talk About Platforms," Evan Bottcher, 2018; "Mind the platform execution gap," Cristóbal García García & Chris Ford, 2021; "Why You Should Treat Platform as a Product," Zac Bergquist & Joe Fitzgerald, 2018 to 2021.

# Cloud native architectures give dev and ops more capabilities and quality

- Cloud-native capabilities – decoupling, multi-version in prod, CI/CD, etc.
- Standardize & centralize infrastructure to avoid variation waste.
- Reduce ops & dev toil
- Increase reliability, rollbacks, rolling deploys, quick patching, enforce policy.

What benefits has your organization realized from operating Kubernetes?



Sources: “State of Kubernetes 2021” report, VMware.

# Which “developer” are we talking about?

# Platform waste* to *app* devs vs. value to ops devs

- **Planning** - selecting from all the tools (e.g., CNCF landscape)
- **Integration/Creation** – making all the tools work together.
- **Product Managing** - Incremental updates, maintenance, and patching.
- **Operating** - Running the platform, diagnosing problems, security.
- **Funding** – who pays for it? Right before the check comes, developers always end up going to the toilet.

- All of these are a full time job on there own, esp. the last two.

* Built by app dev, negligence, shifting priorities, and poor decisions/skills often create a negative flywheel.

# Getting there

- Platform as a product mentality – devs as customers.
- Lead with business-case/value – productivity, security, stability. (5 S's.)
- Take advantage of vendor and community innovation and work.
- Start small, learn - don't feel rushed: "By 2026, 20% of all enterprise applications will run in containers, which is an increase from fewer than 10% in 2020."

More: "Platform as a Product," Coté. Sources: "The Innovation Leader's Guide to Navigating the Cloud-Native Container Ecosystem," Arun Chandrasekaran & Wataru Katsurashima, Gartner, Aug, 2021; "What I Talk About When I Talk About Platforms," Evan Bottcher, 2018; "Mind the platform execution gap," Cristóbal García García & Chris Ford, 2021; "Why You Should Treat Platform as a Product," Zac Bergquist & Joe Fitzgerald, 2018 to 2021.

# There are many appdev layers for Kubernetes

- Knative – simplifying event architectures, deployments w/CNB, simplified configuration

- Tanzu Application Platform

- Cloud Foundry

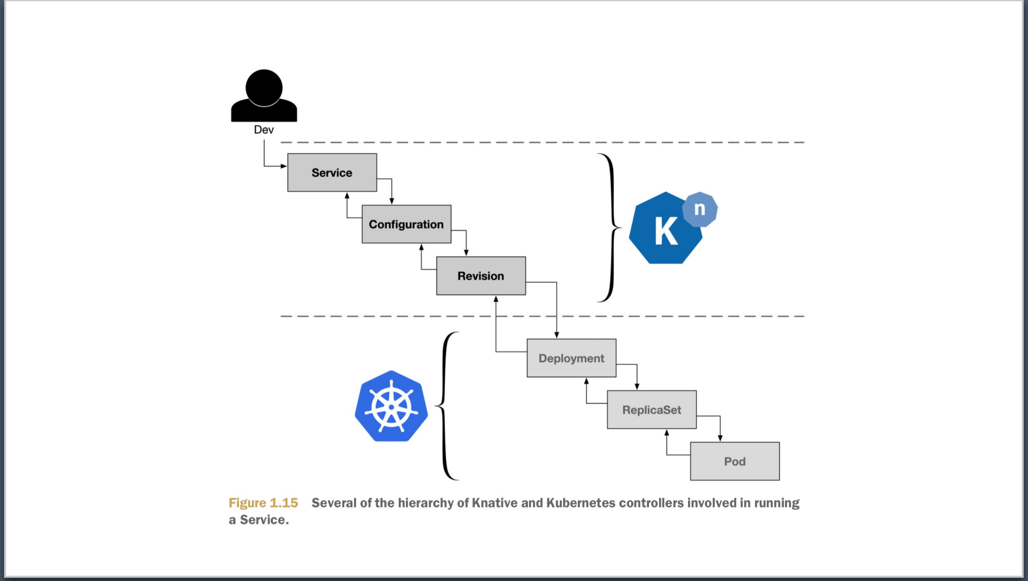
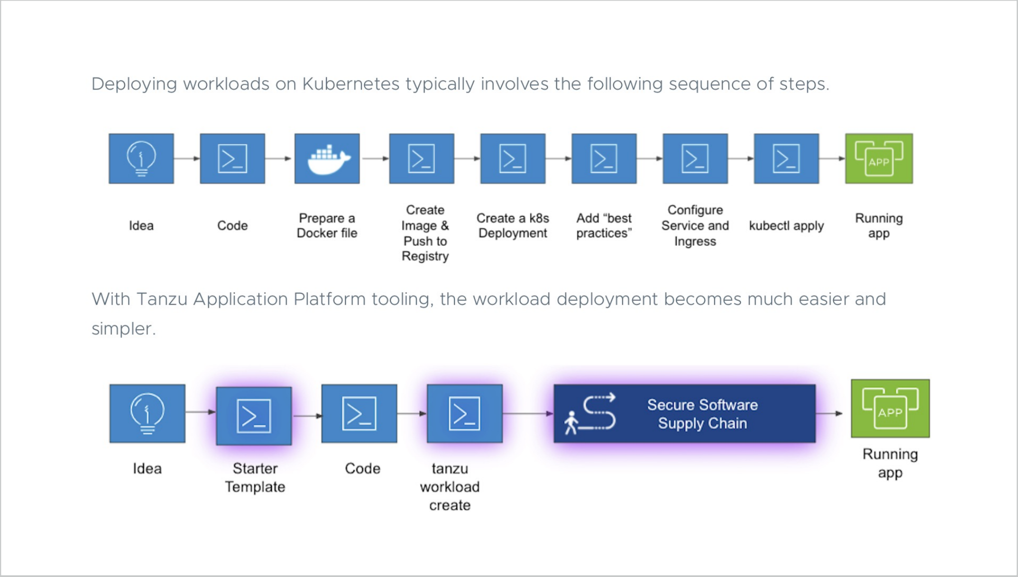


Figure 1.15 Several of the hierarchy of Knative and Kubernetes controllers involved in running a Service.


Deploying workloads on Kubernetes typically involves the following sequence of steps.

With Tanzu Application Platform tooling, the workload deployment becomes much easier and simpler.

Sources: *Knative Fundamentals*, Jacques Chester, 2020; "Recognizing and Removing Friction Points in the Developer Experience on Kubernetes," Nivedita Ghosh, Nov, 2021.

So…?

“It depends”… 🤔

# Thanks!

@cote – cotem@vmware.com – http://cote.io