

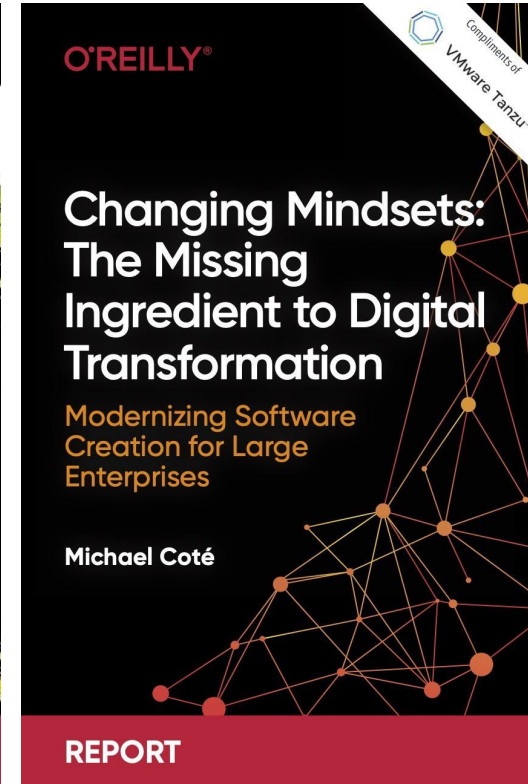
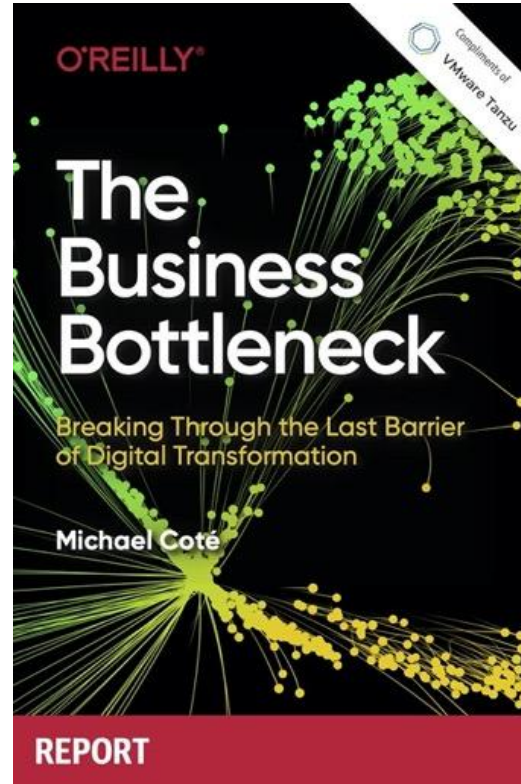
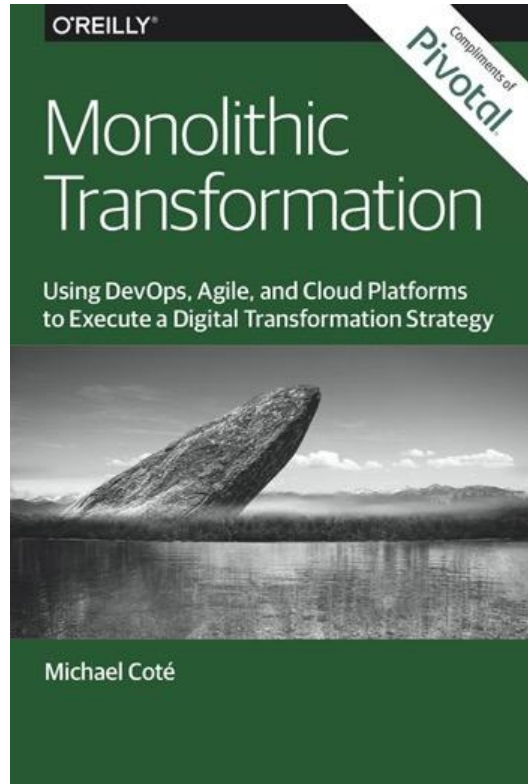
k8s4devs...?

Spoilers/conclusions

- People complain about Kubernetes, but the (potential) benefits push people to use it.
- So far, Kubernetes has low developer experience, but that is by design (mostly).
- Kubernetes is not for APPLICATION developers, but it is for platform engineers.
- Start a platform as a product group, keep your app devs dev'ing those apps.

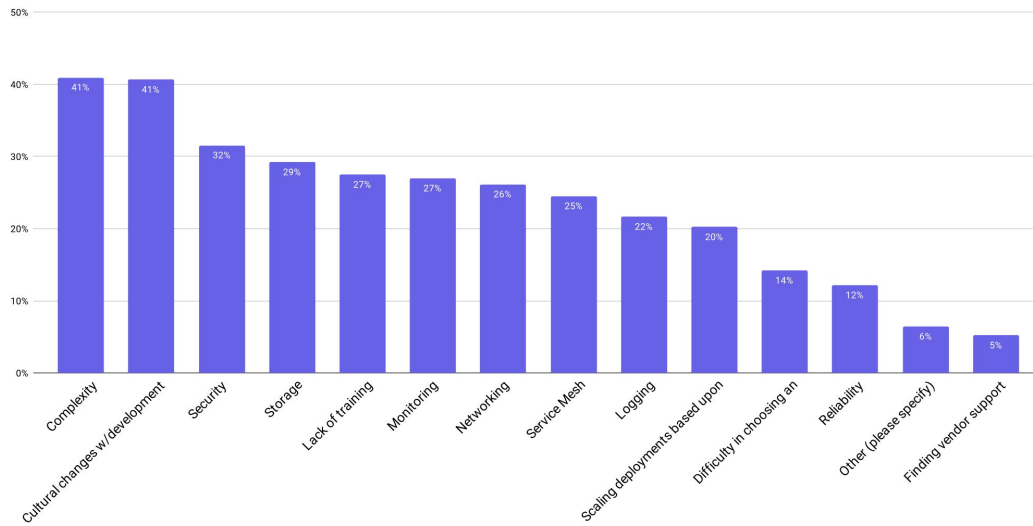
Coté

@cote | <https://cote.io> | cotem@vmware.com



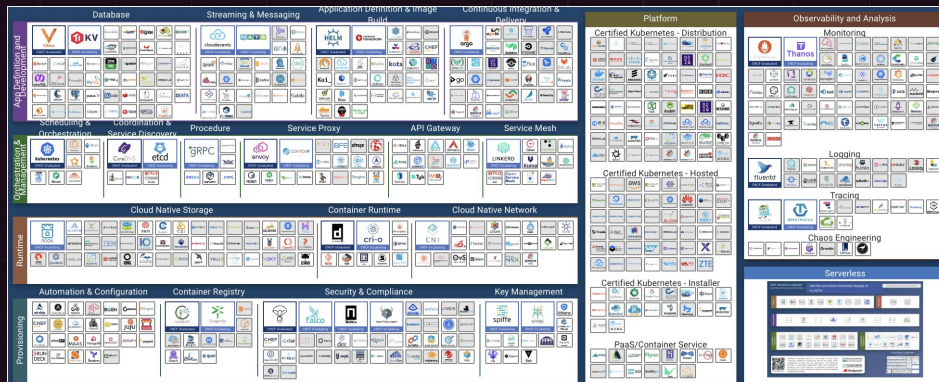
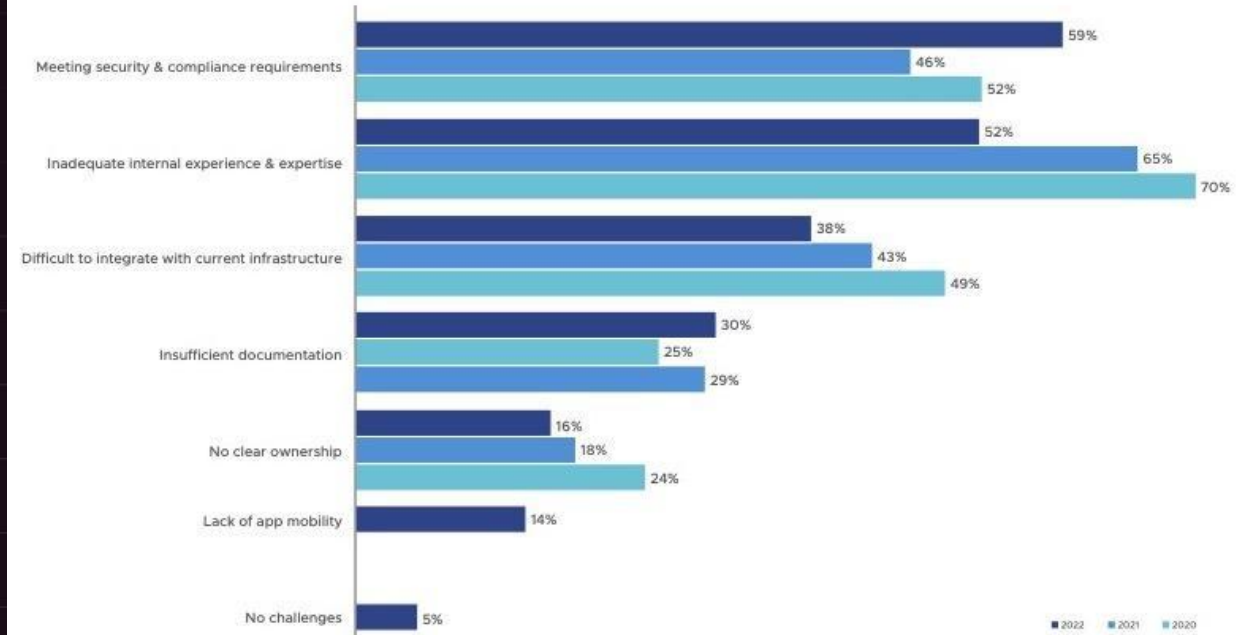
So many complaints

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



What challenges has your organization encountered in DEPLOYING Kubernetes?

(Choose all that apply)



vmware®

Sources: CNCF Survey 2020, May and June 2020; "State of Kubernetes 2022" report, VMware; InfoWorld article.

How to draw an Owl.

"A fun and creative guide for beginners"

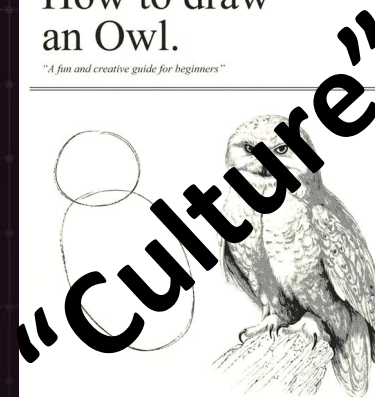


Fig 1. Draw two circles

Fig 2. Draw the rest of the damn Owl

InfoWorld UNITED STATES

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?

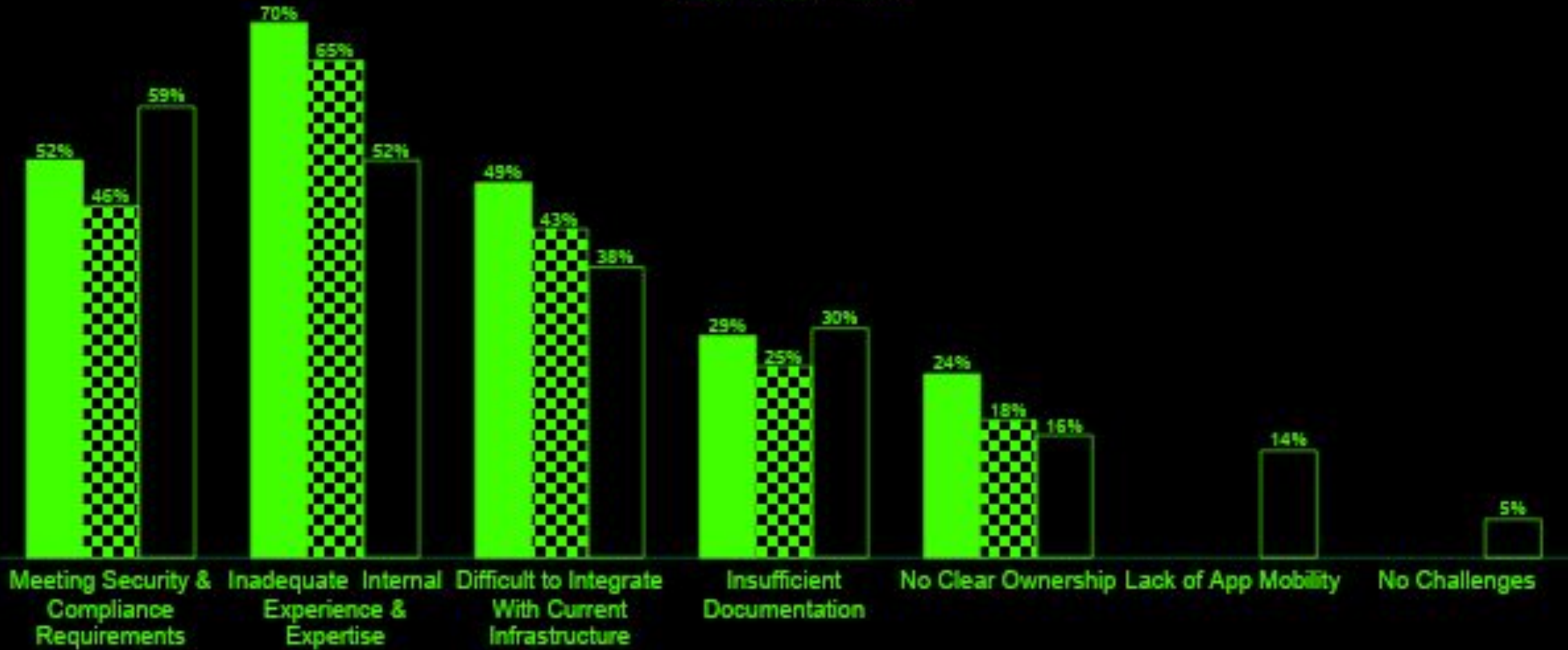
By Scott Carey

UK Group Editor, InfoWorld

NOV 1, 2021 3:00 AM PDT

What challenges has your organization encountered in DEPLOYING Kubernetes?

■ 2020 ■ 2021 □ 2022



Despite complaints, Kubernetes is attractive

- 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.

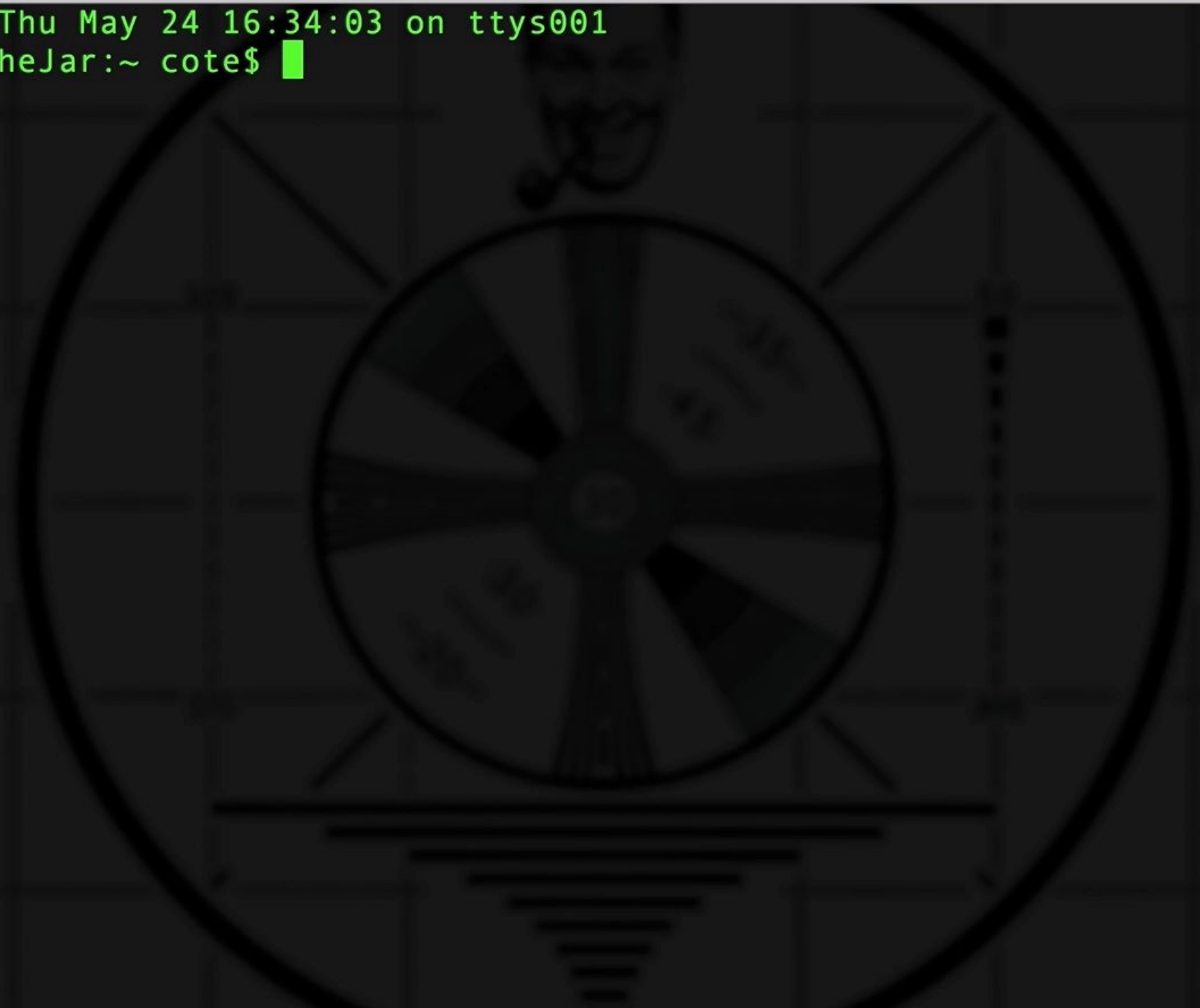
What benefits has your organization realized from operating Kubernetes? Choose all that apply.

■ 2018 ■ 2020 ■ 2021 ■ 2022

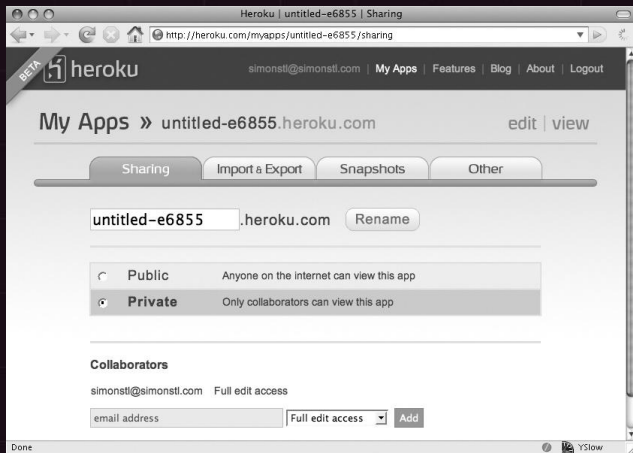


Last login: Thu May 24 16:34:03 on ttys001

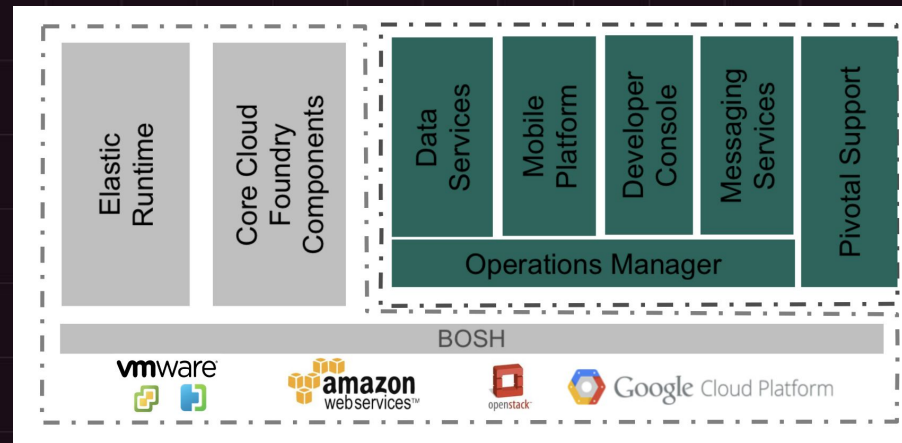
PicklesFromTheJar:~ cote\$ █



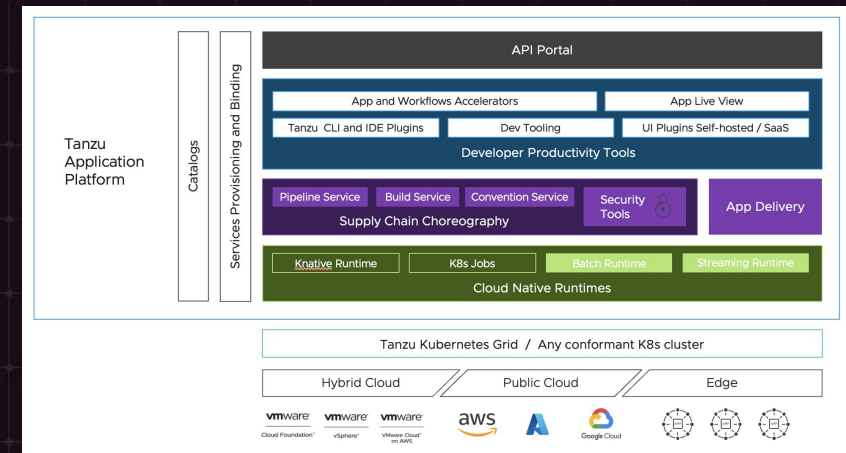
The eternal reoccurrence of platforms



2007



2015



2022



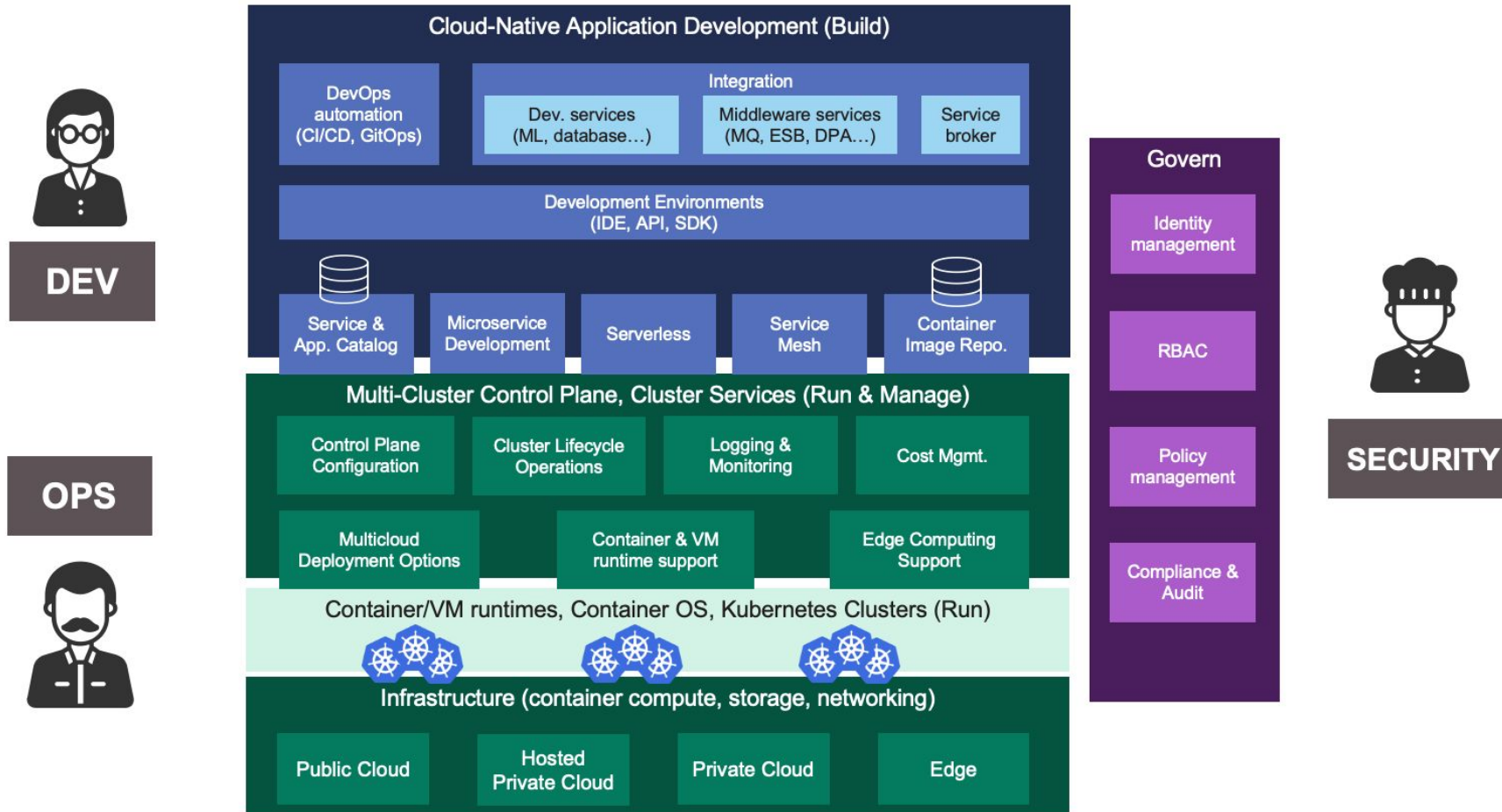
High performing organizations rely on platforms

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.

DevOps evolution and self-service offerings



Should you build it? Run it? Or use it?



Building a platform is waste for APP 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.

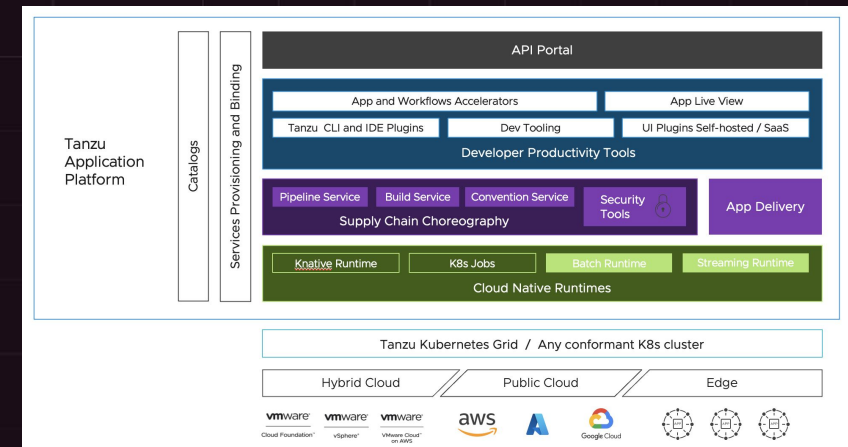
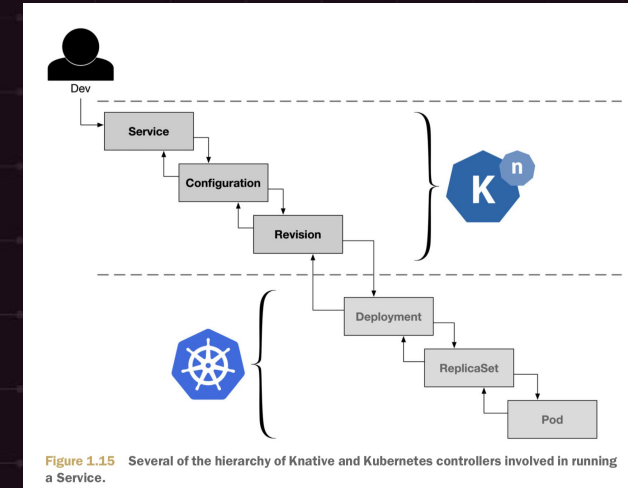
- All of these are a full time job on there own, esp. the last two.

Planning out your platform strategy

- Establish platform engineering group.
- 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."

Avoid building your own platform, start with existing platforms & components

- There are many, many options.
- Knative – simplifying event architectures, deployments w/CNB, simplified configuration
- Tanzu Application Platform - app dev platform toolkit & modules
- PaaSes like Cloud Foundry
- Commit to a public cloud platform



Thanks!

@cote - cotem@vmware.com - https://cote.io