

Being the Developer's Developer



Your Journey to Awesomeness



DevOps



Self-Service,
On-Demand,
Elastic
Infrastructure



Automation



CI & CD
Pipelines



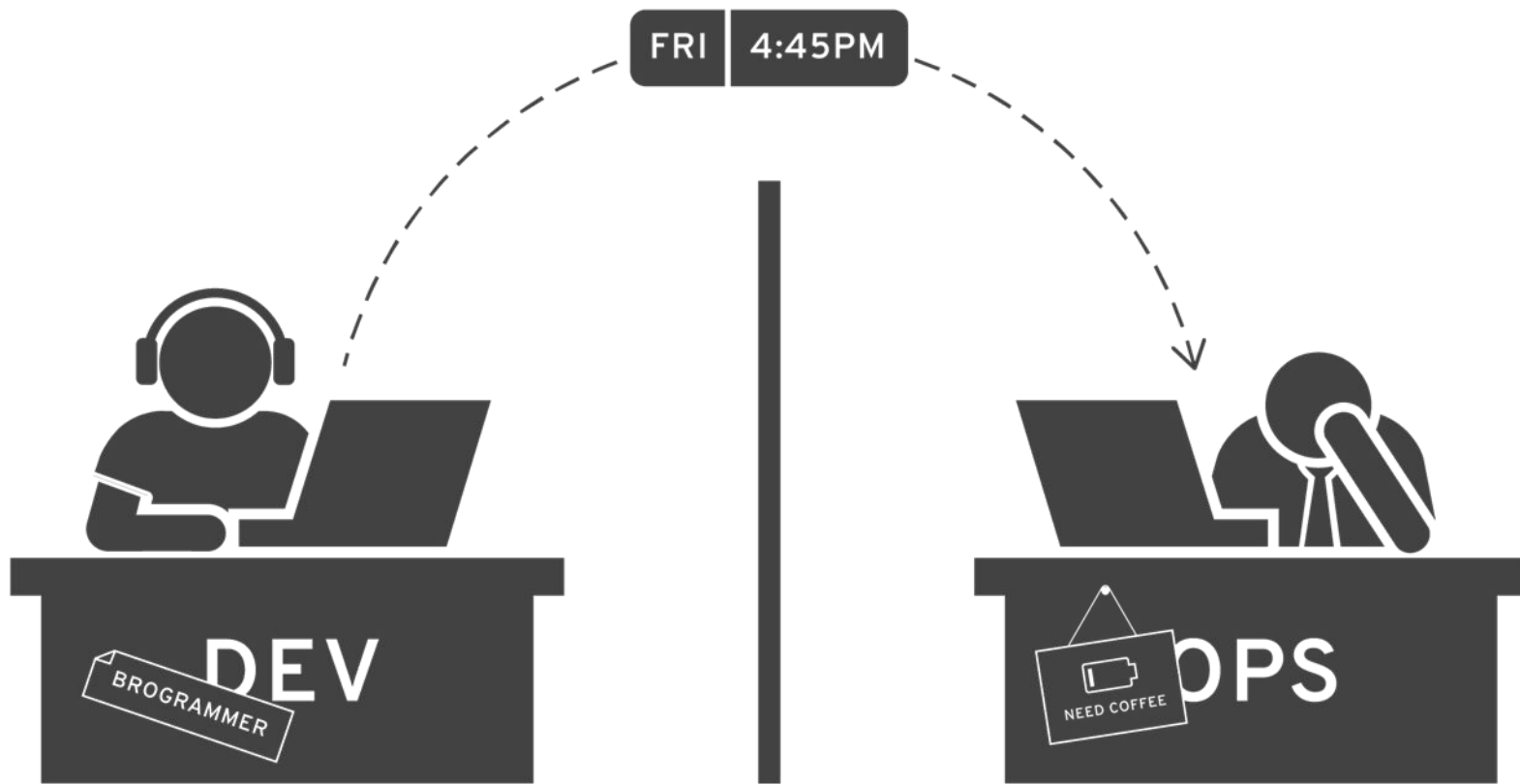
Advanced
Deployment
Techniques



**Platform
Engineer
with an
IDP**
(Backstage)

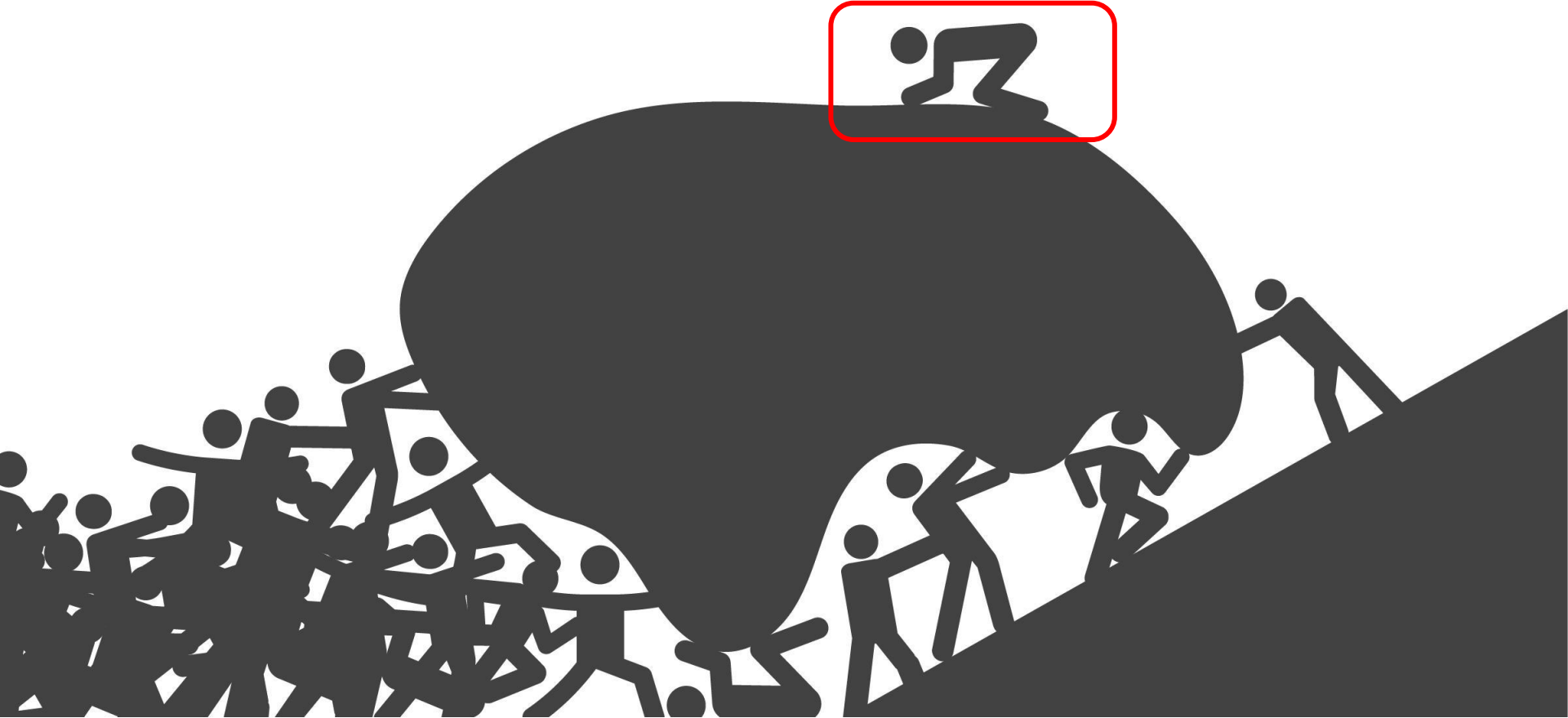


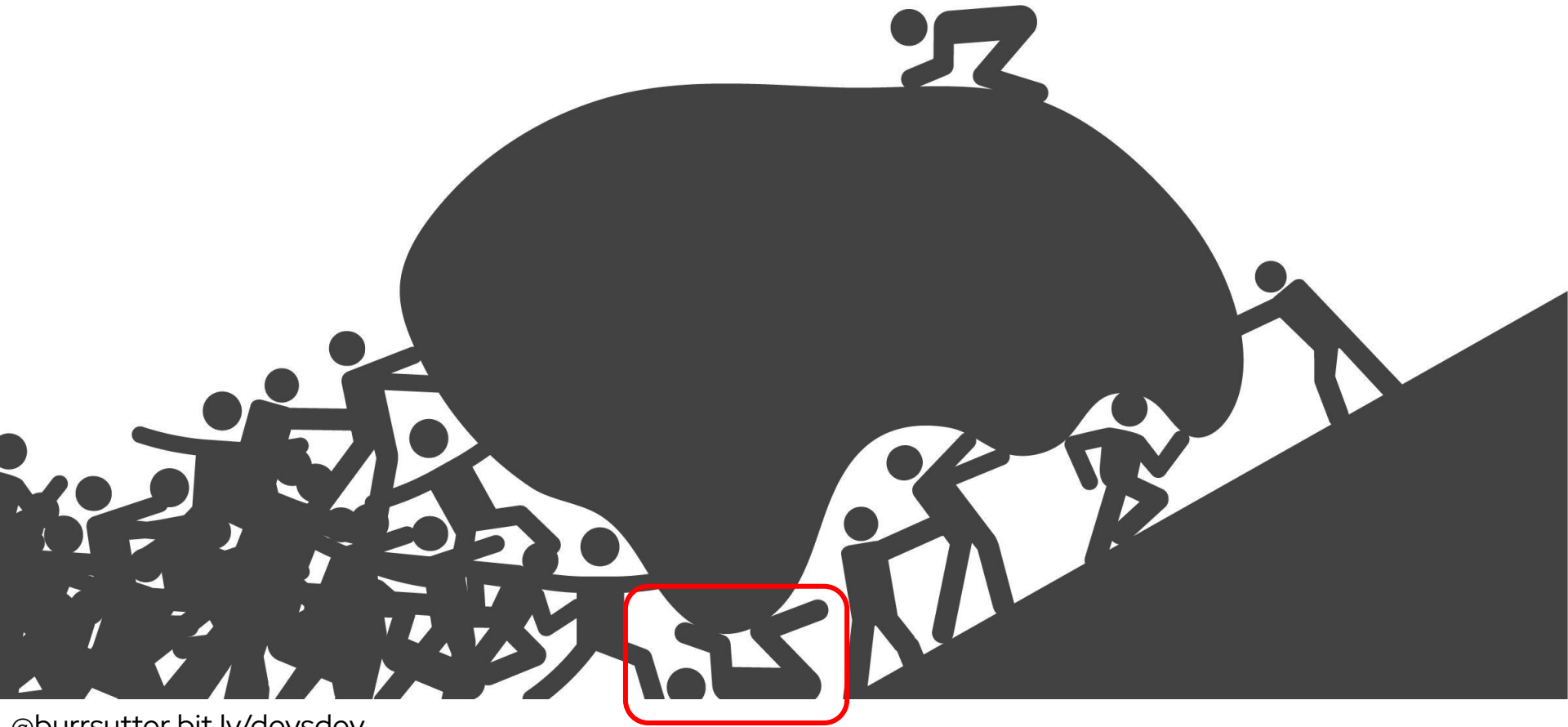
<https://www.menti.com/al8gkktjnbsv>

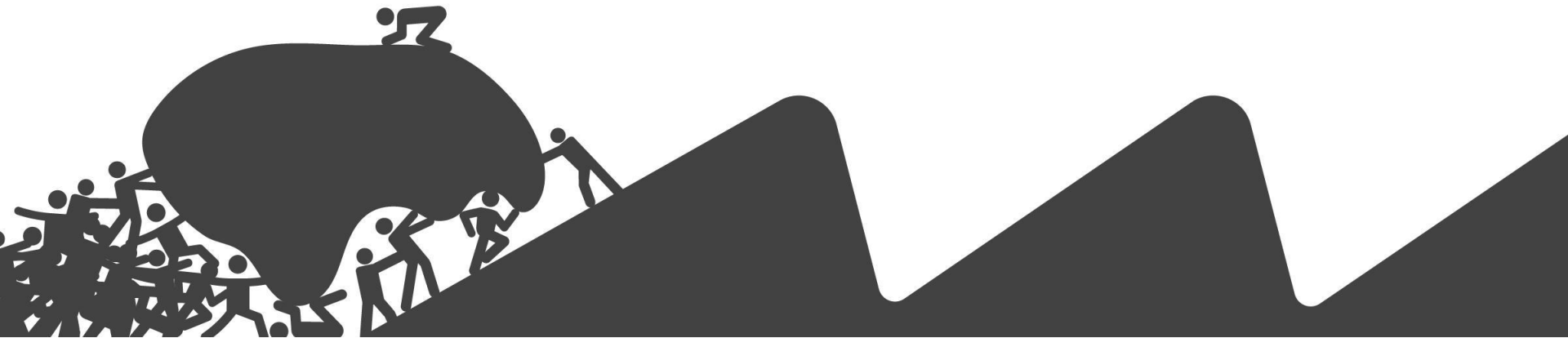






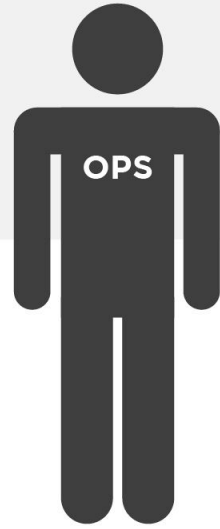


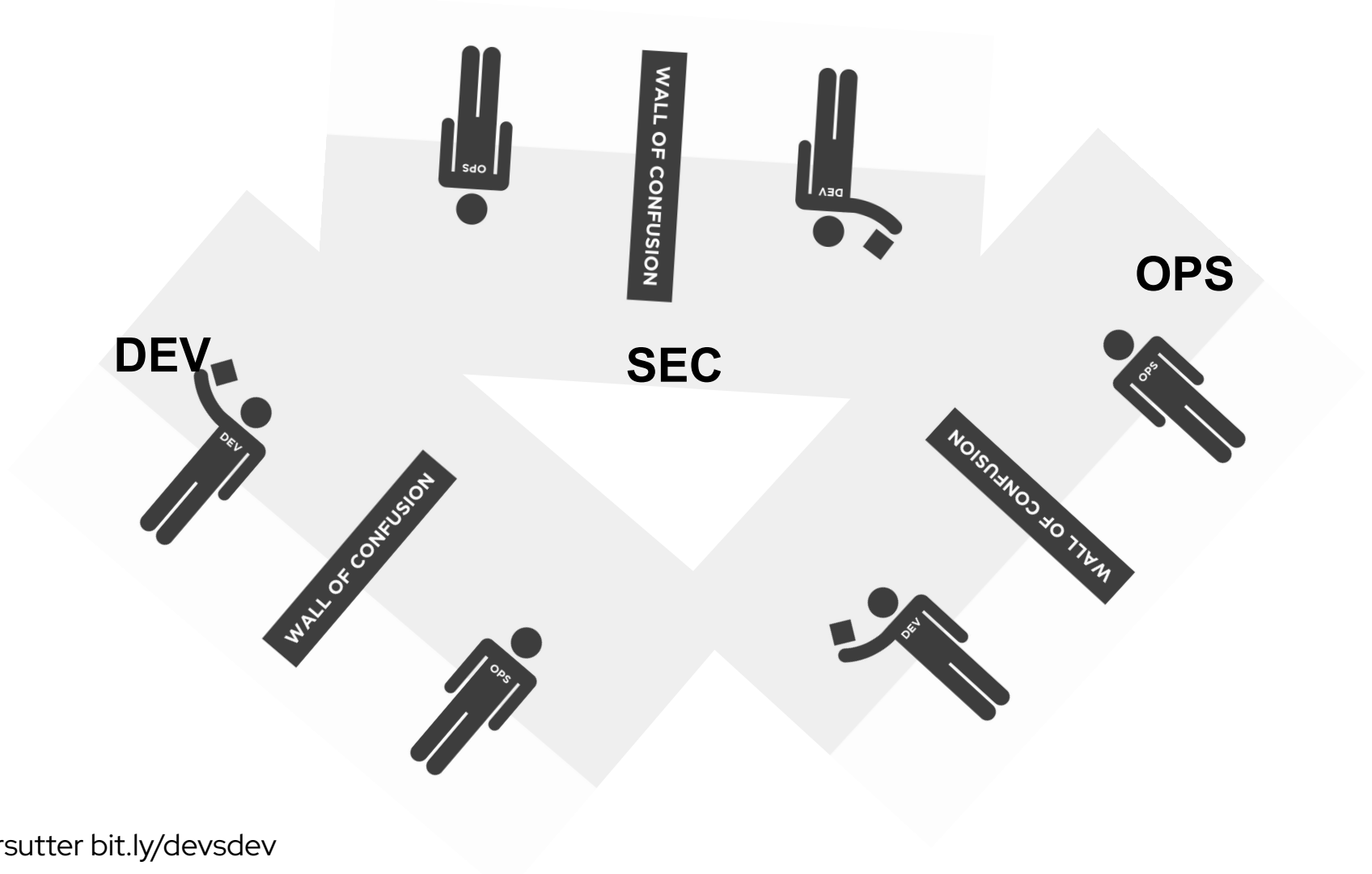






WALL OF CONFUSION







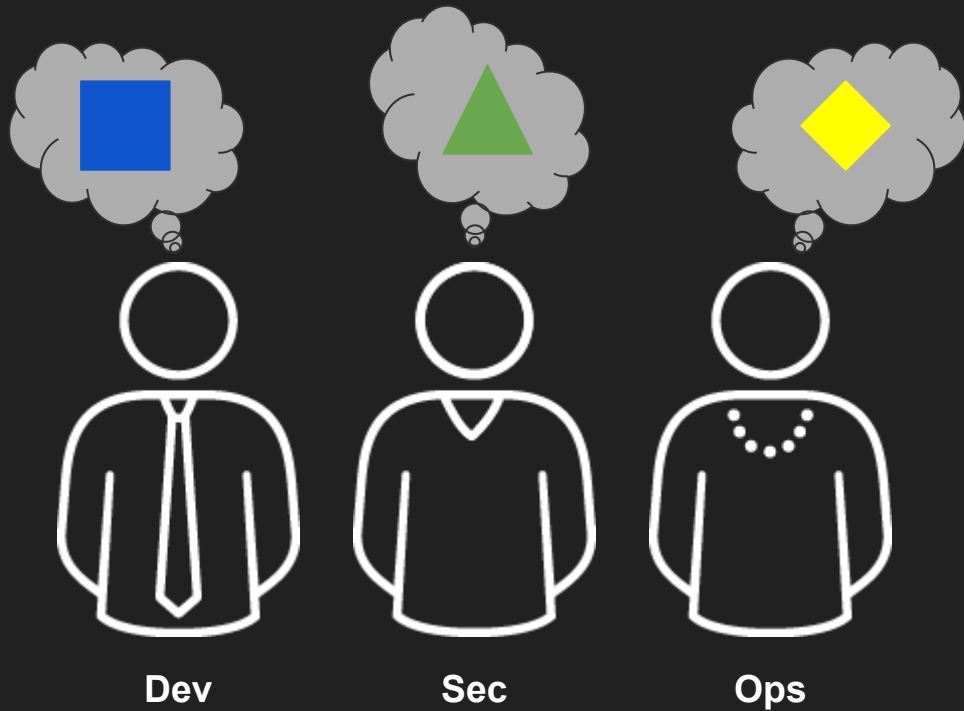
Dev



Sec



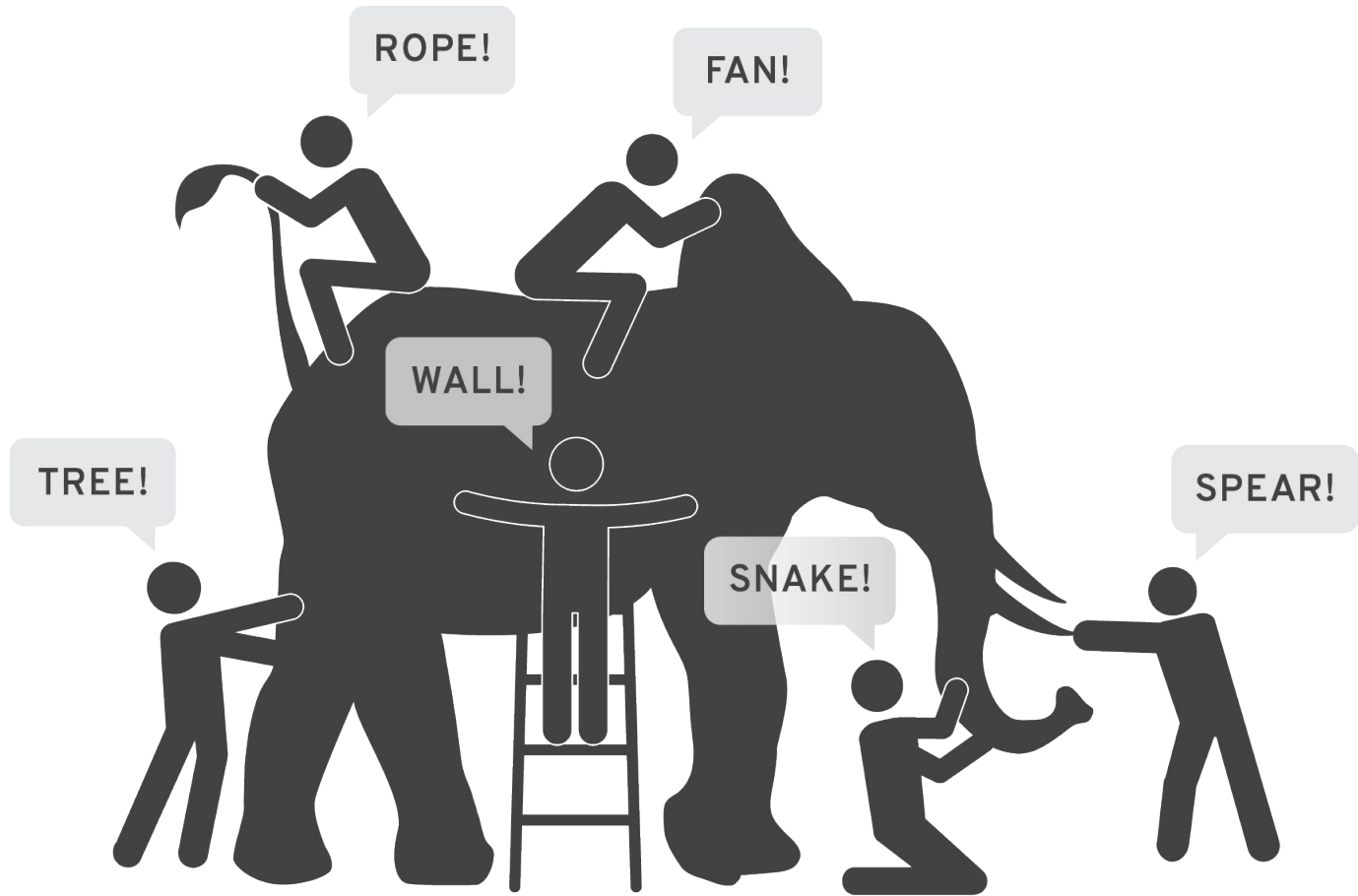
Ops

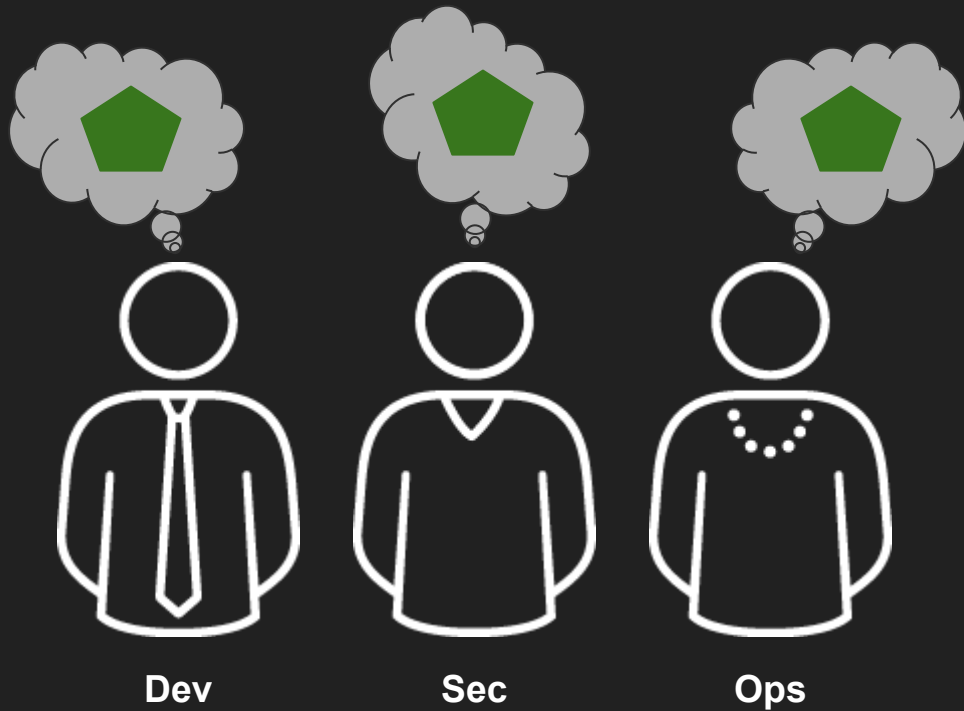


Dev

Sec

Ops

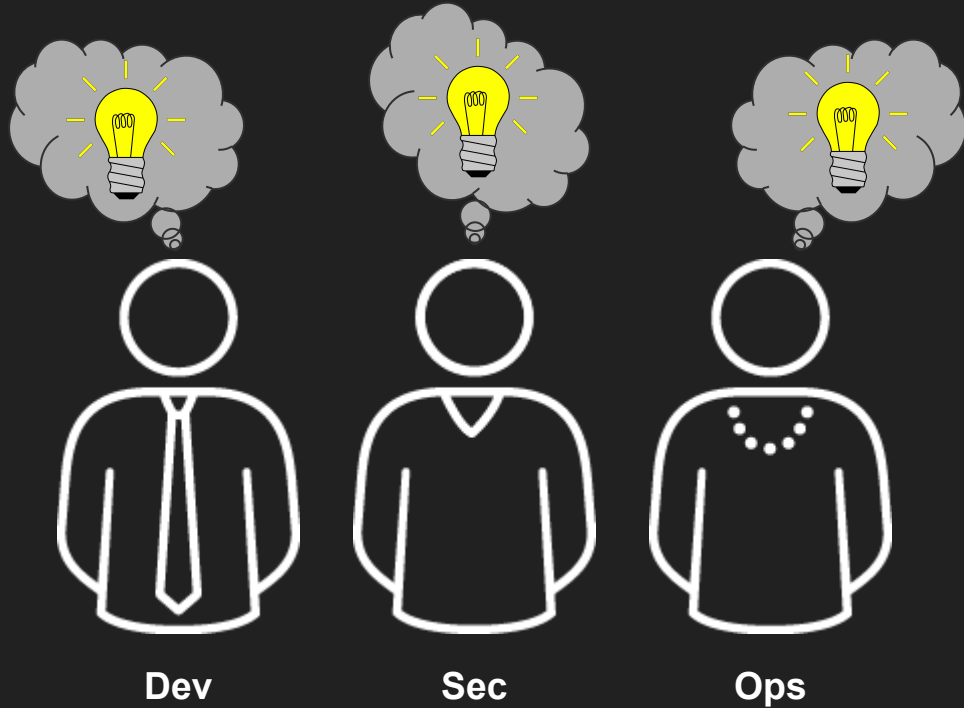




Dev

Sec

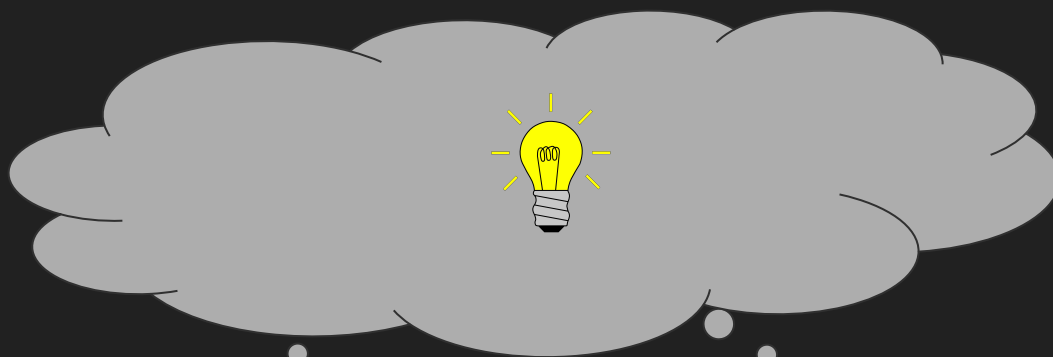
Ops



Dev

Sec

Ops



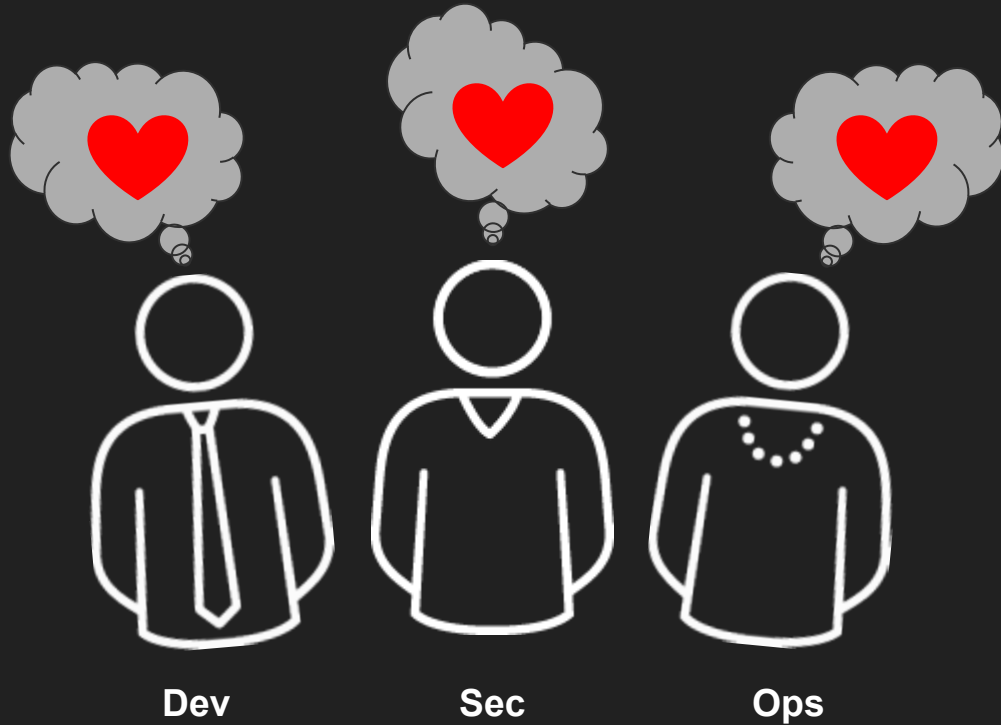
Dev



Sec



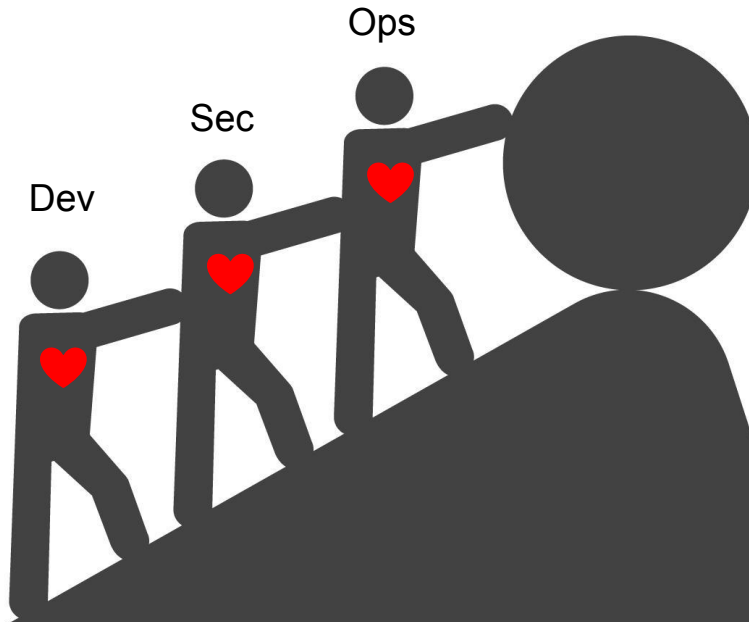
Ops



Dev

Sec

Ops



Shared goals

- App security
- Dev agility
- Observability
- Consistent configs
- Self-serve services
- Easy troubleshooting
- Relevant alerts
- Great docs
- Reliable infra

**Shipping faster,
smarter, and safer!**

[Free ebook](#)

Published by
**Red Hat
Developer**

A developer's guide to **setting supply chain security in DevSecOps**

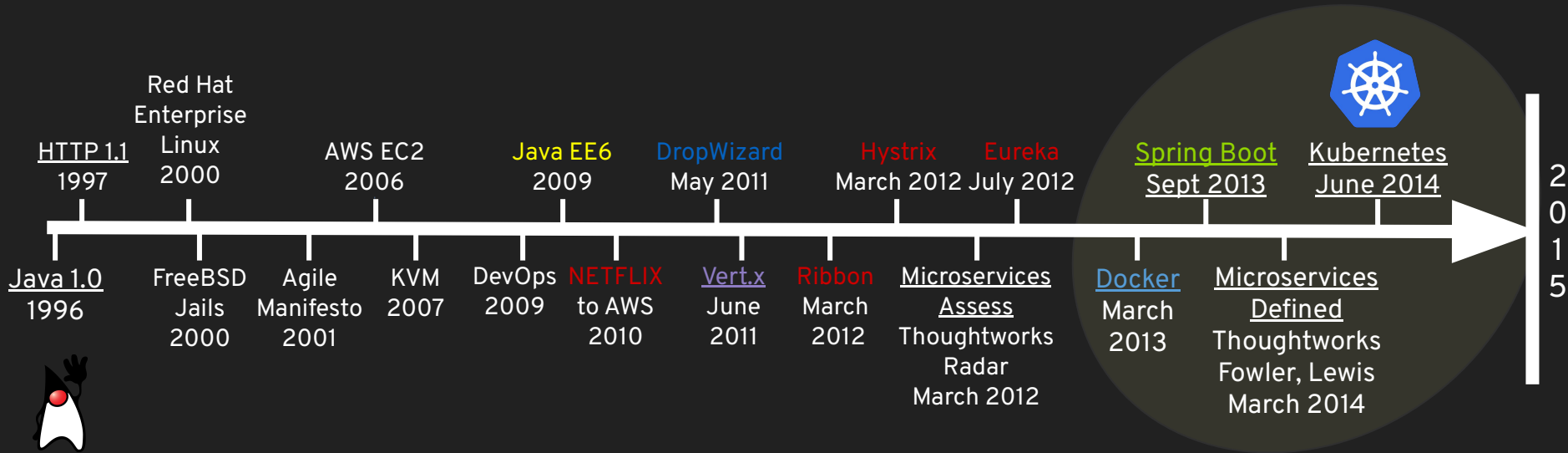
5 steps to building security early into your software development



Collin Chau, Dash Copeland, and Markus Eisele

History

Journey to Cloud Native





2015

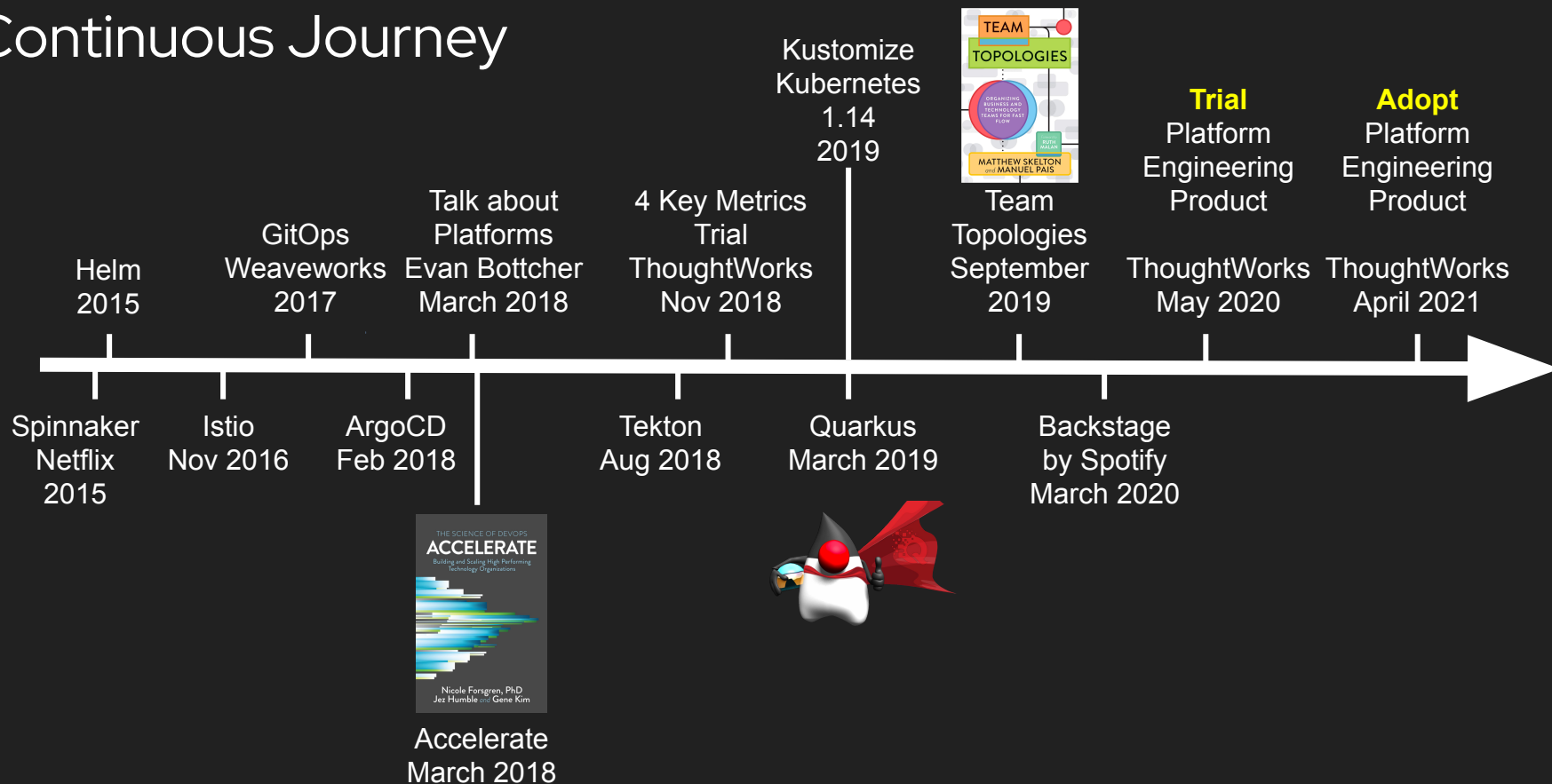
Launch 1000+
Containers
OpenShift v3



OPENSHIFT

<https://www.youtube.com/watch?v=GCtpncA0Fa0&feature=youtu.be&t=1031>

Continuous Journey



Techniques

Adopt

1. Applying product management to internal platforms
2. Infrastructure as code
3. Micro frontends
4. Pipelines as code
5. Pragmatic remote pairing
6. Simplest possible feature toggle

Trial

7. Continuous delivery for machine learning (CD4ML)
8. Ethical bias testing
9. GraphQL for server-side resource aggregation
10. Micro frontends for mobile
11. Platform engineering product teams
12. Security policy as code
13. Semi-supervised learning loops
14. Transfer learning for NLP
15. Use "remote native" processes and approaches
16. Zero trust architecture (ZTA)

Assess

17. Data mesh
18. Decentralized identity
19. Declarative data pipeline definition
20. DeepWalk
21. Managing stateful systems via container orchestration
22. Preflight builds

Hold

23. Cloud lift and shift
24. Legacy migration feature parity
25. Log aggregation for business analytics
26. Long-lived branches with Gitflow
27. Snapshot testing only

Hold

Assess

Trial

ThoughtWorks®

TECHNOLOGY RADAR

An opinionated guide
to technology frontiers

Vol.22

thoughtworks.com/radar

#TWTechRadar

Trial

Platform-as-a-Product

What is a Platform?

"A **digital platform** is a foundation of self-service APIs, tools, services, knowledge and support which are arranged as a compelling internal product. "

Evan Bottcher - 2018

"A **digital platform** is a foundation of self-service **APIs, tools, services, knowledge** and support which are arranged as a compelling internal product. "

Evan Bottcher - 2018

Portal (Backstage)

Golden Path Templates

DevX: Collaboration, Tutorials, Videos, Slack

Developer Environments

DBaaS, Kafka-as-a-Service, Cache-as-a-Service, SSO-as-a-Service, etc

CLI (kubectl, helm, git, oc), API

Observability (Ops/SRE view, Dev view, AppOps view, PE view)

Continuous Delivery

Artifact Storage

Security: Runtime scanning, build-time scanners

Continuous Integration Engine & Task Runners

Artifact Builder, Image Builder

Source Control

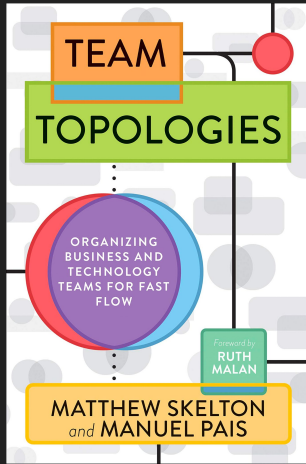
AuthN/AuthZ/RBAC

Compute/Network/Storage, Namespace-as-a-Service/Cluster-as-a-Service, Configuration Management

"A **digital platform** is a foundation of **self-service** APIs, tools, services, knowledge and **support** which are arranged as a **compelling** internal **product**. "

Evan Bottcher - 2018

For Whom?



Team
Topologies



Stream-aligned Team



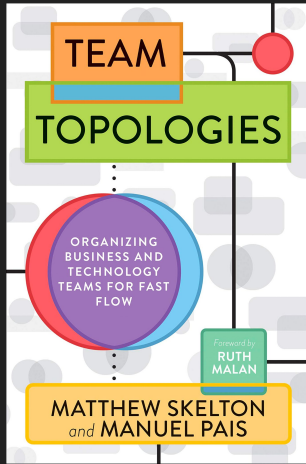
Enabling Team



Complicated
Subsystem Team



Platform Team



Team
Topologies



Stream-aligned Team



Platform Team

KUBERNETES HAS CROSSED THE ADOPTION CHASM

According to the most recent **State of Cloud Native Development Report**, developed for CNCF by **SlashData**, Kubernetes has demonstrated impressive growth over the past 12 months with **5.6 million developers** using Kubernetes today.

This represents a 67% increase from a year ago when, adjusting for a change in the question methodology, there were 3.9 million Kubernetes developers worldwide. This group now represents **31% of all backend developers** an increase of 4 percentage points in the last year.

Kubernetes' ubiquity is also supported by strong evidence from New Relic, which saw a 37% year-on-year increase in Kubernetes adoption based on accounts and a 49% year-on-year increase in overall container adoption. Based on these statistics and New Relic's **O11y Trends Report**, they believe the rise in adoption of Kubernetes and container-based platforms has increased the need for observability maturity amongst organizations looking to gain visibility into their complex application architectures.

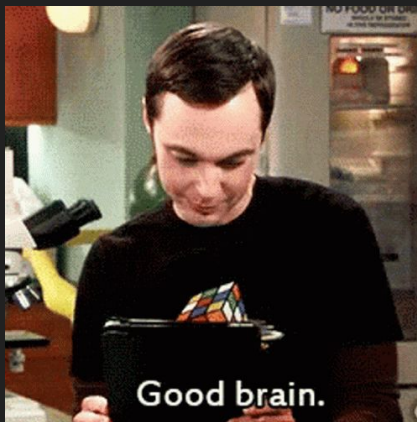
An interesting trend CNCF has seen develop is the correlation between Kubernetes and large organizations: respondents from organizations larger than 5,000 FTEs are far more likely to use Kubernetes than those working at smaller organizations. This is reinforced by findings from SlashData, who found Kubernetes usage skyrockets in large companies. Additionally, Datadog reported that larger companies (500-1000 and 1000+) are using Kubernetes more than smaller ones (<500).



 **CLOUD NATIVE**
COMPUTING FOUNDATION

ANNUAL SURVEY 2021

Cognitive Load



Overloaded

COBOL
JCL/WFL

Batch
Centralized

C/C++
4GLs
RDBMS/
SQL
Unix

Interactive
Distributed

Mouse
PushButton Click
DropDown List
Windows

GUI/Event-Driven

HTML
HTTP
CGI
GET/POST
Cookies
Java
Servlet
EJB
Solaris/AIX

Request/Response

MVC-Struts
DI-Spring
ORM-Hibernate
XML
WS-*
JSF
Agile
Automated Testing
CI
SVN
Linux

AJAX

Java EE 6
HTML5 (JavaScript)
iOS/Android
Phonegap/Cordova
Cucumber
REST
Maven/Gradle
git
MongoDB/Redis
Hadoop
DevOps
CD
Microservices

Asynch, Functional
Reactive

Java EE 8-12, NodeJS, Go
Serverless
AWS, Azure
Flutter, Typescript
Kafka, GCP
gRPC, Avro
Vue, React, Next.js
Serverless, Wasm
Terraform, Vault
CRI-O, Podman
ArgoCD, Tekton, CircleCI
Backstage, Helm
Kustomize
Istio/ServiceMesh

Cloud, Kubernetes



60's-70's



1985-1995



1993-1999



1995-2003



2003-2011



2010-2016



2018-2022

Database **Streaming & Messaging** **Application Definition & Image Build** **Continuous Integration & Delivery**

App Definition and Development

Orchestration & Management

Runtime

Provisioning

Special

Scheduling & Orchestration **Coordination & Service Discovery** **Remote Procedure Call** **Service Proxy** **API Gateway** **Service Mesh**

Cloud Native Storage **Container Runtime** **Cloud Native Network**

Automation & Configuration **Container Registry** **Security & Compliance** **Key Management**

Kubernetes Certified Service Provider **Kubernetes Training Partner** **Certified CNPs**

Platform

Certified Kubernetes - Distribution

Certified Kubernetes - Hosted

Certified Kubernetes - Installer

PaaS/Container Service

Observability and Analysis

Monitoring

Logging

Tracing

Chaos Engineering

Continuous Optimization

Serverless

Members

CD Foundation Landscape

CLOUD NATIVE LANDSCAPE

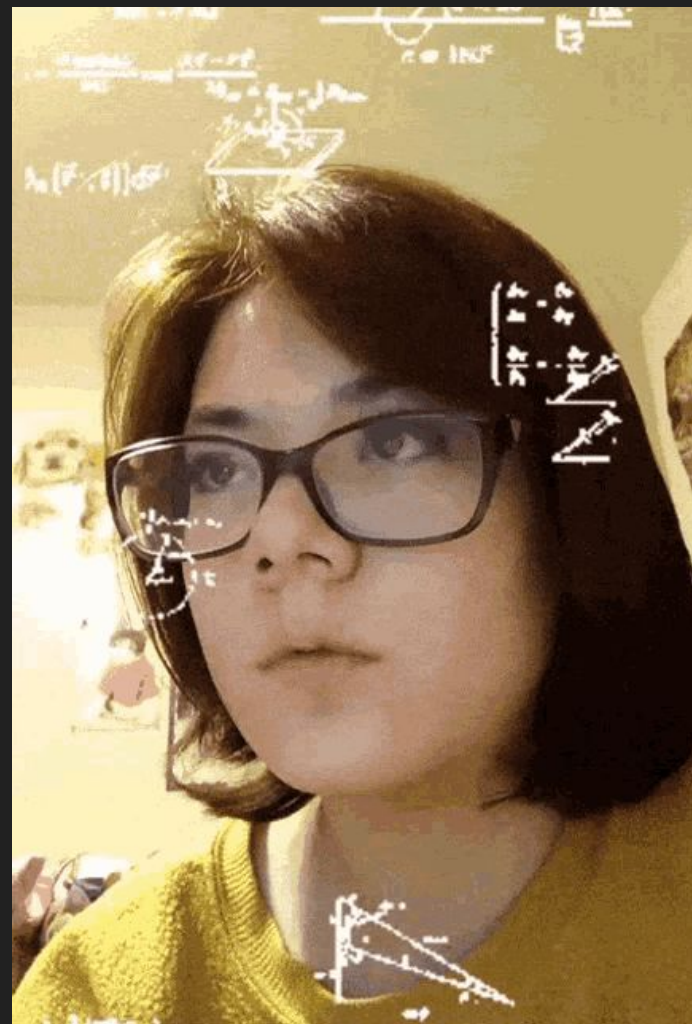
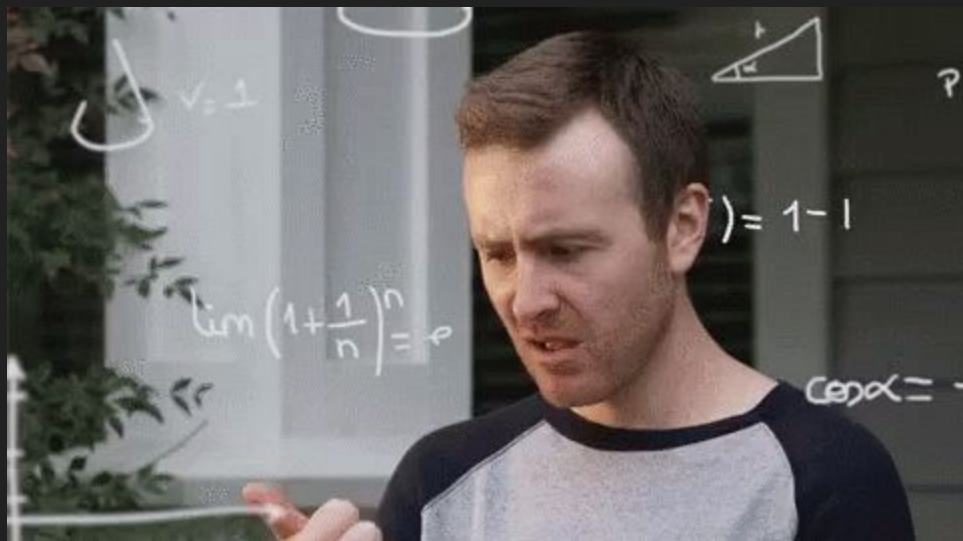
CLOUD NATIVE OPERATIONS

Logzai **Amplify**

Chaos Engineering

Continuous Optimization

l.cncf.io

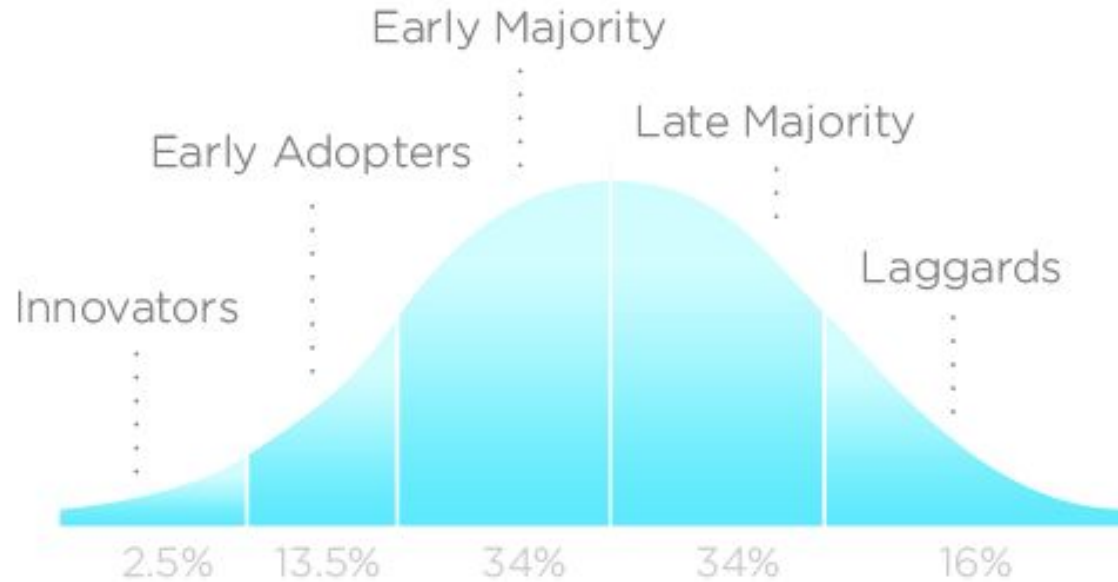


Compelling

"A **digital platform** is a foundation of self-service APIs, tools, services, knowledge and support which are arranged as a **compelling** internal product. "

Evan Bottcher - 2018

The technology adoption lifecycle is a sociological model that describes the adoption or acceptance of a new product or innovation, according to the demographic and psychological characteristics of defined adopter groups



INNOVATION ADOPTION LIFECYCLE

https://en.wikipedia.org/wiki/Technology_adoption_life_cycle

A BUSINESSWEEK BESTSELLER

One of *Inc.* Magazine's
Top 10 Marketing Books of All Time

CROSSING THE CHASM

MARKETING AND SELLING
DISRUPTIVE PRODUCTS TO
MAINSTREAM CUSTOMERS

3RD EDITION

WITH ALL NEW EXAMPLES FROM 21ST-CENTURY SUCCESS STORIES

OVER ONE MILLION
COPIES IN PRINT
WORLDWIDE



GEOFFREY A. MOORE

Author of Escape Velocity

Pathfinders

Builders

Innovators

Early Adopters

Early Majority

Late Majority

Laggards

Uber Geeks

Bright/Shiny

Productivity

Career

Builds from Source

Download a .zip
Or use your SaaS API

Tools Wizards
Examples Tutorials

Initial patterns & practices with obvious productivity gains

Move when Boss is sold

Uses Google for Q&A

Never attends conferences

Fork your project on github or start their own

Wants API docs

Reads, retweets, Recommends

Formal Training Classes

Demands Training (loves a week off from work)

Infrequently peruses tech books/blogs that Boss drops on their desk

Read your sources to learn your API

Participates in forums

Publishes Books, Blogs

Asks to Attend Conferences

Ideally an internal mentor assists with learning curve

Might go to conferences when Boss forces the issue

Attends Brown Bags for free lunch

Often considering another career path that has less churn

Causing Disruption

Contributes Writes Speaks

Robust Community is critical

Sometimes attends UGs

Delivers Brown Bag

Jobs - Adopt when it begins impacting career

Might just use MS Access to solve problem

Basic Maintenance of COTS

Working at Dot Coms or Startups

Working as consultants vendors next-gen enterprises

Owens their laptop

Greenfield Development at enterprises that see IT as a competitive advantage

Brownfield Development

does not select their tools, can not install

2.5%

13.5%

34.5%

34.5%

16.0%

Unicorn is a unicorn.



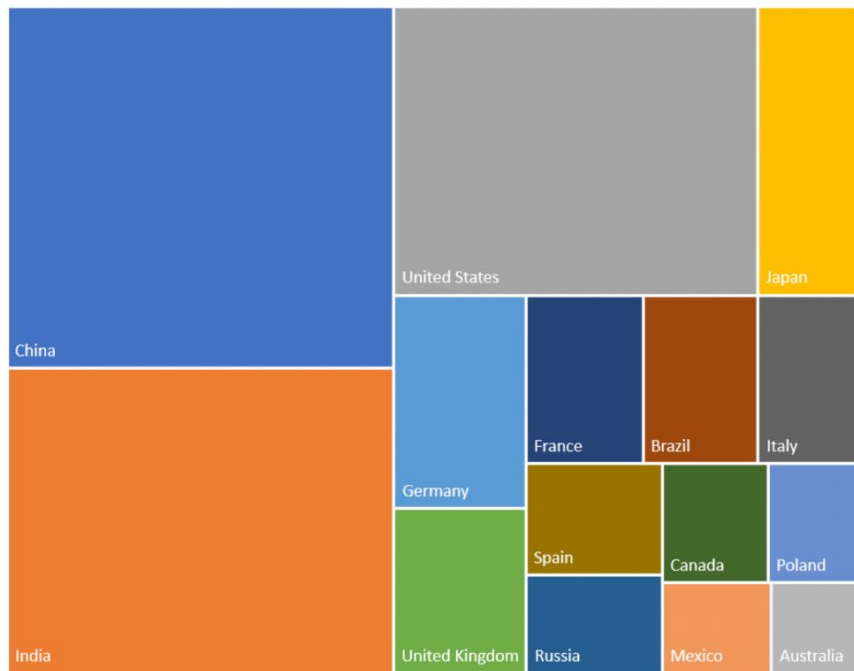
And Horse is, well . . .

not.

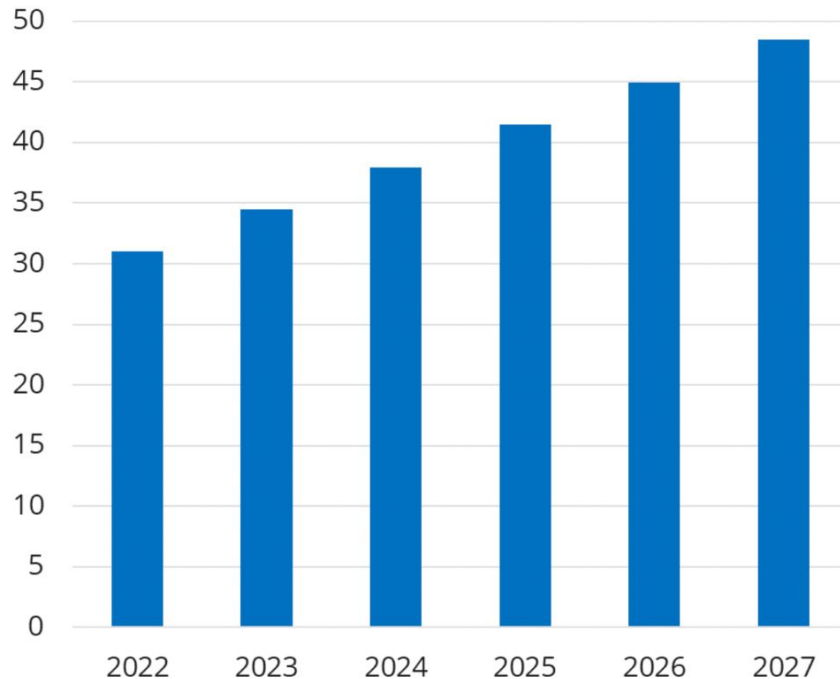


The Worldwide Developer Population is Forecasted to Grow from 31 Million in 2022 to 48.5 Million in 2027

Number of Developers Worldwide in Top 15 Countries Ranked by Developer Population

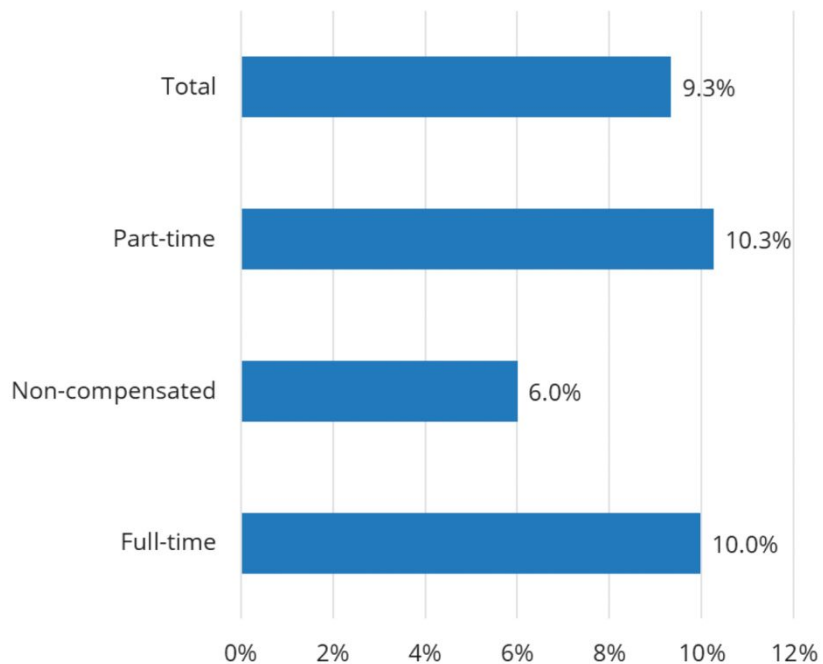


Number of Developers Worldwide (Millions)

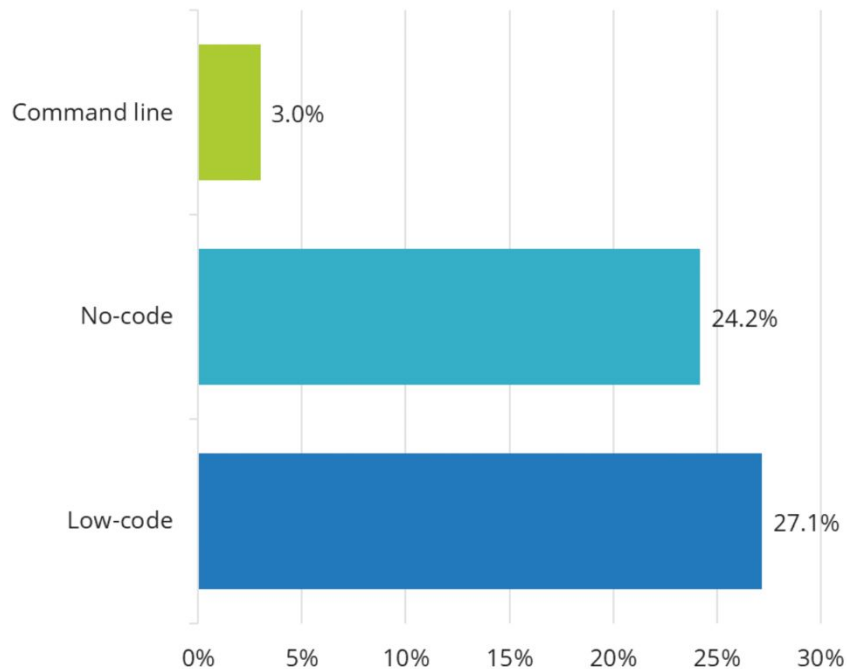


The CAGR of Low-code Developers Between 2022 and 2027 is 9x that of Command-Line Developers

CAGR for Total, Full-time, Part-time and Non-compensated Developers: 2022 to 2027



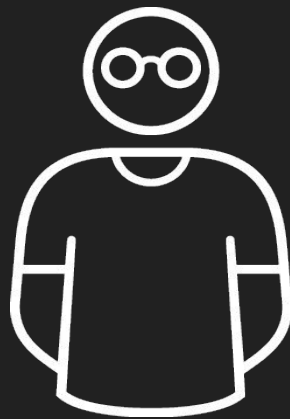
CAGR of low-code, no-code, command-line and overall developer population



Becoming the Developer's Developer



Product
Manager



UX



Developer
Advocate
(DevRel)

4 Risks

<https://www.svpg.com/four-big-risks/>

Feasibility

Usability

Viability

Value

How

Focus on the Story

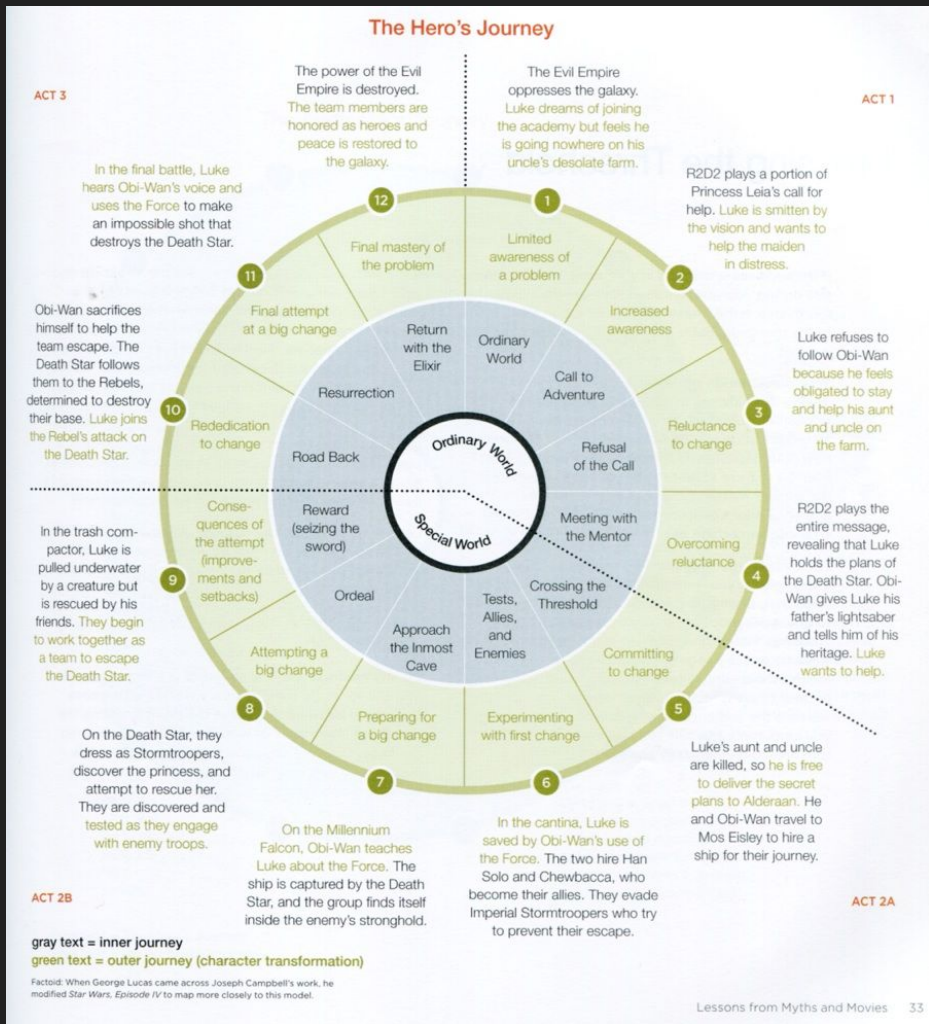
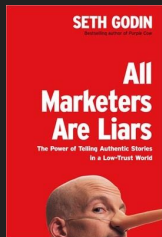
Repeating a strong message
on theme, on topic
is power

How: Duarte Storytelling

Nancy Duarte

Joseph Campbell

George Lucas



gray text = inner journey
green text = outer journey (character transformation)

Factoid: When George Lucas came across Joseph Campbell's work, he modified Star Wars, Episode IV to map more closely to this model.



"No! Try not. Do. Or do not. There is no try."





"It is not despair, for despair is only for those who see the end beyond all doubt. We do not."

@bursutter bit.ly/devsdev

Developers are Creators

In a world that is digitally transforming,
digital makers are the catalysts



I made it work

I made
something
new & cool

I bent it to my will

Be the Producer, not
just the Player

Be the Creator,
not just the
Consumer

Be a
Maker

Be a Builder

Empower Energize
Educate Enable

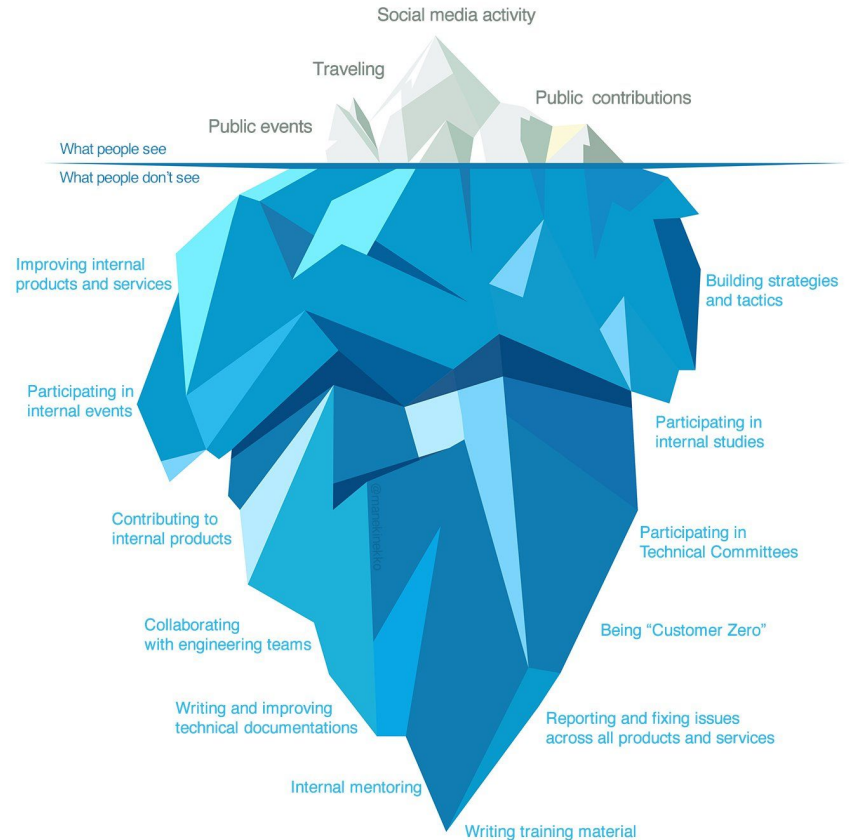
DevRel

- Welcome Email
- Slack/Discord Chat
- Stack Overflow Enterprise
- Email List
- Monthly Newsletter
- Getting Started Guide (per programming language)
- Written Tutorials for each major service (e.g. MongoDB-as-a-Service)
- Written Tutorial for each onboarding tool: S2I, Templates, Helm Charts, ArgoCD, DevFiles
- Video-based Tutorials of the same
- FAQ
- Blogging Engine
- API Explorer
- Reference Documentation for CLIs, APIs
- [Golden Path as Product Templates](#)
- Portal for all of the above - one-stop shop

@bursutter bit.ly/devsdev

THE DEVELOPER ADVOCATE ICEBERG

Wassim Chegham | @manekineko



backstage.spotify.net/create-component

Create... ☆

Filter

Backend Service #backend-support

fabric
Apollo Service

Golden Path

Backend service template based on [Apollo Standalone](#).
Usually the best choice for consumer-facing services.
Includes support for gRPC, Hermes, Http, deployment (Kubernetes and Helios), logging and metrics.

CHOOSE

java
Java Library

Experimental

Java library template following the [Java Library Best Practices](#).
Use this for creating a new repository for your library.
If you want to add a library to an existing repository, consider using the [java-library-archetype](#) instead.

CHOOSE

mcf
Spring Boot Service

Experimental

Backend service template based on the [Spring Boot Spotify Support Project](#).
This template is great for anyone looking to use open source tooling inside of the Spotify ecosystem.
It comes equipped with reactive and non-reactive GRPC, Hermes, Metrics, Logging, and Nameless setup.
You can find a guide [here](#). You will also find a ton of example projects [here](#).

CHOOSE

python
Python Library

Experimental

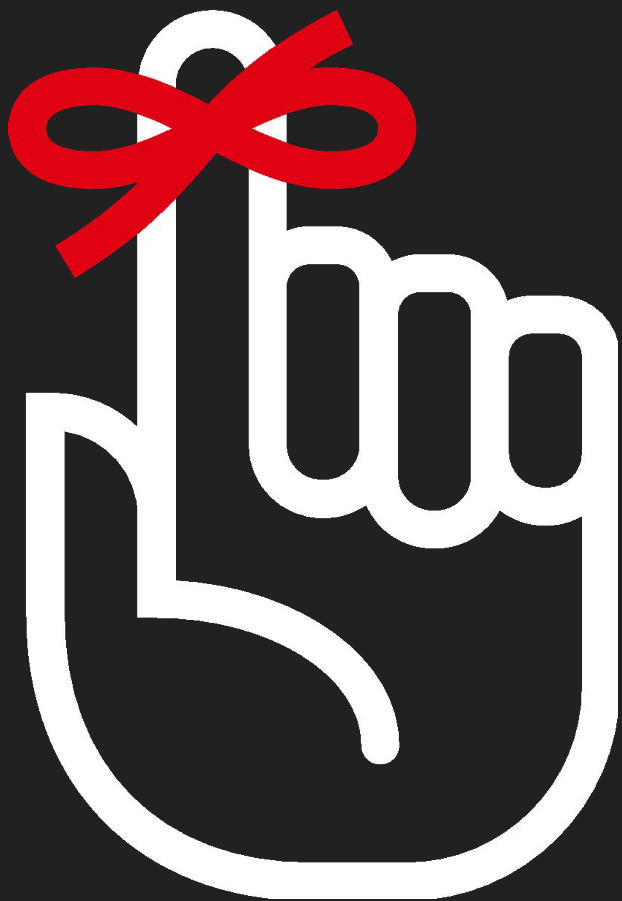
Template for an internal Python Library. Includes support for building on Tingle, deploying to our internal PyPI, and (optionally) build & upload a Debian package.

CHOOSE

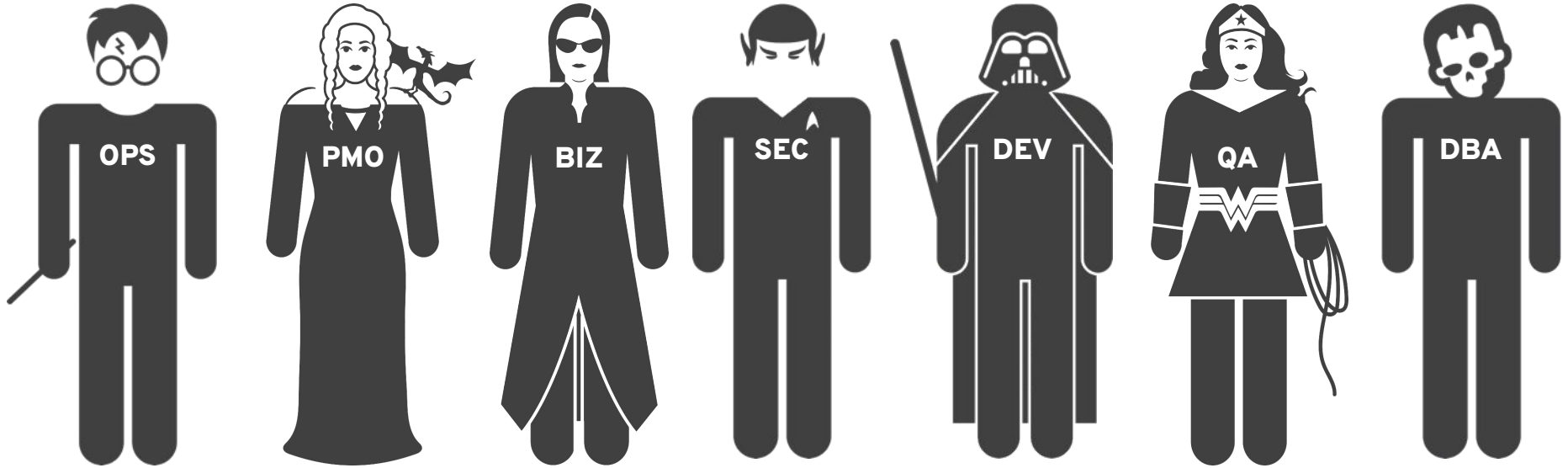
set-tools

setadel

adstudio

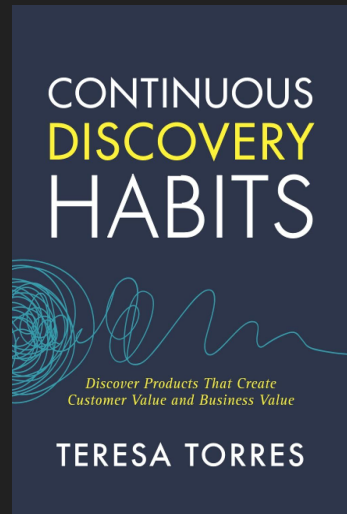
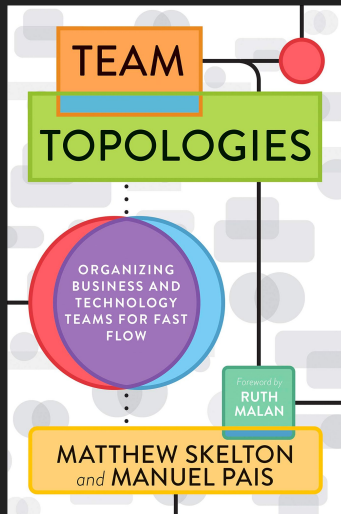
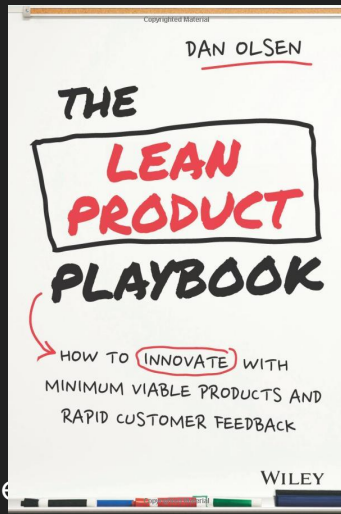
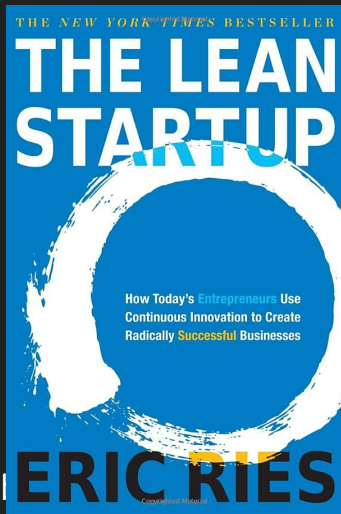
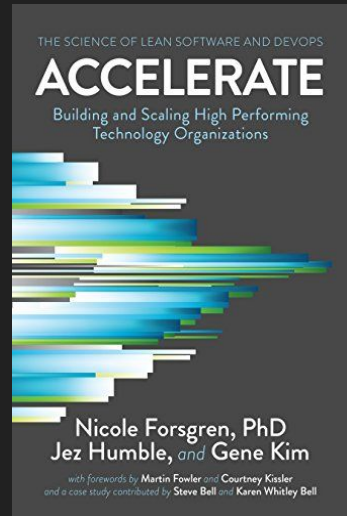
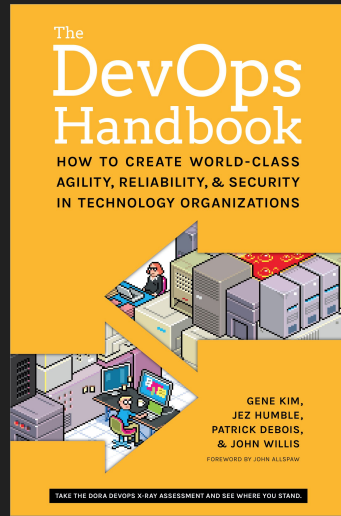
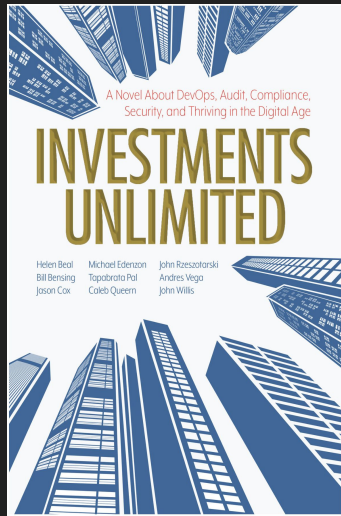
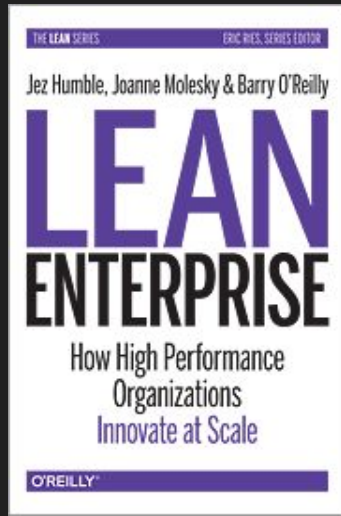
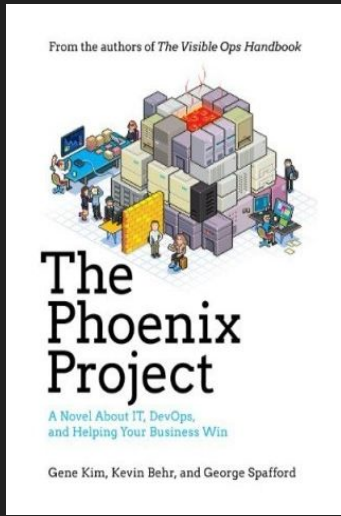


Focus on Team



Software is a Team Effort

Continuous Improvement



The End