Octopus Deploy

What's this DevEx and Platform Engineering I keep hearing about?

Bob Walker | Field CTO | Octopus Deploy

@bobjwalker

bob.walker@octopus.com

@bobjwalker

Bob Walker

Field CTO
Octonaut since 2018

@bobjwalker

bob.walker@octopus.com

@bobjwalker

DTN

Farm Credit Services of America

Sandhills Global

ELECTION Systems & Software

west®

Octopus Deploy

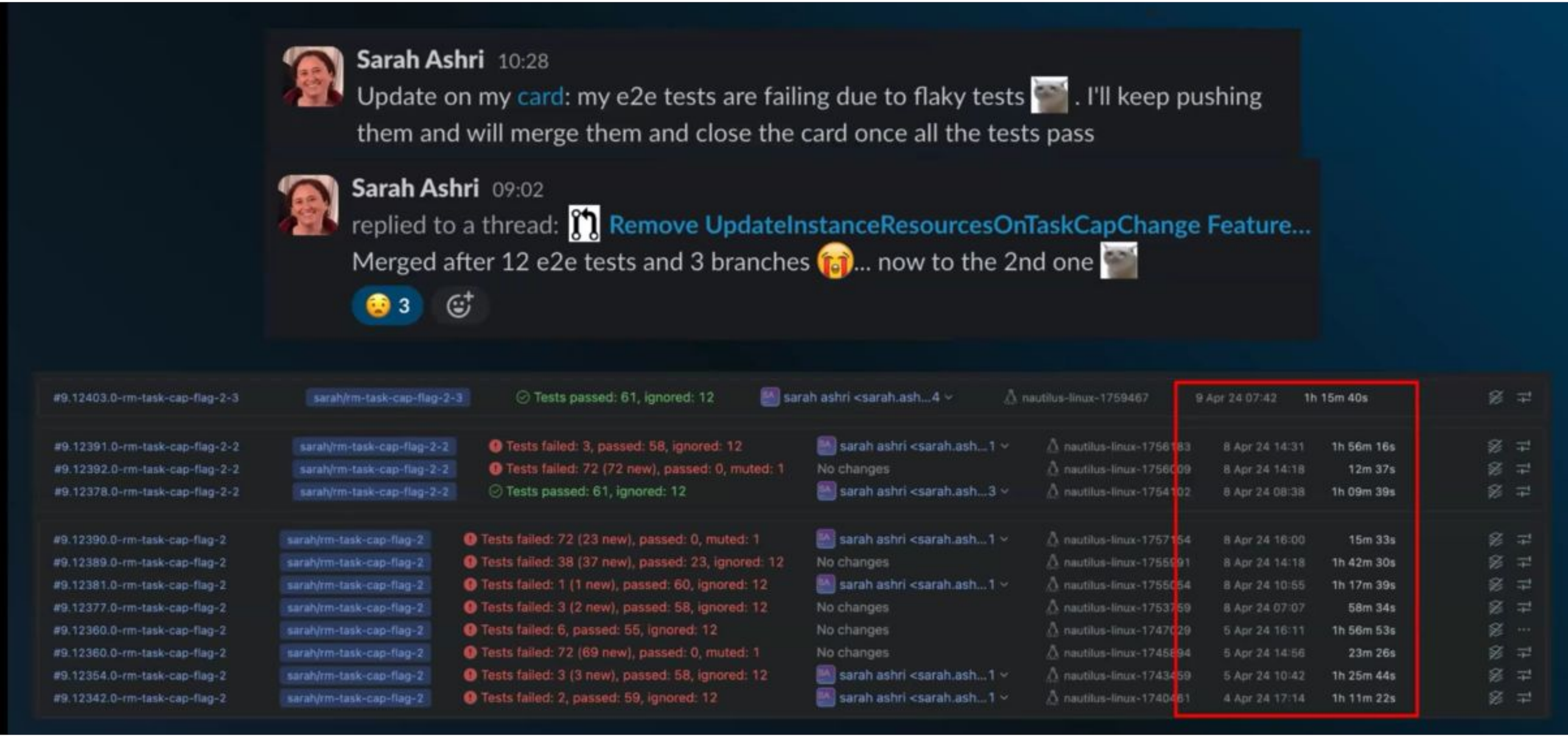
Session is for anyone who has minimal knowledge of DevEx and Platform Engineering

Agenda

- Intro to Developer Experience
- Intro to Platform Engineering
- Paved vs. Golden Paths
- Why now?
- Avoiding Pitfalls
- Measuring success
- Resources

Intro to Developer Experience (DevEx)

Multi-Day Effort to Merge to Main



Another Term: Yak Shaving

THE TERM YAK SHAVING REFERS TO THE SEEMINGLY ENDLESS SERIES OF SMALL TASKS THAT HAVE TO GET DONE BEFORE A PROJECT MOVES FORWARD.

Image Source: https://www.cryan.com/daily/20230725.jsp

Developer Experience

You can tell a lot by a company based on *the onboarding experience* and *how hard it is to get to production*.

Developer Experience

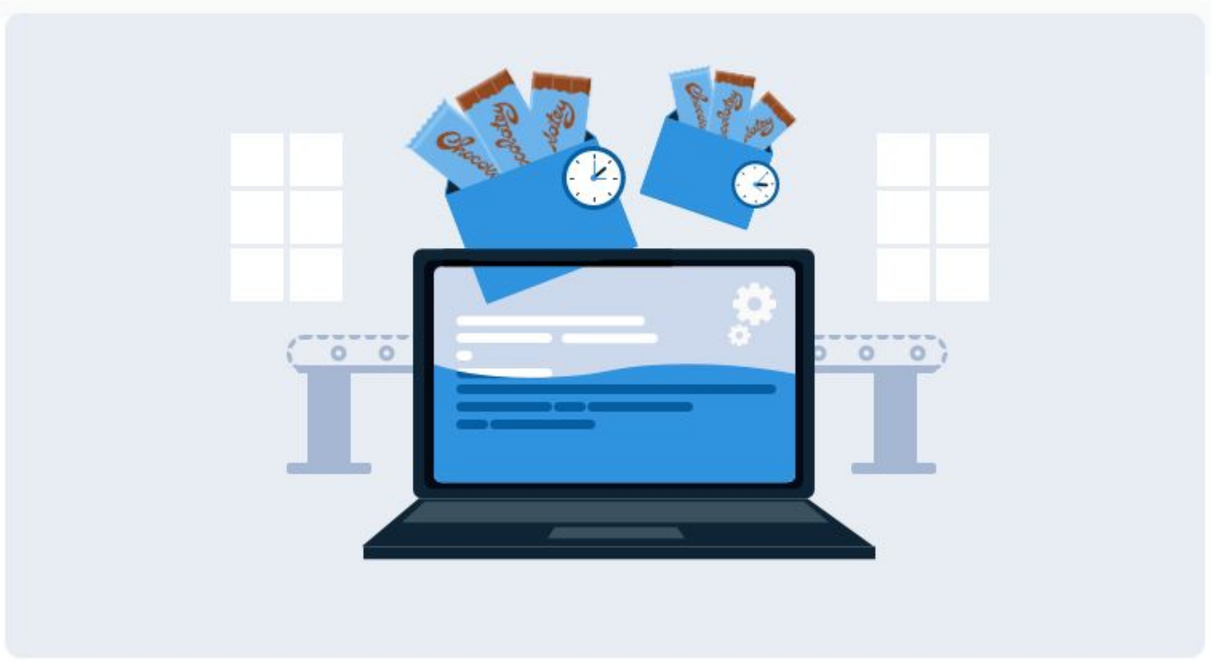
The systems, technology, process, and culture that influence the effectiveness of software development

Quote Source: https://github.blog/2023-06-08-developer-experience-what-is-it-and-why-should-you-care/

Developer Experience

It gives a name to something senior people and leaders have been partially focusing on for years

Example: Automating developer machine setup



Automating developer machine setup with Chocolatey

Bob Walker

August 26, 2019 • 8 mins

Image Source: https://octopus.com/blog/automate-developer-machine-setup-with-chocolatey

DevEx People Impact

- **Productivity**: how quickly or simply a change can be made to a codebase
- **Impact**: how frictionless it is to move from idea to production
- **Satisfaction**: how the environment, workflows, and tools affect developer happiness

Quote Source: https://github.blog/2023-06-08-developer-experience-what-is-it-and-why-should-you-care/

Tangible Impact

- New people
  - The F5 Contract
  - Time to first commit
- Feature work
  - Lead time for changes
  - Deployment frequency
  - Change failure rate
  - Mean time to recovery
- Engineering NPS

Tangible Impact

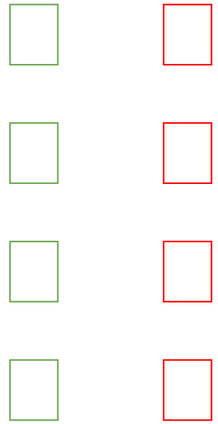
- New people
  - The F5 Contract
  - Time to first commit
- Feature work
  - Lead time for changes
  - Deployment frequency
  - Change failure rate
  - Mean time to recovery
- Engineering NPS

Ideal DORA Metrics

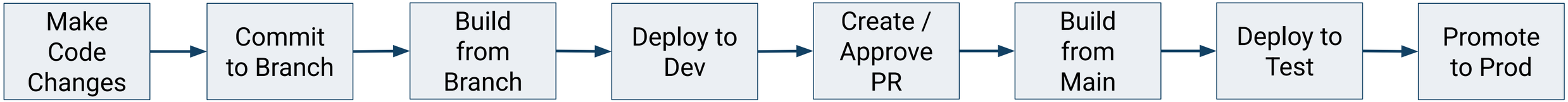
- Lead Time for Changes (commit to prod) 
- Deployment Frequency (to production) 
- Change fail percentage (failures in production) 
- Time to recovery (fix changes in production) 

Yak Shaving DORA Metrics

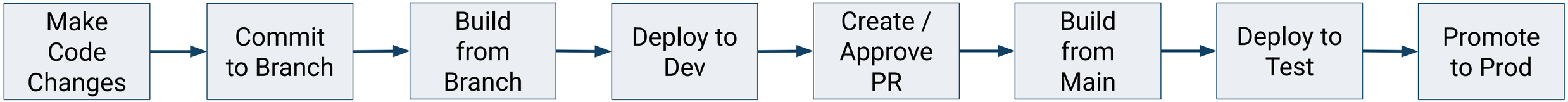
- Lead Time for Changes (commit to prod)
- Deployment Frequency (to production)
- Change fail percentage (failures in production)
- Time to recovery (fix changes in production)

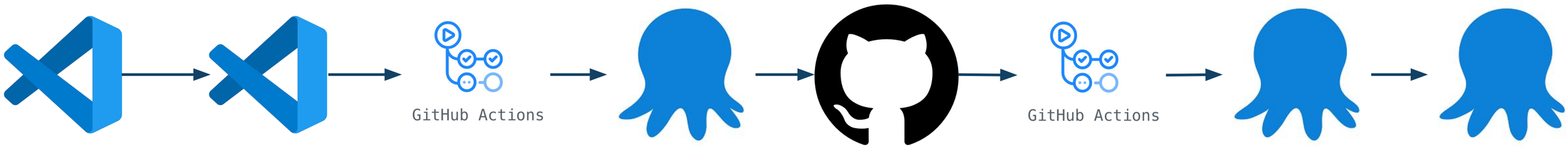


Typical Workflow

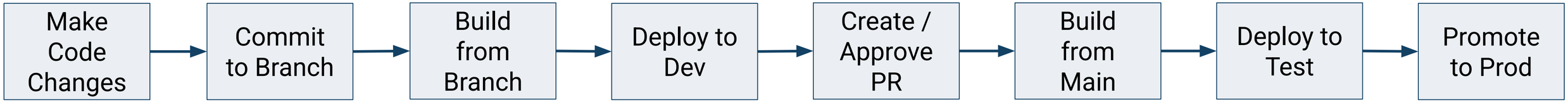


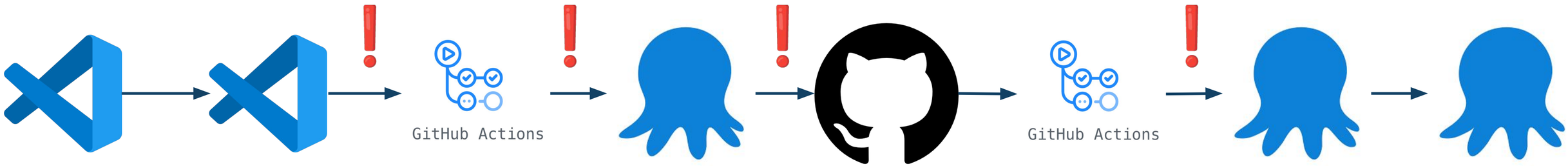
Typical Workflow



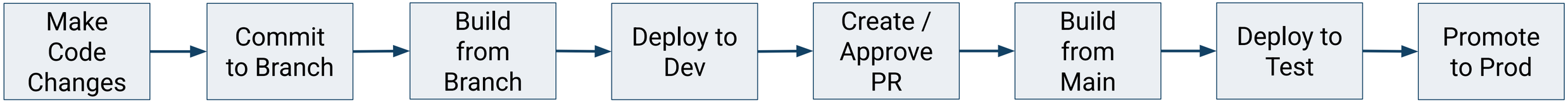


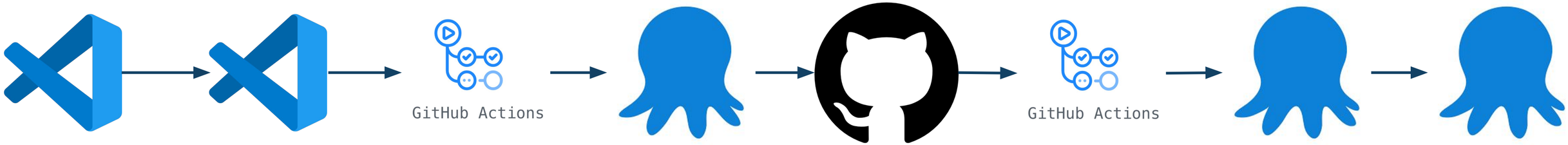
Typical Workflow



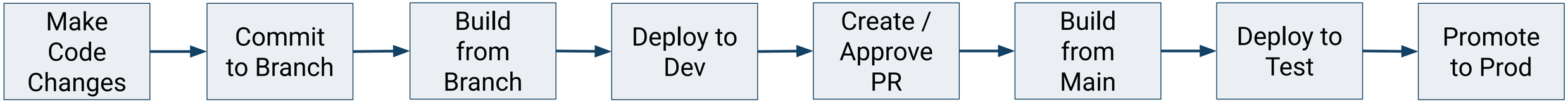


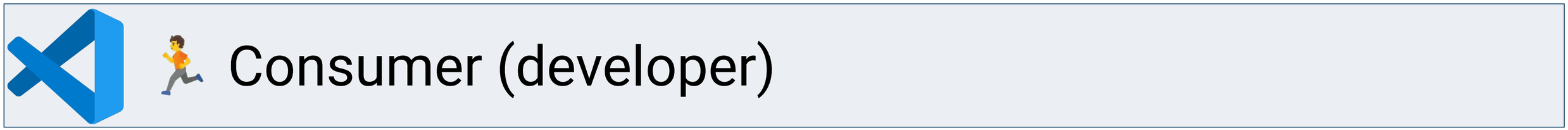
Typical Workflow

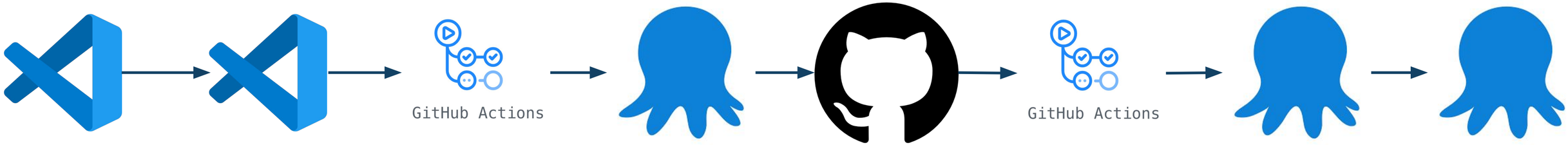




Typical Workflow







DevEx is the responsibility of multiple people and teams

Internally at Octopus

- **R&D Leadership Team (RLT):** The senior people in R&D
- **Individual Teams:** Improving the developer experience for their corner of Octopus.
- **Foundations Group**: Make Octopus simpler and safer to make changes to
  - Backend Foundations
  - Front-end Foundations
  - Build and Delivery
- **Solutions Engineering:** Demo space creator

Developer Experience

It’s not just automation, some of it can be solved by better documentation

New People

🏃 Getting started

> ✋ Make sure you've read the Contributing guide before continuing.
>
> 📣 Please make some noise in #team-backend-foundations-requests if something in this guide isn't working correctly.
>
> 😎 If you discover a problem please update the guide to help the next person!
>
> 📖 Want to edit one of the images in this guide? Click here.

Welcome to Octopus Deploy! We're excited to have you join us. This guide will help you get set up and oriented with Octopus Deploy, and the development environment. If we have done a good job, you will achieve all these goals in under two hours.

1. You will run the latest public release of Octopus Server with a single command then create and run an operations runbook
2. You will orient yourself with the development environment by querying the HTTP API and SQL Database, inspecting some logs and other files, and making a code change
3. You will have a fully capable development environment
4. You will make some simple code changes and see their effects

In this first part of the guide, you will install the necessary tools and set up your development environment.

Pre-requisites

1. Ensure you're operating system is up to date.
2. Install git and authenticate with GitHub.
3. Install Docker Desktop.
    - Due to licensing of Docker Desktop, please make sure you follow the Docker Hub guide in Confluence to ensure you have the correct license.
4. Install the latest version of the .NET SDK.
5. Install the latest LTS version of NodeJS.
6. Install pnpm.

Feel free to install these using your systems package manager, or by downloading them from the links provided. Now would also be a good time to download and other tools you'd like to use such as git clients, IDEs, etc.

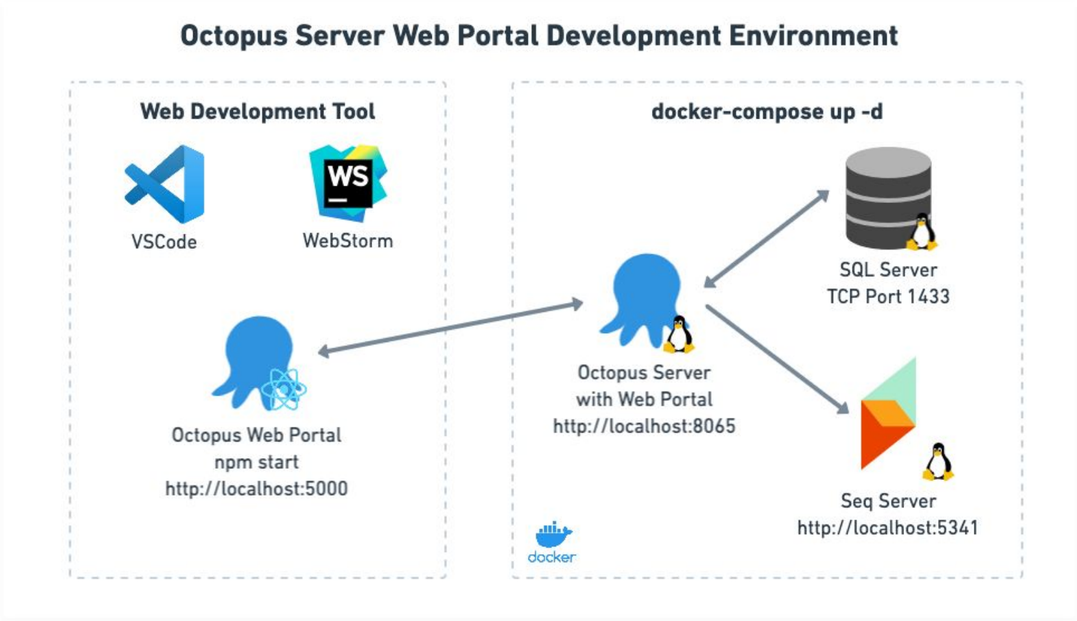
Platform Support

If you have a new developer asking which platform they should use, are considering switching platforms, or otherwise want guidance on how to deal with a platform difference, please read Developer Platform Choices and Support.

The F5 Contract

Octopus Web Portal Development Environment

This section is where we will run the Octopus Server web application in a hybrid native/docker mode and start making some code changes and seeing their effects.



This guide should be enough for you to get started with the Octopus Web Portal development environment. For a more in-depth guide, see the Octopus Web Portal development guide.

We will start with web application development so you can make some code changes and see them take effect. We will install `npm`, make some simple code changes, and see their effect.

Step 1: Compose up the Octopus Server development stack

These steps will stand up the Octopus Server stack using the code on your computer from GitHub. In the next steps, you will stop the service you want to develop and debug, leaving the rest running as-is.

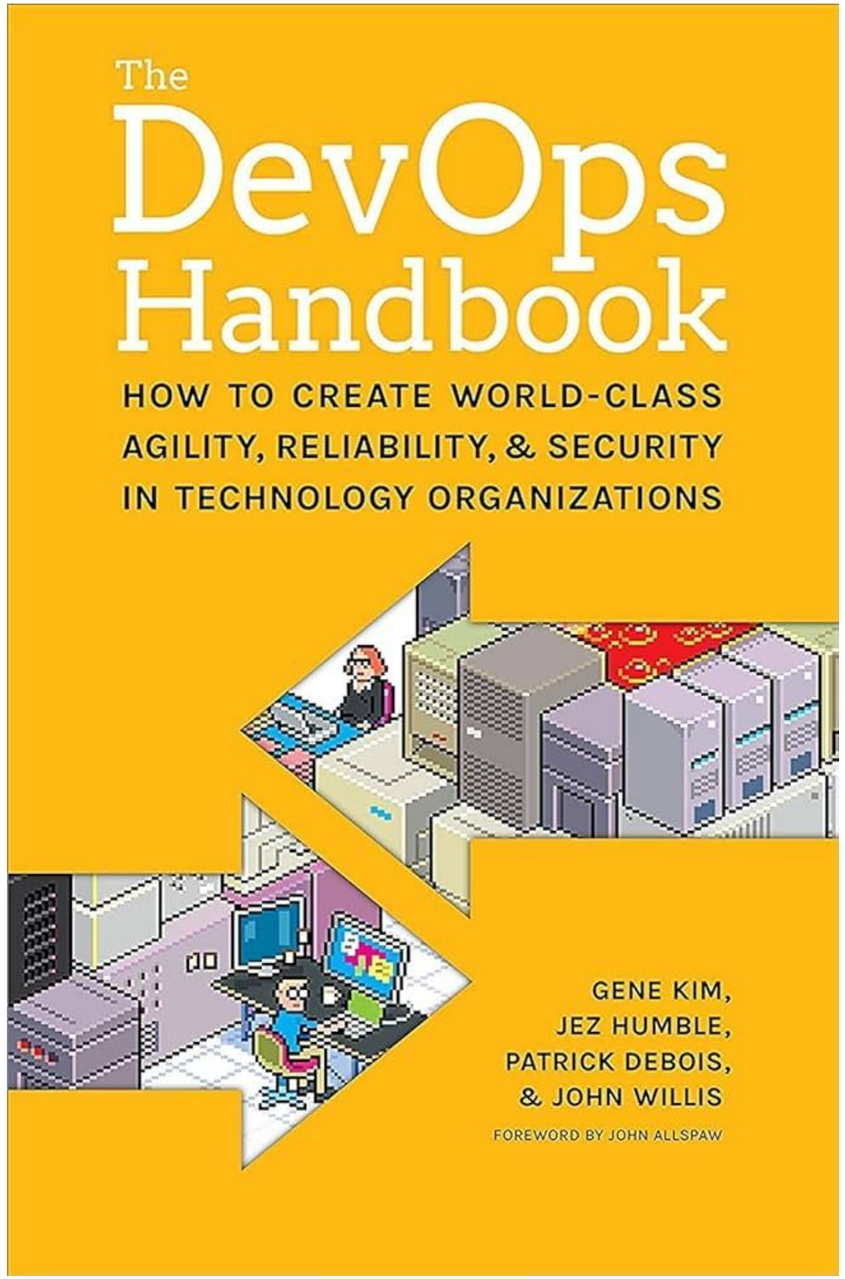
1. Compose up the Octopus Server development stack using the `docker-compose.yml` file.

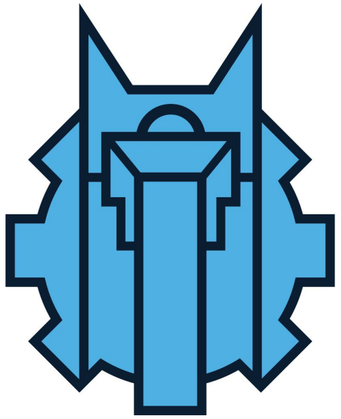
```
docker compose up -d
```

Step 2: Build and run the Octopus Server web portal

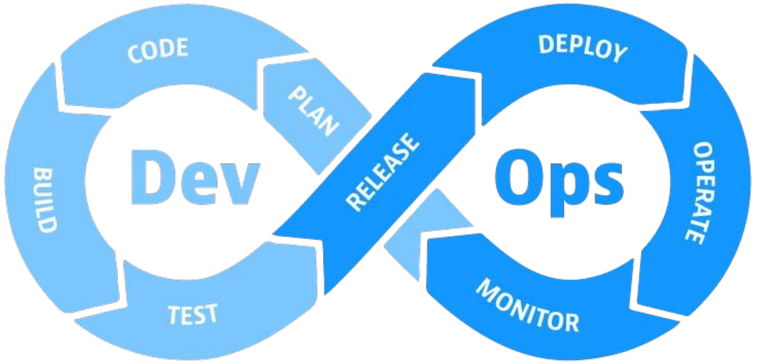
Intro to Platform Engineering

DevOps has been really successful!





Nashville
DevOps
Days 2024



Three Stages of DevOps

- **Divided:** Pre-DevOps, there are separate silos for development and operations with conflicting goals
- **Aligned:** Aligned Goals and objectives to remove conflict between teams
- **Combined:** Removing metaphorical walls allows cross-functional teams to build and run software

Combined Stage

You build it, you run it

Platform Engineering

Helps you get the benefits of *DevOps at Scale*

Platform Engineering

Build self-service internal developer platforms (IDPs) to manage infrastructure, software delivery, and operation tasks.

Quote Source: https://octopus.com/devops/platform-engineering/

Platform Engineering

Helps improve the developer experience.

Platform Engineering is not the only way to improve DevEx

Image Source: https://getyarn.io/yarn-clip/e68c17ca-35e6-4352-bb84-ac0f00881cd1

Paved vs. Golden Paths

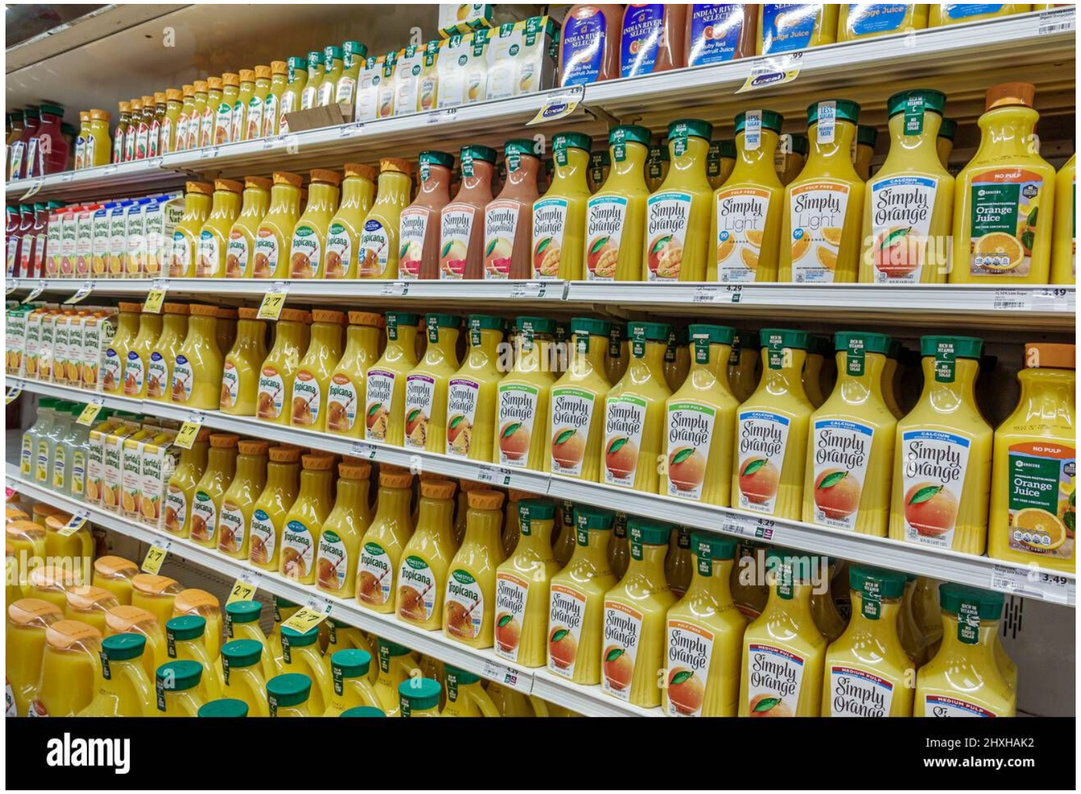
Paved Path


YOU MEAN
THERE'S A BETTER WAY?

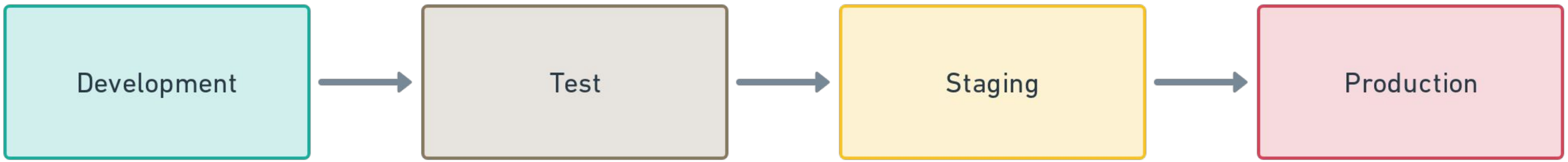

Fresh Squeezed
ORANGE JUICE
2.99 4.99 8.99
Fantastic
Juice
fruit-to-go
Sunlype
Enter to win every week!
$50

Golden Path

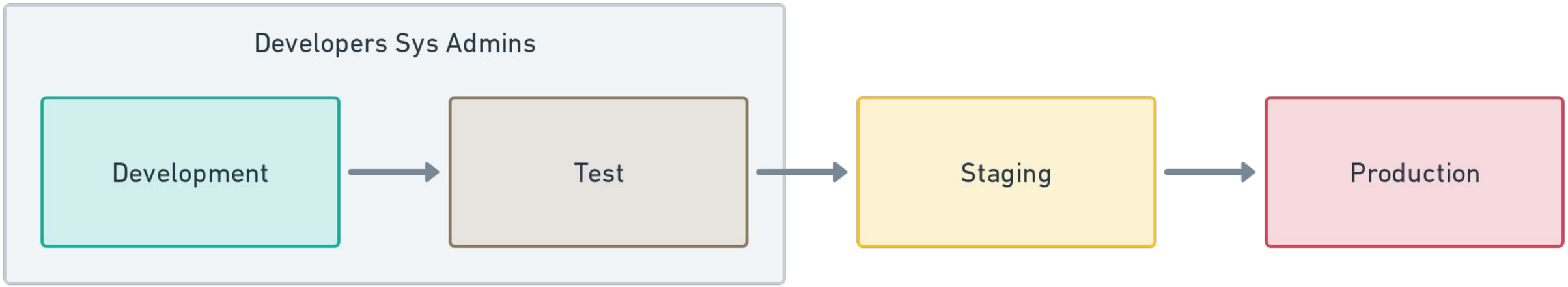
- Reduce burden and remove pain points.
- Solves specific problems developers are encountering.
- May involve automation, or just a new way of doing something.
- Better than what developers currently do, and is easy to adopt.
- Not forced onto a team, it is too enticing to not use it.



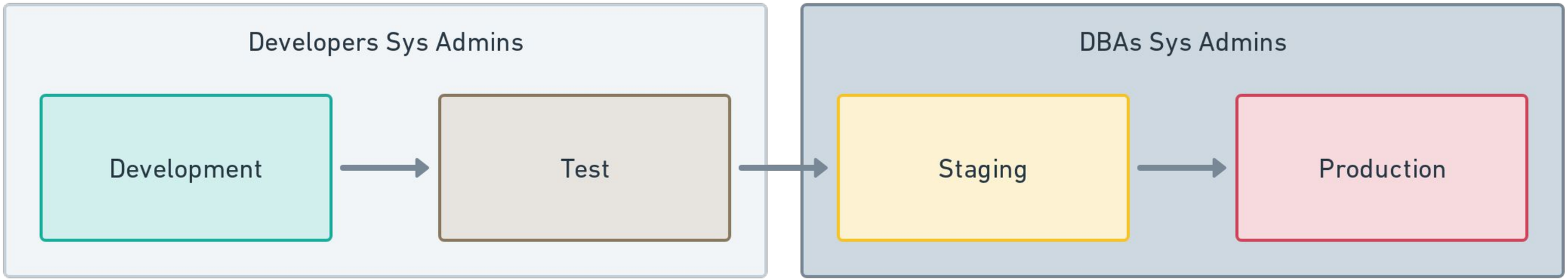
Paved Path Example



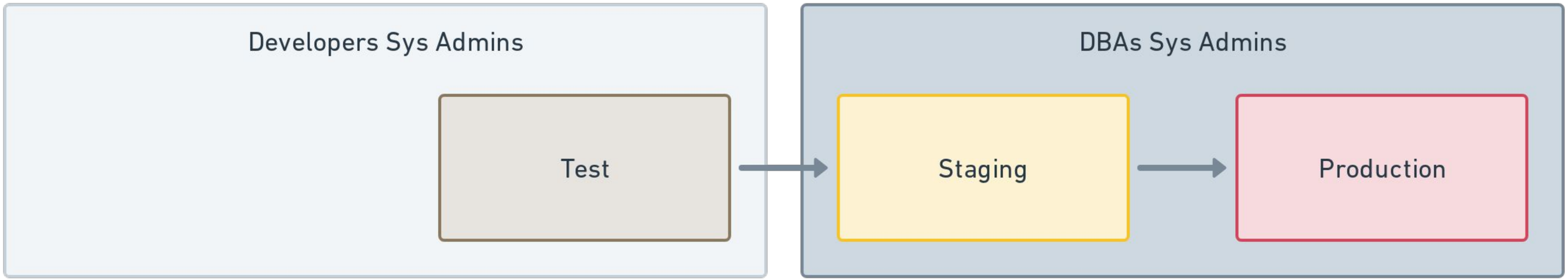
Paved Path Example



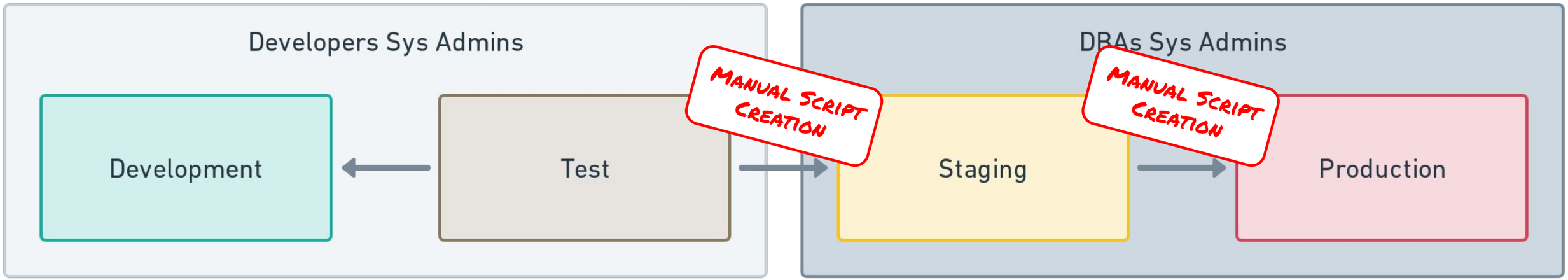
Paved Path Example



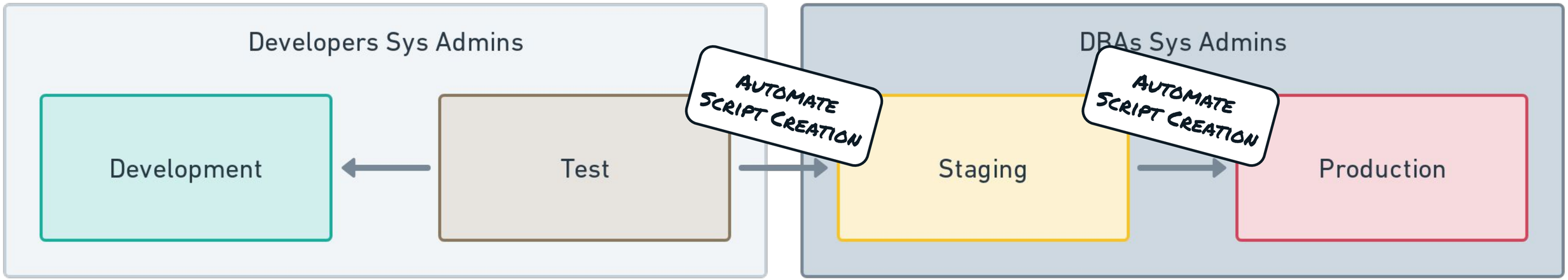
Paved Path Example



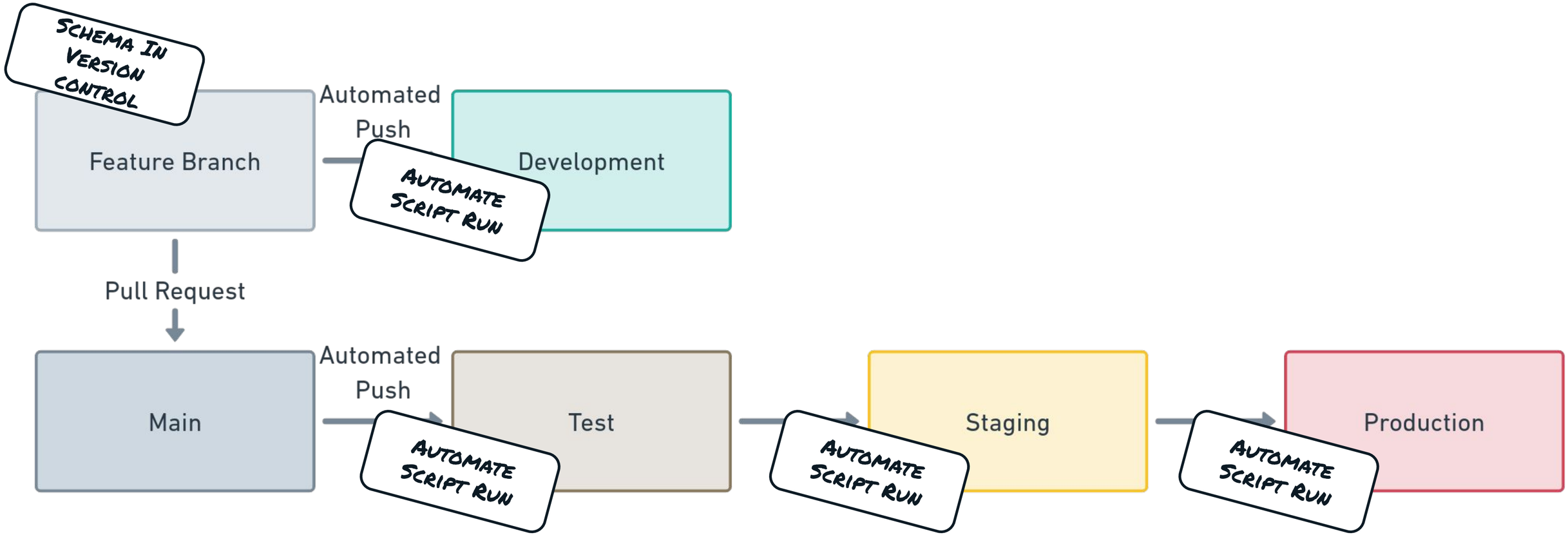
Paved Path Example



Paved Path Example



Golden Path Example



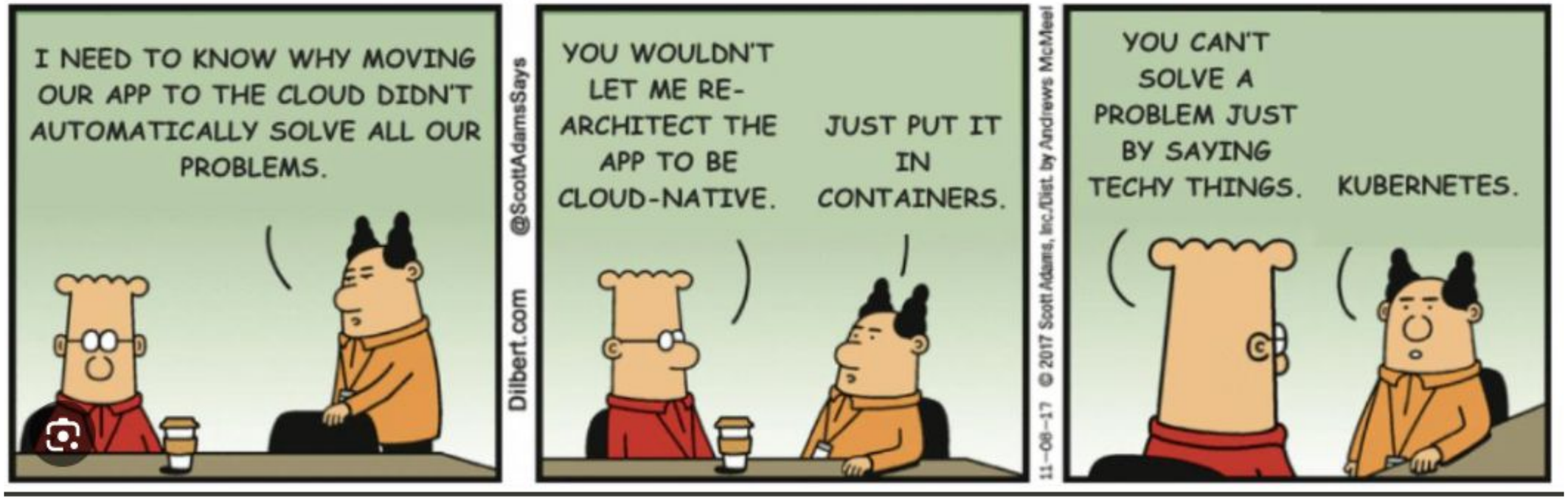
Why Now?

Developer Experience

Many people have been doing this on the side for years without calling it DevEx.

What about Platform Engineering?

Digital Transformations



Kubernetes required to learn for first deployment

- Nodes
- Pods
- ReplicaSets
- Secrets
- Load Balancer / ClusterIP / NodePort
- Ingress Controller / Ingress Rules
- Namespaces
- Deployments
- Services
- Service Accounts and Tokens
- Service Roles and assignment

Containerization requires a mindshift

- SDLC
  - All Dev and Testing must be done in containers
  - Containerize an application code base
  - Containers are immutable - any changes require a rebuild
- Production Workloads
  - Containers can fail and restart for any number of reasons
  - Only worry about how many replicas are needed, not which “VM” it is hosted on
  - Cannot “remote in and debug” like a person could with VMs

Very easy to misconfigure something

The Chicago Way: An Electronic Voting Firm Exposes 1.8M Chicagoans

UpGuard Team

Published Aug 17, 2017

Image Source: https://www.upguard.com/breaches/cloud-leak-chicago-voters

Digital Transformations

To have a successful digital transformation you have to invest in DevEx and Platform Engineering

Avoiding Pitfalls

Platform Engineering isn’t operations 2.0

IDPs are not a shared service

- Shared service - developers raise tickets and wait for someone to do work.
- IDPs
  - Enable developers to do work.
  - Create the processes for repeatable tasks.
  - Make it easy to spin up and create new projects.
  - Work with all parties to ensure processes are created to meet standards.
- Operation work is still required - not the purview of platform engineers
  - The application hosts are up and running.
  - Monitoring and logging.
  - Security and auditing.

Treat IDPs like a product

The developers and operations are the customers

Build Golden Paths not paved paths

IDPs should be open source

Allow developers and operations to contribute to it.

IDPs are the open source

- Doesn’t mean you need to use an open source product.
- Make the processes themselves open source.
  - Publish clear contribution guidelines
  - Keep documentation up to date
  - Guide contributors to solve problems with appropriate abstraction
  - Help contributors to use existing features to solve their problems

IDPs should be easy to use

The higher the friction, the slower the adoption.

IDPs can’t solve company culture problems

No tool on earth can fix “I don’t trust developers.”

Measuring Success

You have to measure to know the impact

Tangible Impact

- New people
  - The F5 Contract
  - Time to first commit
- Feature work
  - Lead time for changes
  - Deployment frequency
  - Change failure rate
  - Mean time to recovery
- Engineering NPS

DORA Metrics are Good

But sometimes you want to go a bit deeper

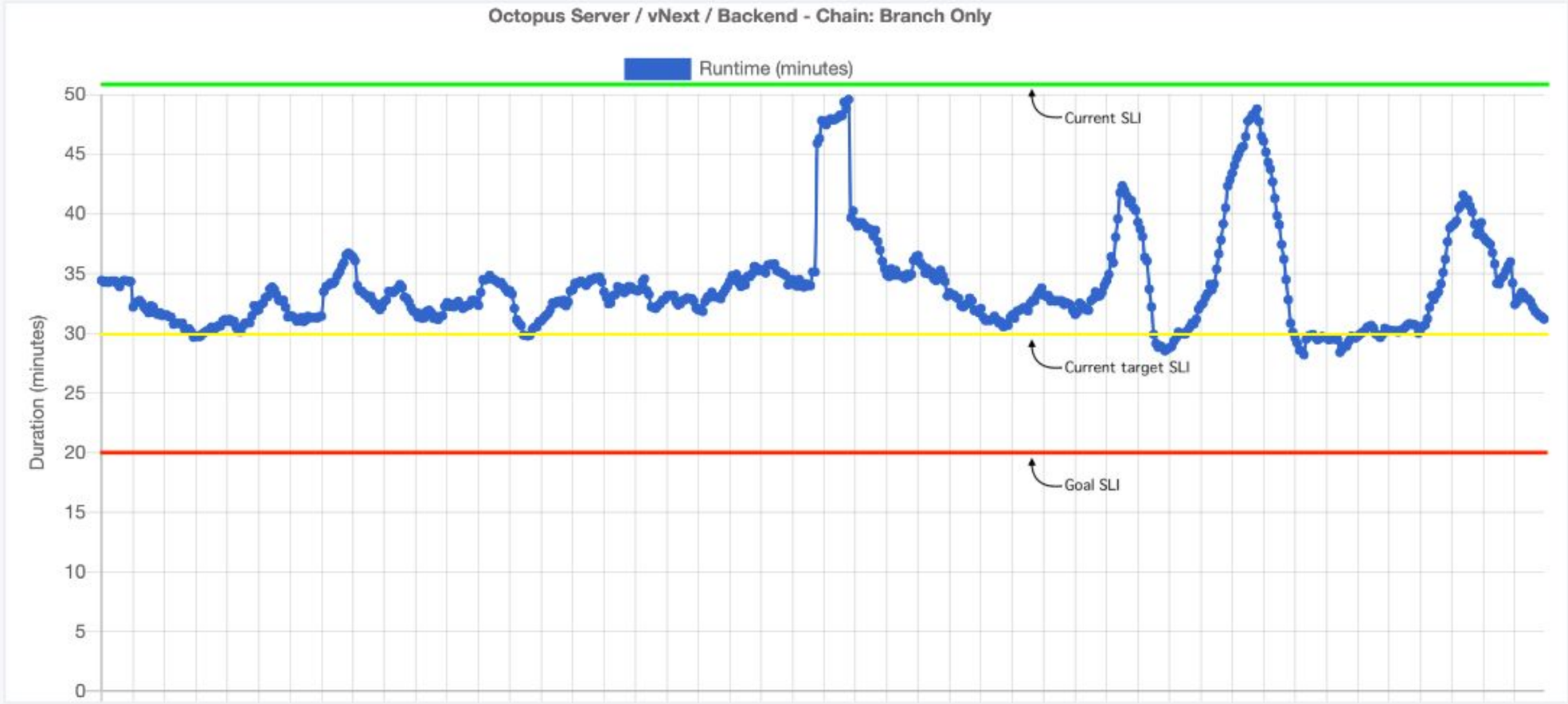
Measuring Success of feature work at Octopus

- PR build duration
- Lead time to changes
- Batch Sizes
- Mean time to recovery (fix a defect)
- High severity defect escape rate (commits causing high severity issues)

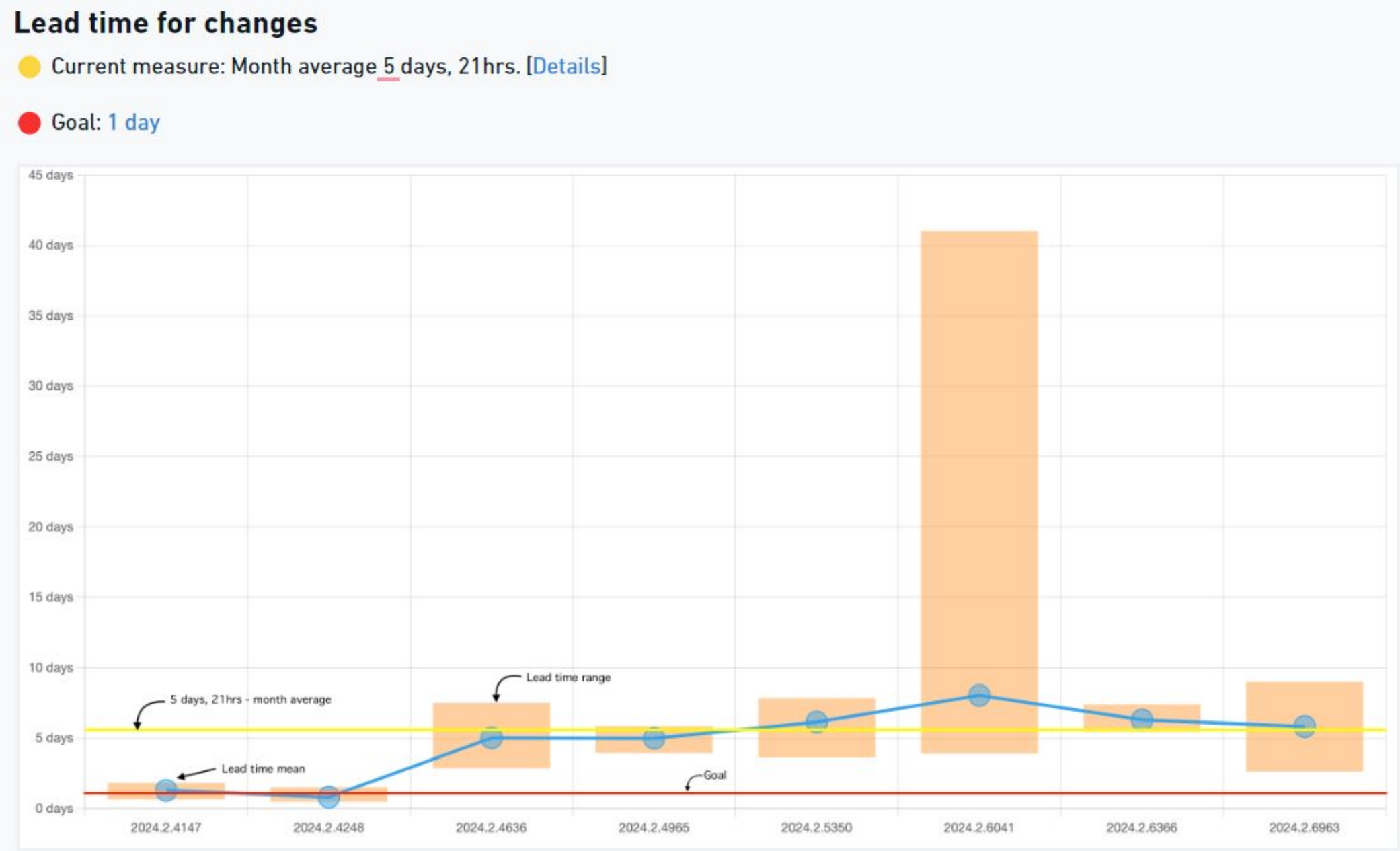
PR Build Duration

PR Build Duration

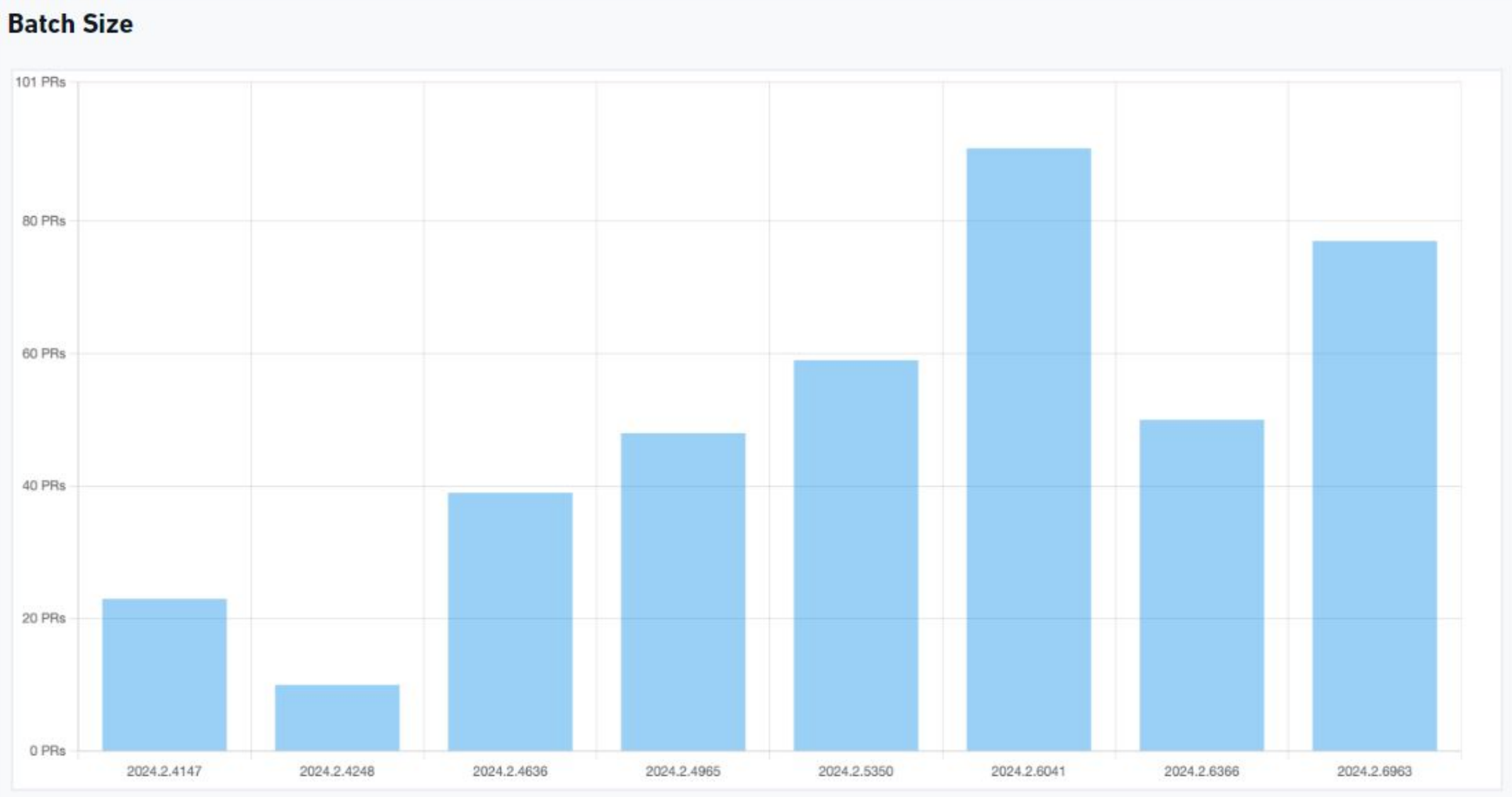
- Current SLI: < 75 minutes [SLI details]
- Current target SLI: 30 minutes [The 30 Minute Challenge]
- Long term goal: < 20 minutes



Lead Time to Changes



Batch Sizes


Batch Size
101 PRs
80 PRs
60 PRs
40 PRs
20 PRs
0 PRs
2024.2.4147
2024.2.4248
2024.2.4636
2024.2.4965
2024.2.5350
2024.2.6041
2024.2.6366
2024.2.6963

MTTR

Mean time to recovery (MTTR)

- Current measure [Details]:
  - 4 days average to detect
  - Less than 1 days to fix, with outliers being over weekend or holiday period
- Goal: < 1 hour to fix

Source

Ignore the `-5 days` for now, looks to be a bug in the graphing code.

`#6542` had a long time to detect as the bug was related to data in customers database that wasn't expected and when changes as part of the EF migration.

Platform Engineers are responsible for gathering and visualizing this data

Resources

Resources

- https://DeveloperExperience.io
- https://dora.dev
- https://octopus.com/devops/platform-engineering/
- https://octopus.com/blog/paved-versus-golden-paths-platform-engineering
- https://octopus.com/blog/devops-platform-engineering
- https://learn.microsoft.com/en-us/platform-engineering/what-is-platform-engineering
- https://microsoft.github.io/code-with-engineering-playbook/developer-experience/
- https://github.blog/2023-06-08-developer-experience-what-is-it-and-why-should-you-care/

@bobjwalker
bob.walker@octopus.com
@bobjwalker

Octopus Deploy

Thank you!

Bob Walker - Field CTO at Octopus Deploy

@bobjwalker

bob.walker@octopus.com

@bobjwalker