Feature Flags

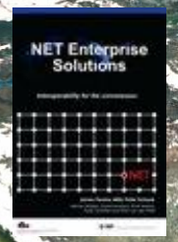
Experiment, release on demand, and limit your blast radius by progressive exposure

WorkSafeBC

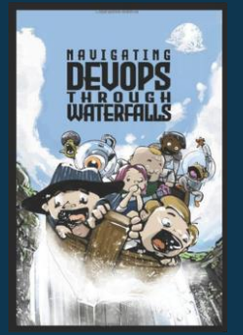
I work here

I was **born** here

I grew up here



Best Practices for the Connoisseur, 2004
Interoperability for the Connoisseur, 2006
Software Engineers on their way to Pluto, 2007
Managing Agile OSS Projects with MS VSO, 2015
Navigating DevOps through Waterfalls, 2020



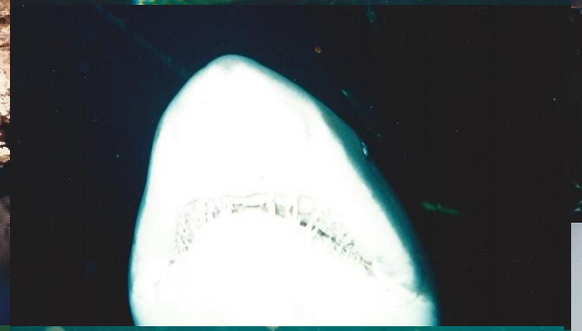
Willy Schaub

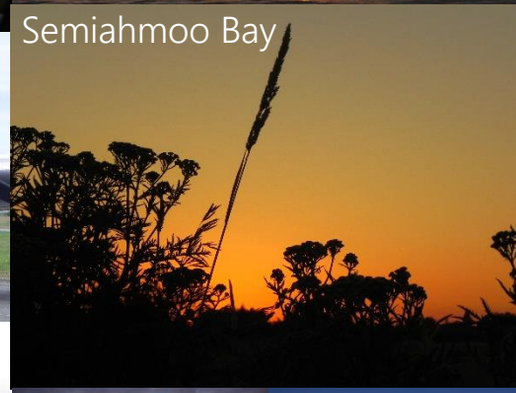
Cloud Systems Engineer, Cloud CeS


Aliwal Shoal

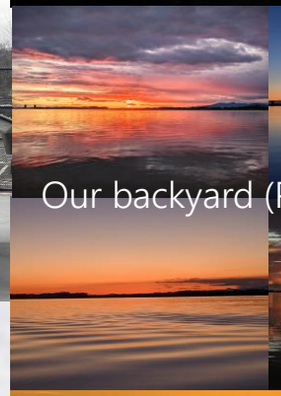

Truk Lagoon


Wondergat

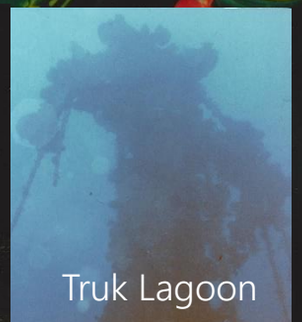


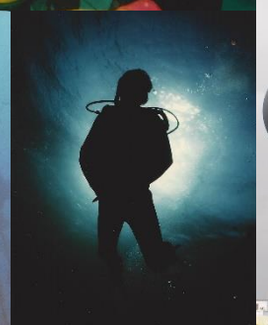

Semiahmoo Bay


DREAMLIFTER




Our backyard (Port Guichon)


Truk Lagoon




Hawaii

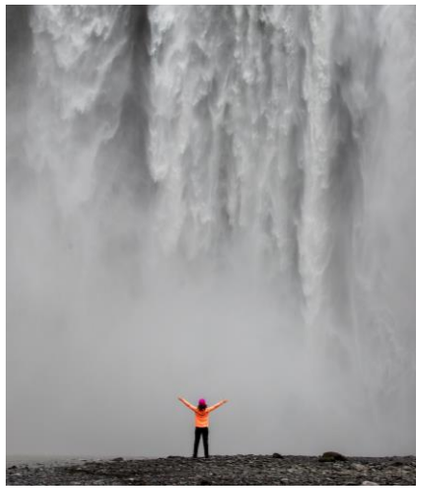
This session is based on willy's views, **not** the **feature flag working group** or **common engineering system**.

COUNTRY FLAGS

WHY?

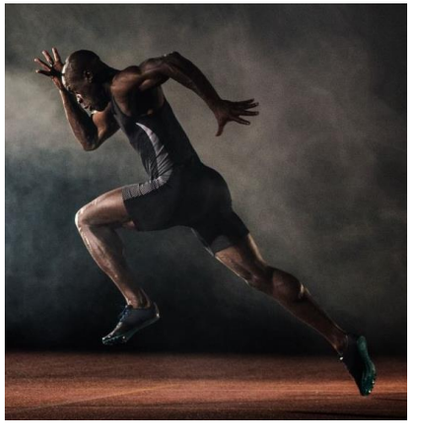
DevOps is the union of people, process, and products to enable continuous delivery of value to our customers.

Donovan Brown,
donovanbrown.com/post/what-is-devops

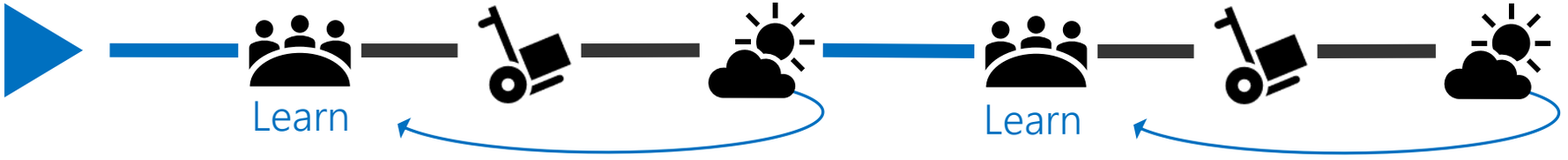


BUILD UAT Experiment Release Customer Feedback

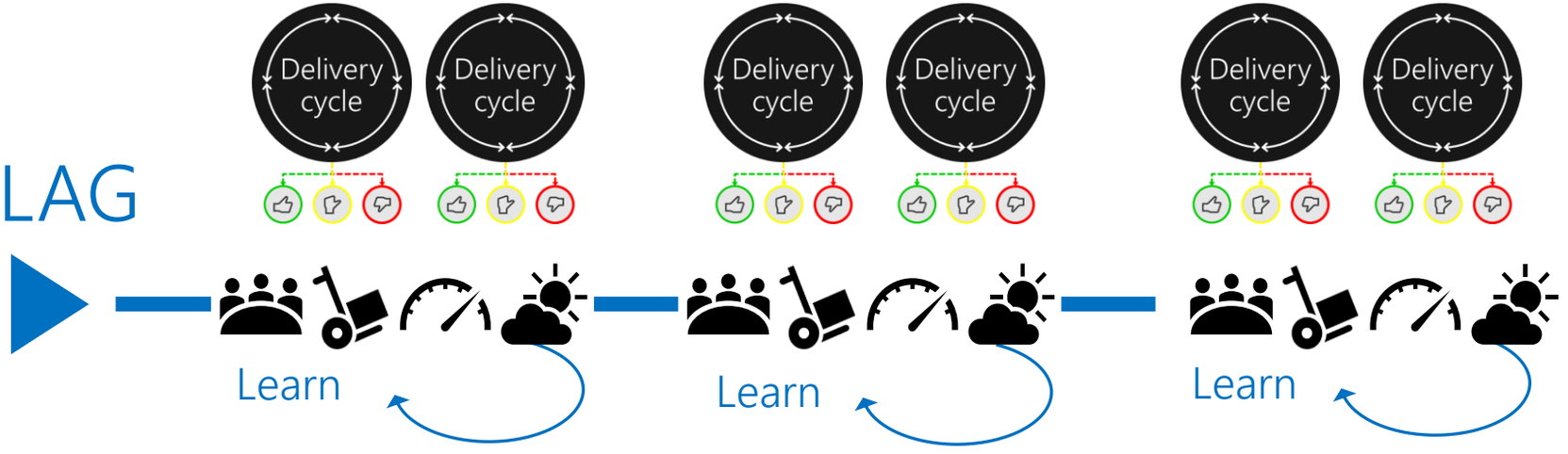
WATERFALL (predictable, process driven)



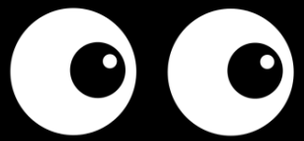
AGILE (create and respond to change)



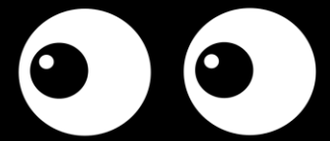
FEATURE FLAG
(experiment)



A way to decouple feature **deployment** from **release**, and fine-tune feature **exposure**, down to the individual user

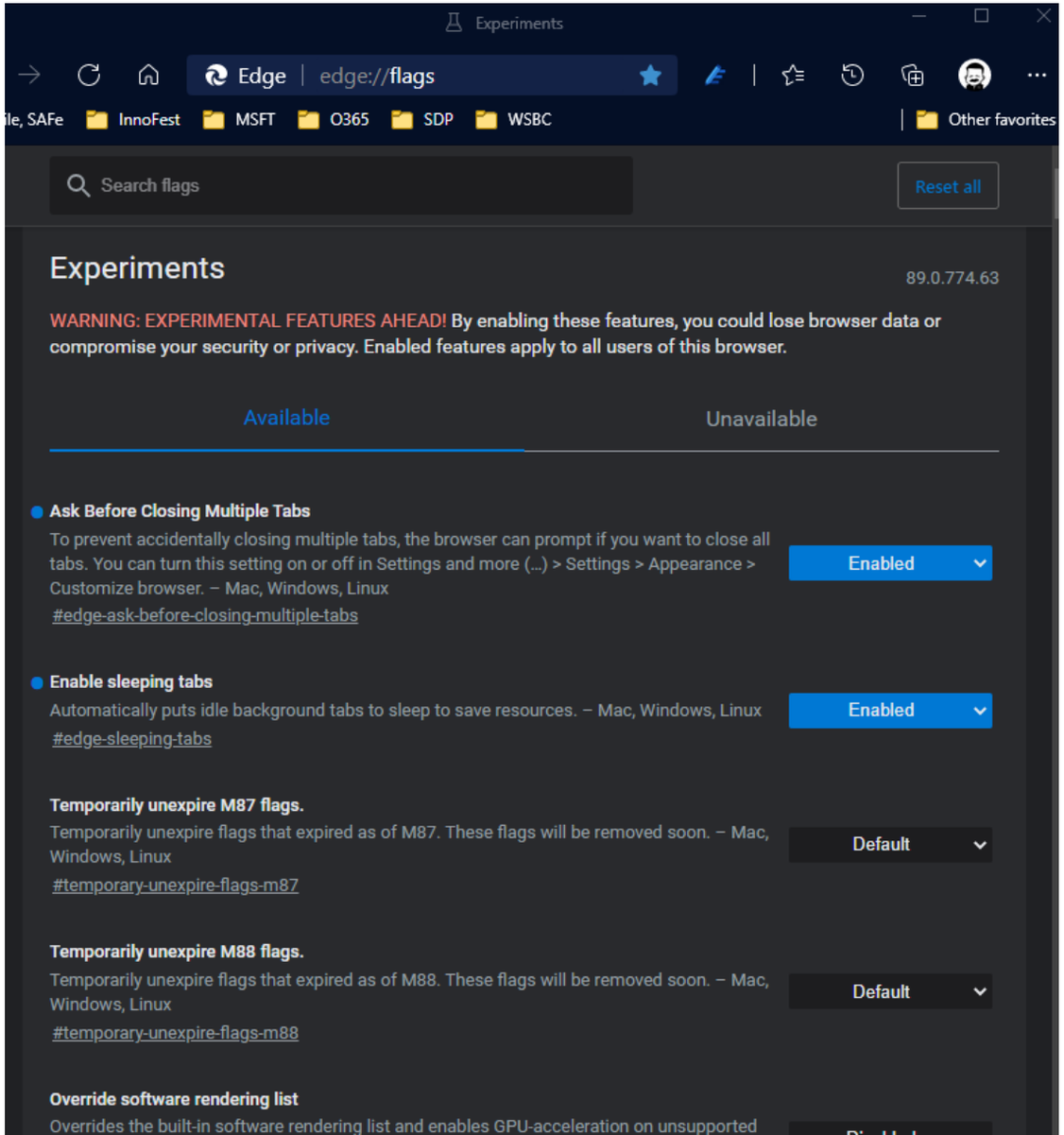


DARK LAUNCH



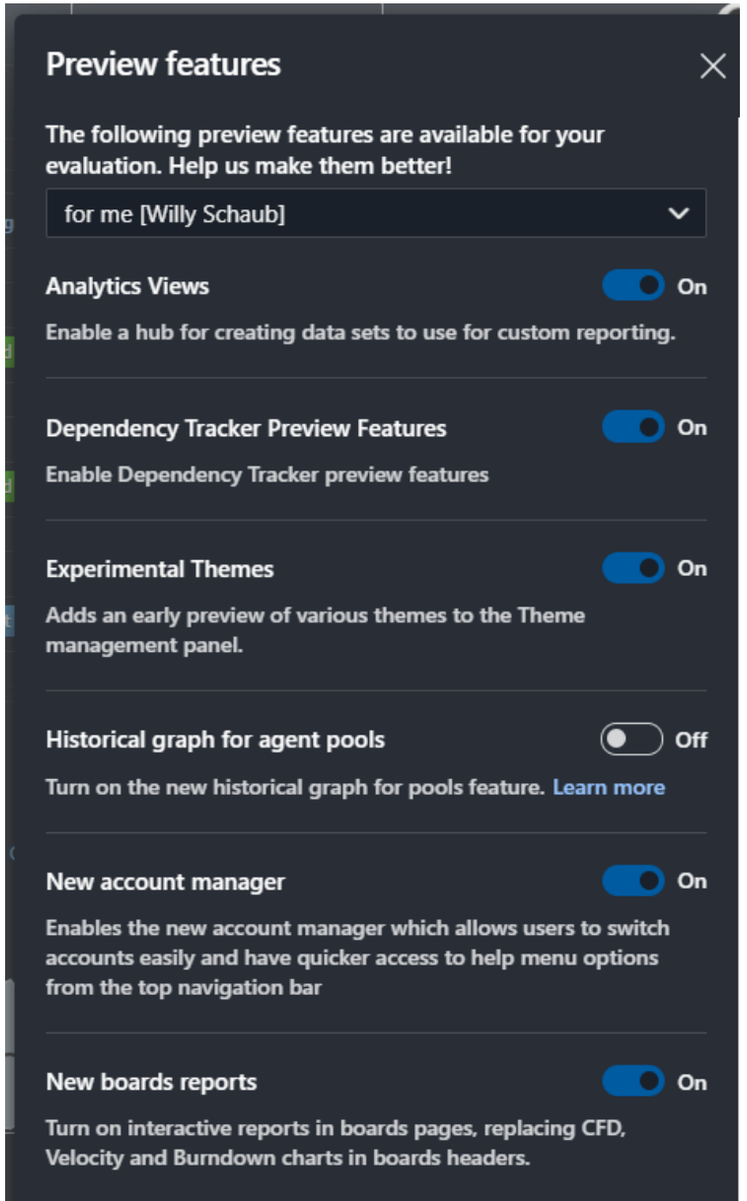

OFF

CLIENT-SIDE PREVIEW EXAMPLES



Azure DevOps ►

◄ Edge

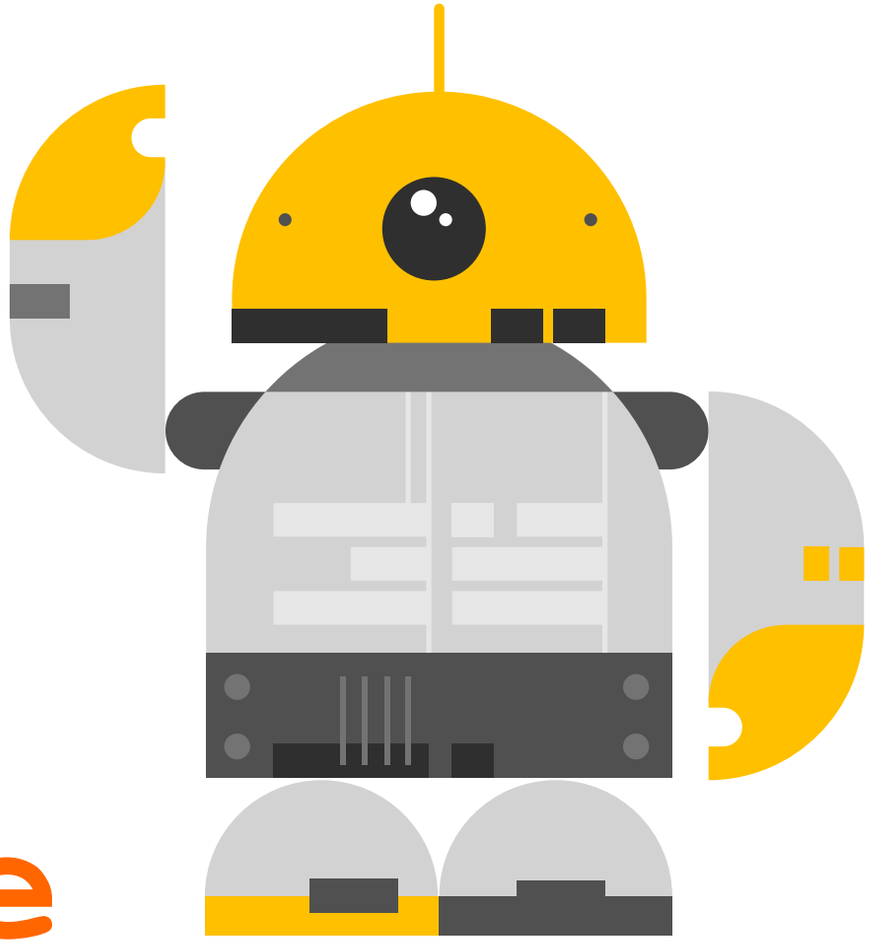


WHAT?

PRODUCT OWNER

```
if
else
```

DEVELOPERS



CODE EXAMPLE

```
using LaunchDarkly.Client;
LdClient ldClient = new LdClient("YOUR_SDK_KEY");
User user = User.WithKey(username);
…
bool featureOn = ldClient.BoolVariation("your.feature.key",
user, false);
if ( featureOn ) {
    // application code to show the feature
}
else {
    // the code to run if the feature is off
}
```

HOW?

Don’t just flip the feature flag

GOVERNANCE

- Usage
- Ownership
- Maintenance

FEATURE FLAGS

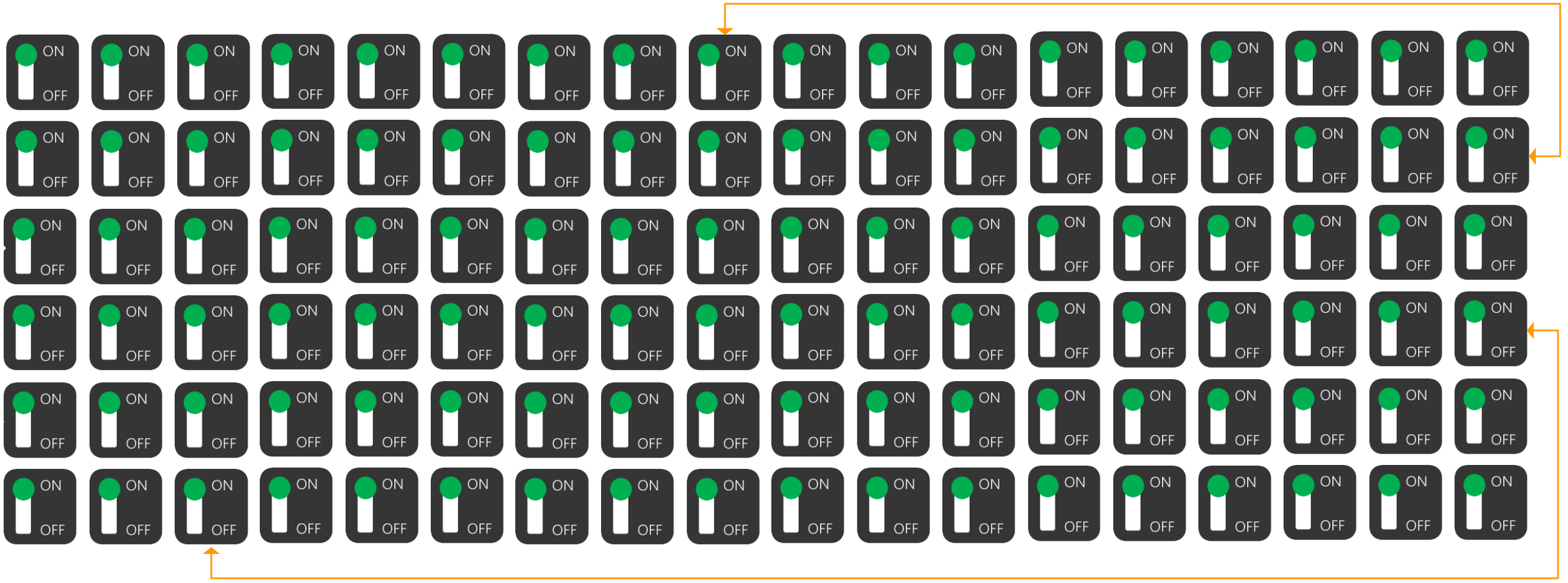
Feature Flag

```
if (flag ON)
else
```

code path #1 → test path #1

code path #2 → test path #2

- Additional code and test paths, and flag maintenance
- Dependencies
- Implications



Understand the impact of flipping a flag!

Server & Tools Blogs > Developer Tools Blogs > Brian Harrys blog

Executive Bloggers | Visual Studio Products | DevOps | Languages | Features | .NET

Brian Harrys blog

Everything you want to know about Visual Studio ALM and Farming

A Rough Patch

11/25/2013 by Brian Harry MS // 10 Comments

Share 0 | 2 | 0

Either I'm going to get increasingly good at apologizing to fewer and fewer p

We've had some issues with the service over the past week and a half. I feel
cident we've had since the instability created by our service refactoring in the
can assure you that we have taken the issue very seriously and there are a fai
recently.

The incident started the morning of the Visual Studio 2013 launch when we i
we made. You may not have noticed it by my presentation but for the couple

ALM | DevOps Rangers

Providing practical guidance, experience, and gap-filling solutions to the developer community.

How we checked and fixed the 503 error and Performance issue in our Azure Function

mikaelkrief April 3, 2018

Share 4 | 40 | 0 | 1

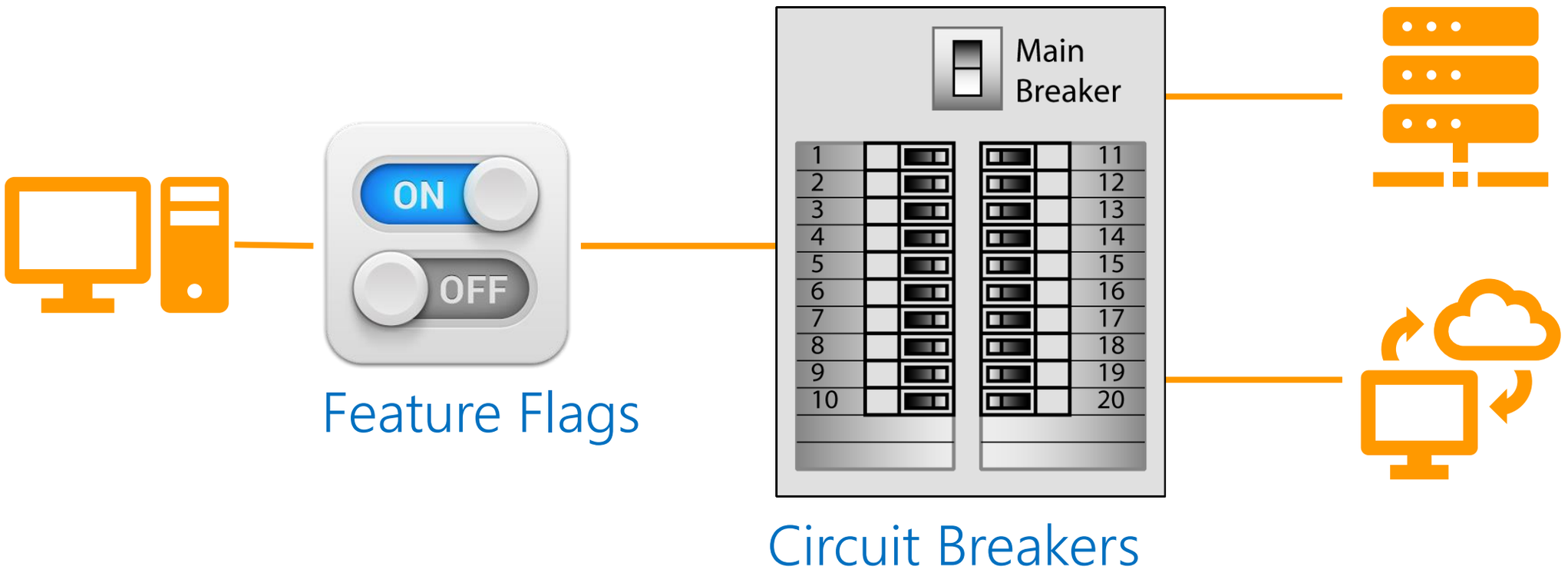
How we checked and fixed the 503 error and Performance issue in our Azure Function

Since we released our Azure Function that allows you to use LaunchDarkly services with our VSTS Roll Up Board widget extension, we have recently evolved it further by adding an Azure Function that allows you to retrieve the flags of the user connected to Visual Studio Team Services (VSTS).

The goal of this evolution was to centralize all the business software of the features flags in Azure Function and thus leave it free of the code of the VSTS extension. This allows it to plug other extensions with the features flags in a much more simplified way thus requiring the least possible development.

Shortly after release, a user reported an issue loading the extension.

Feature Flag != Circuit Breaker



Solutions?

.NET

Feature Toggle , NFeature , FeatureSwitcher

Java

Togglz , FF4J

JavaScript front-end frameworks

JavaScript Feature Flags , Angular Feature Flags , Ember Feature Flags , React feature flags

SaaS solution for all stacks

FeatureOps, LaunchDarkly

All feature flags must support the kill switch

enable us to toggle a feature in case of an emergency

FFs managed by Business, Security, & QA

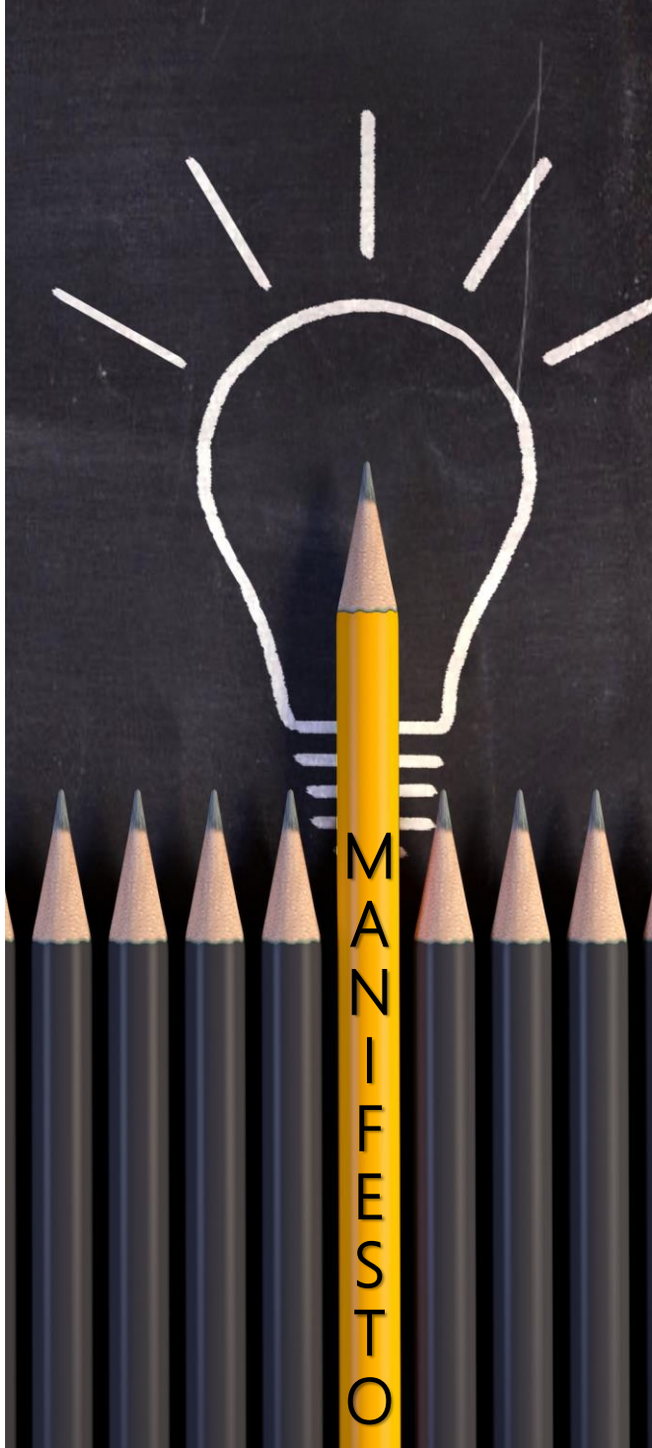
ensure we satisfy governance and customers with continuous delivery of value

FFs protect only production ready code

reduce risk of deploying accidental release of fragile or insecure code to customers

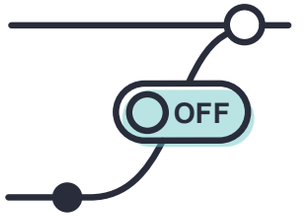
Favour software as a service over custom code

avoid long-term ownership, support, and maintenance of feature flag frameworks



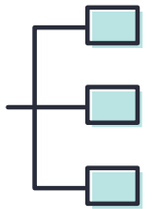
Ship Code When You're Ready

Test in Production



In Case of Emergency, Kill Switch

Control Your User's Experience



Experiment & Get Feedback Easier

If you need to **rebuild** or **redeploy** your solution you are not using feature flags

Feature flags are not a solution to hide **non-production** ready code

Knowing IF and WHAT A|B test experiments to use

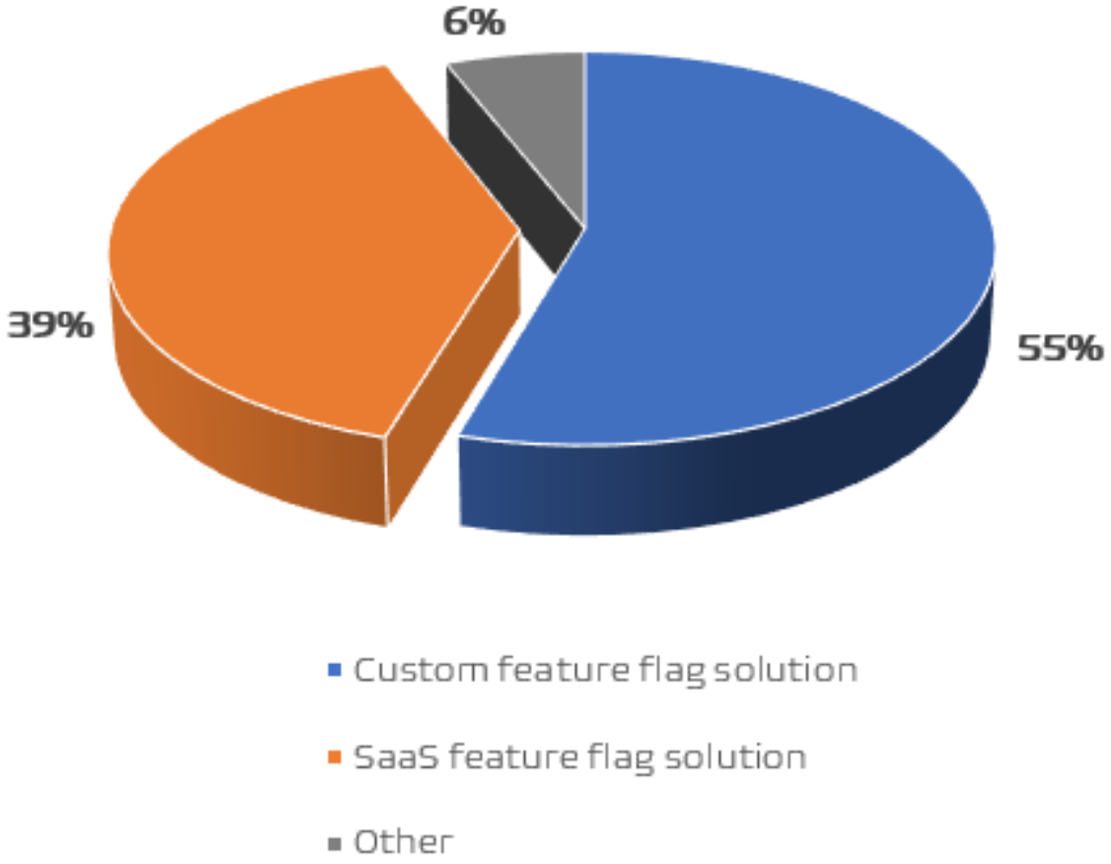
WHY
do you need feature flags?

x5

WHO
owns and grooms your (custom) F/F service?

COMMUNITY POLL

Have you or are you using feature flags (FF) to enable release on demand? If yes, what strategy are you using?



Raw, unedited feedback:

- *I've only used short-lived feature flags so there was never any need for a beefier solution. A simple hand-cranked conditional logic almost always suffices for short-lived toggles.*
- *Love using Launch Darkly! Dark launching is the way forward for so many teams! Next week Launch Darkly's Galaxy conference is gonna be awesome!*
- *Launch Darkly all day.*
- *We've been using rollout, but also switched to AAC, recently for a few apps. From config management perspective, it's been a dream for production support issues.*

Q&A

You have

Questions

we (hopefully) have

Answers

REFERENCES

- A Rough Patch | Brian Harry's Blog (microsoft.com)
- Deploying new releases: Feature flags or rings? | Opensource.com
- How we checked and fixed the 503 error and Performance issue in our Azure Function | Microsoft Docs
- 3 ways to handle transient faults for DevOps | Opensource.com
- Use Feature Flagging with LaunchDarkly SaaS | CeS
- Why hypothesis-driven development is key to DevOps | Opensource.com
- What's the cost of feature flags? | Opensource.com
- Summary of my publications | LInkedIn

Thank you!

Willy[-Peter] Schaub

Cloud Systems Engineer

willy.schaub@worksafebc.com

@wpschaub

WorkSafeBC