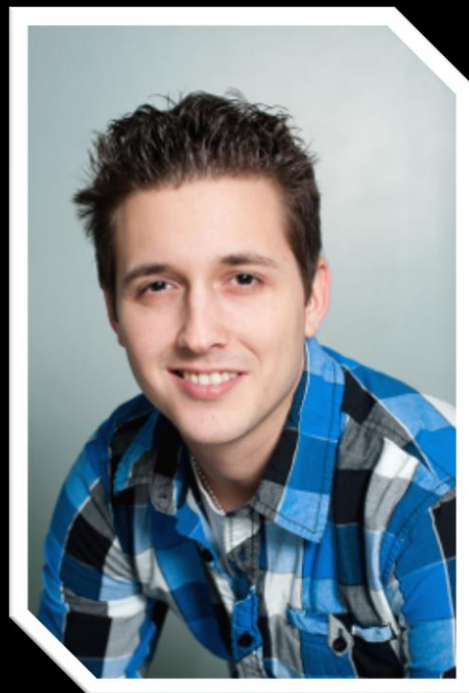
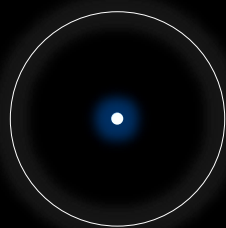




SPS COMMERCE

UNLEASHING DEPLOY VELOCITY WITH FEATURE FLAGS

A DEVOPS JOURNEY



TRAVIS GOSSELIN

PRINCIPAL SOFTWARE ENGINEER

DEVELOPER EXPERIENCE 

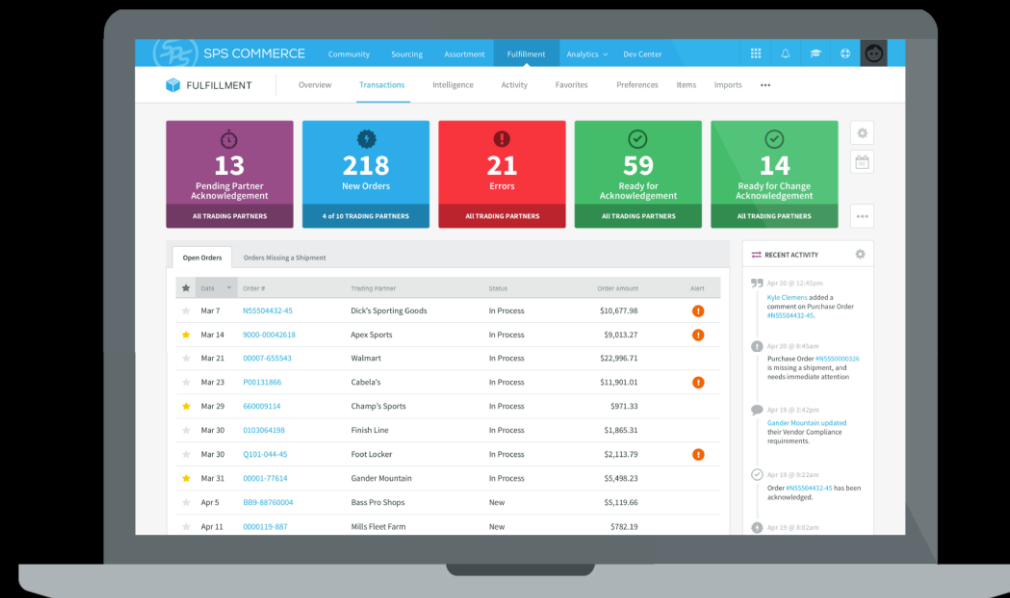
@TRAVISJGOSSELIN 

www.travisgosselein.com 



SPS COMMERCE

INFINITE RETAIL POWER™



//

Developer Experience is the activity of studying, improving and optimizing how developers get their work done.

theapplslab.com (2017)

//

DEVELOPER EXPERIENCE

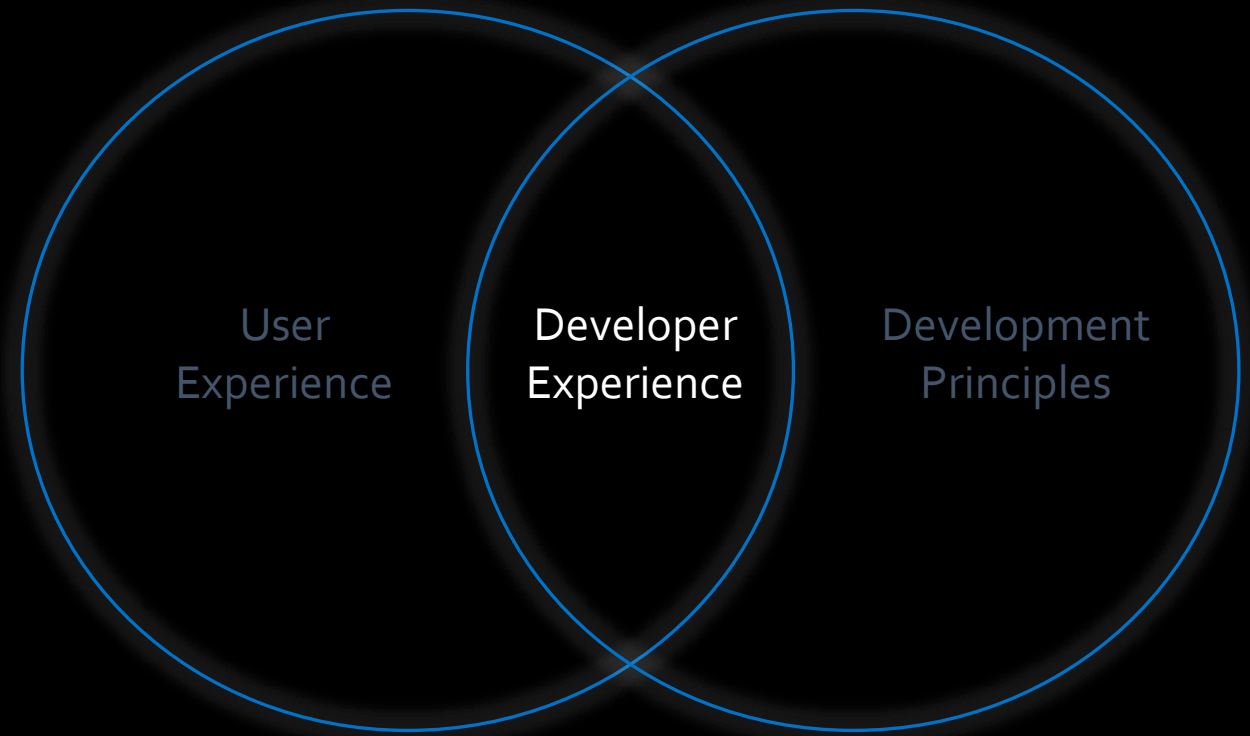
WHAT IS THAT...EXACTLY?

//

Developers work in rainforests, not planned gardens.

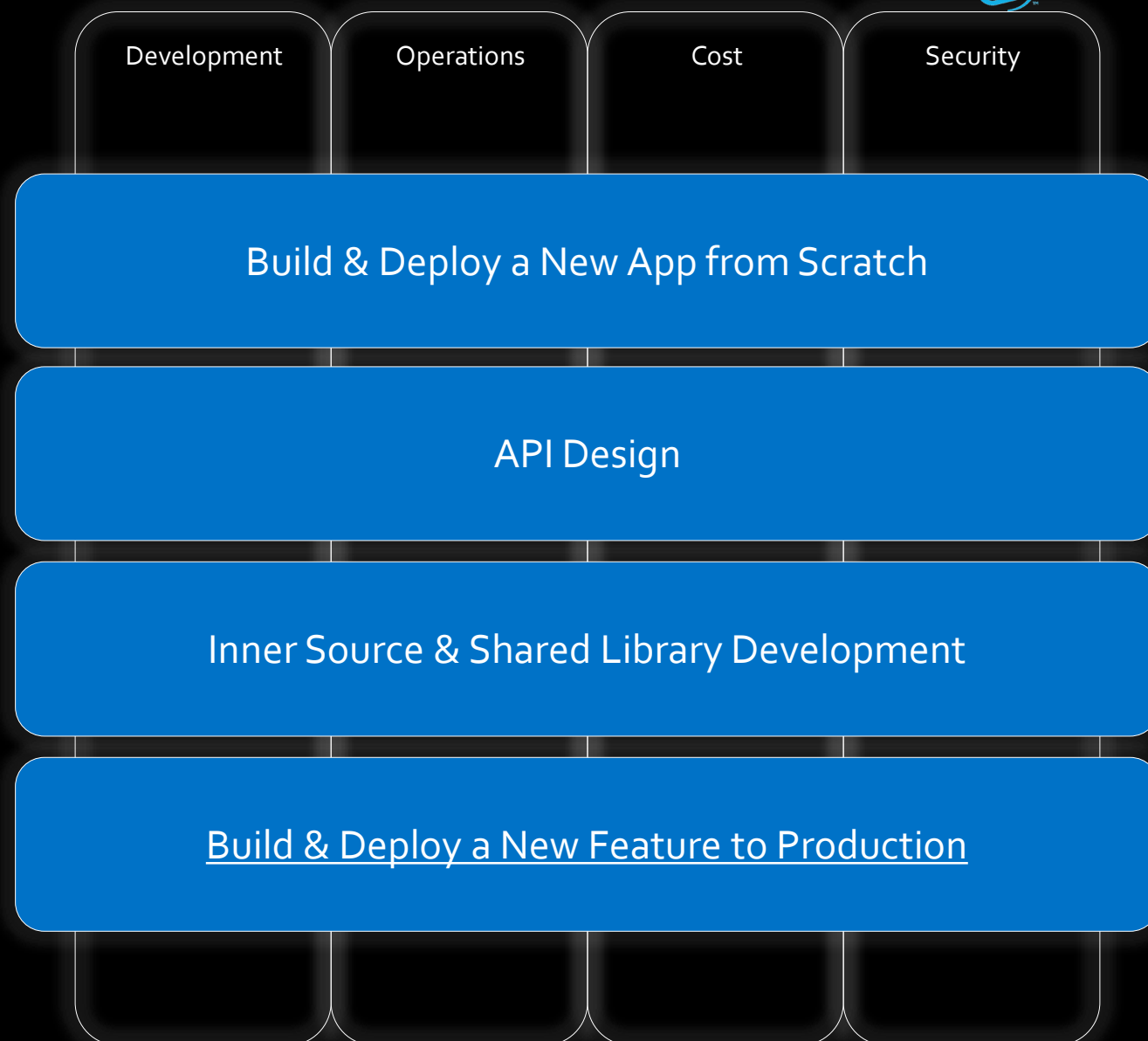
a16z.com

//

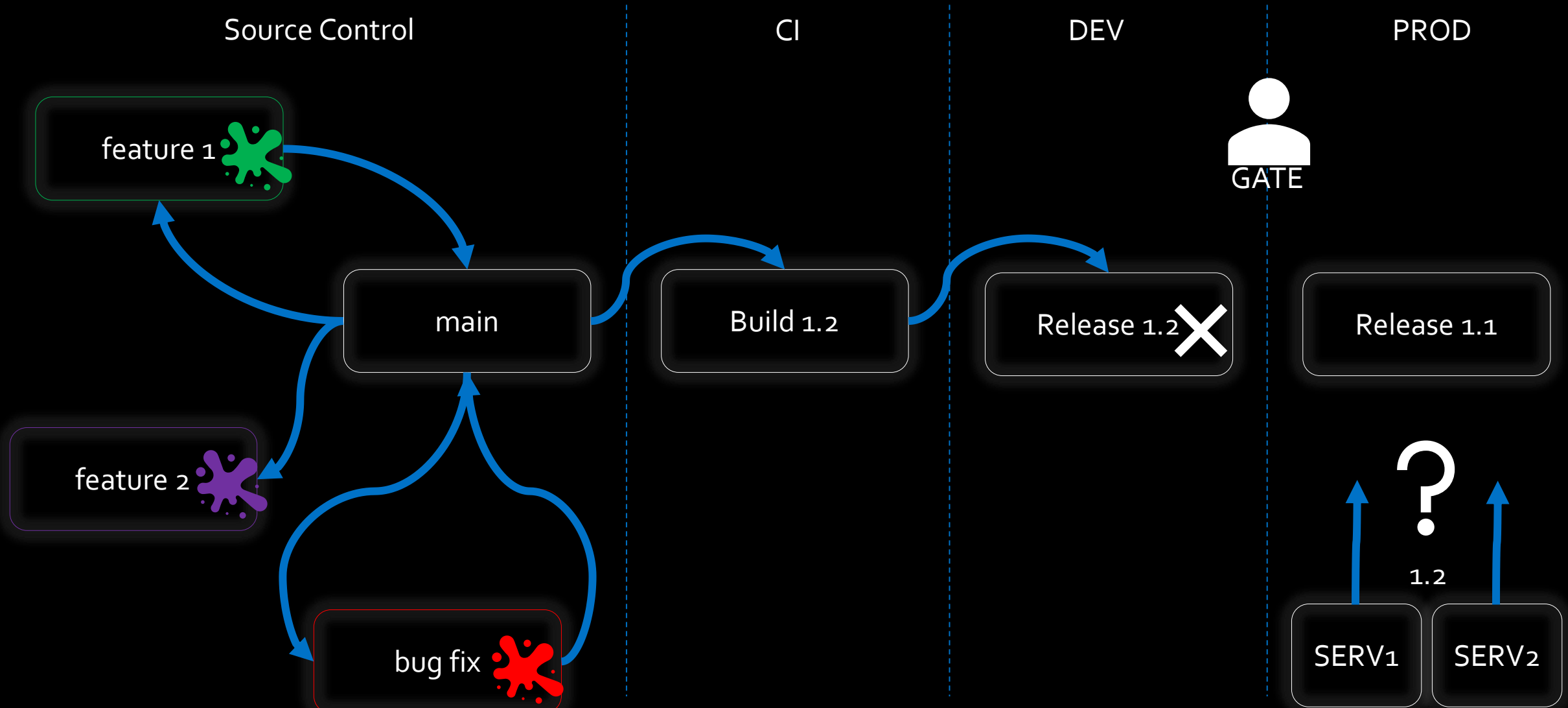


DEVELOPER EXPERIENCE: CAPABILITIES

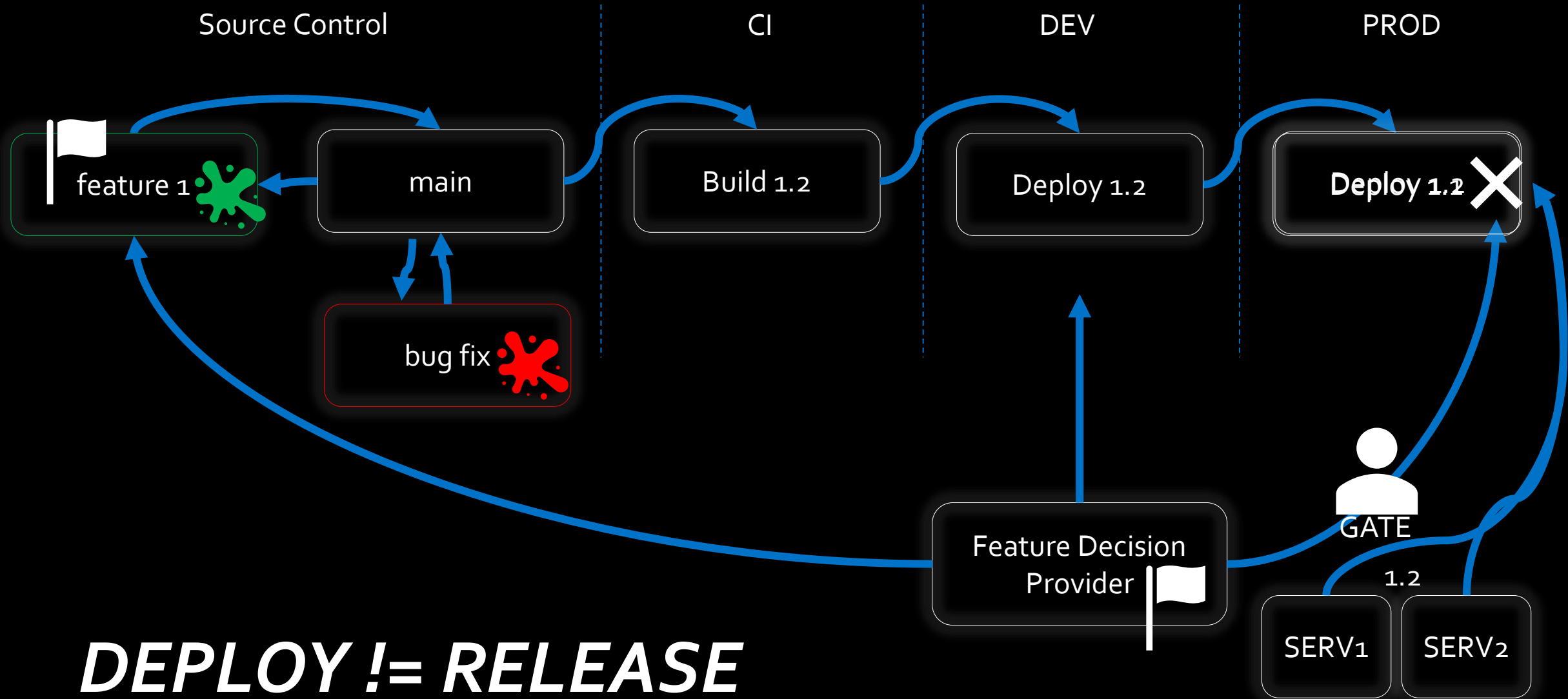
IDENTIFIED HORIZONTAL FAST TRACKS TO
BE CURATED FOR MAXIMUM EFFICIENCY.



The Problem



The Solution



*"...a powerful technique,
allowing teams to modify
system behavior without
changing code"
(MartinFowler.com)*

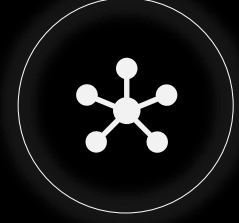


FEATURE FLAGS

AKA "CONFIG FLAGS", "FEATURE TOGGLES", "FLIPPERS"



Release



Operational



Experiment



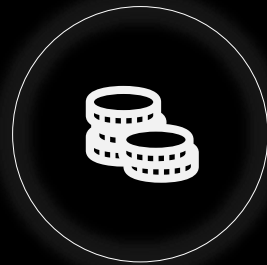
Permission

Flexibility with Feature Flags



Branching Strategy

Shorter, Multiple
Active Releases



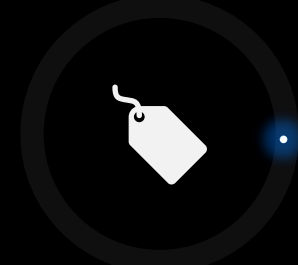
True CI

Continuous
Integration the way it
should be!



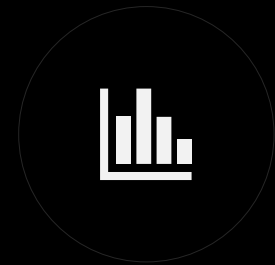
Ship Faster

Confidently Deploy
Rollback Features



Testing in PROD

A/B Testing
Canary Releases
Less Environments



Culture

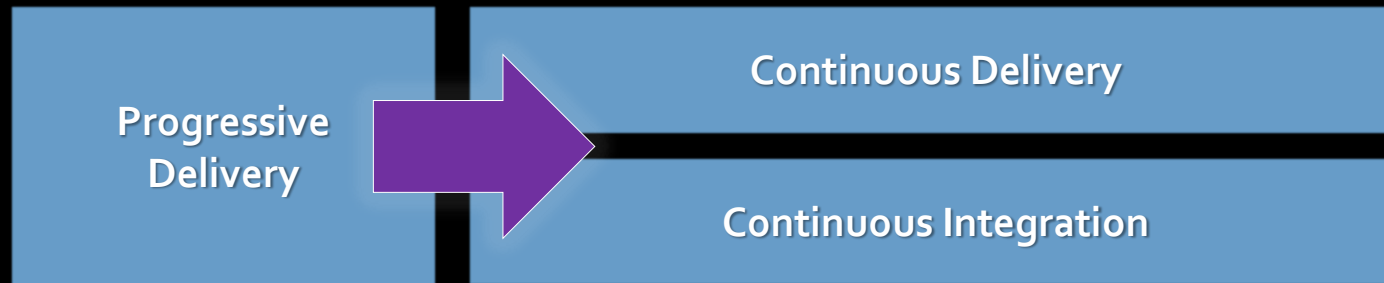
Product Owners
Manage Features

Progressive Delivery

//

Progressive Delivery is a modern software development lifecycle that builds upon the core tenets of Continuous Integration and Continuous Delivery (CI/CD). Organizations that employ Progressive Delivery ship code faster, reduce risk, and continuously improve the customer experience.

Launchdarkly.com



SOFTWARE DELIVERY AND OPERATIONAL PERFORMANCE

Accelerate State of DevOps 2022



Performance Metrics

Higher Performers



Deployment Frequency

On-demand
(whenever we want)



Lead Time for Changes

One day



Mean Time to Restore

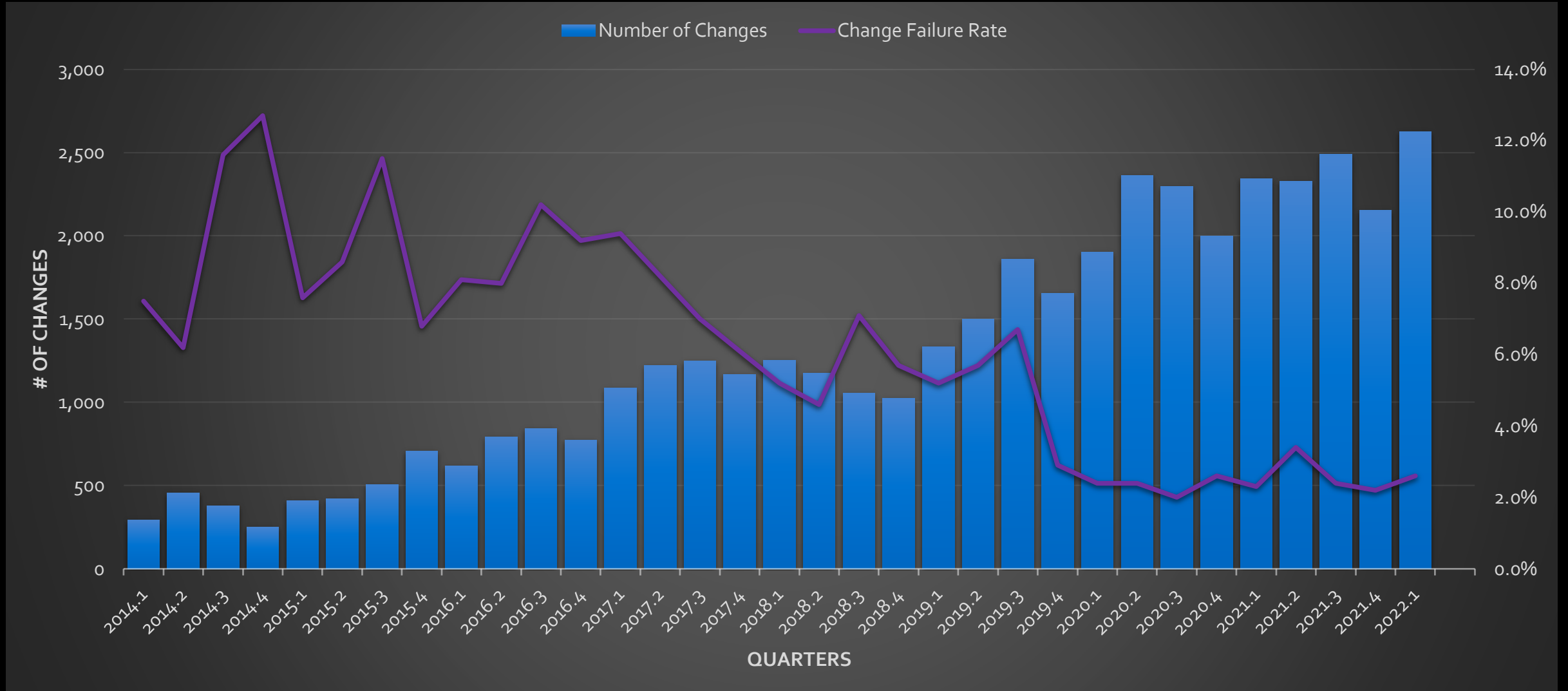
Less than one day



Change Failure Rate

Less than 15%

SPS Commerce Change Performance



Simple Feature Flag

```
public void SetupUser(UserContext user)
{
    CreateUser(user);
```

```
    if (IsSendgridEmailEnabled(user)) ← Feature Flag
```

```
    {
        SendSendgridEmail(user); ← *NEW* Code
```

```
    }
```

```
    else
```

```
    {
```

```
        SendLocalSmtplibEmail(user); ← *OLD* Code
```

```
    }
```

```
}
```

```
public bool IsSendgridEmailEnabled(user)
{
    return service
        .UseFeature(user, "Sendgrid");
}
```



IMPORTANT REALIZATIONS

YOUR FEATURE FLAG HONEYMOON IS OVER!



Shifting Left



Maintainability



Changes to
Existing Code



Feature Decision
Provider

Feature Decision Providers

Technology

Contenders

Leaders



LaunchDarkly



CloudBees

High Presence



AB Tasty



Not Satisfied

Very Satisfied



Niche

Low Presence

High Performers



Flagsmith



LaunchDarkly & SPS Commerce

LaunchDarkly ➔

DEFAULT PROJECT

Production

Feature flags

Users

Segments

Experiments

Debugger

Audit log

Integrations

Account settings

Feature flags

Use this page to see all feature flags in this project. Select a flag to manage the environment-specific targeting and rollout rules.

Save

26 flags

Find a flag by name, key or description

Filter

Sort: Newest

Create flag

<p>Checkout options Added 16 hours ago</p> <p>checkout-options</p>	<p>(+) Never requested</p> <p>Two-click che...</p>	<p>Off</p>
<p>Markdown Link with Title Added 7 days ago</p> <p>markdown-link-with-title</p>	<p>(-) Never requested</p> <p>false - off vari...</p>	<p>Off</p>
<p>Markdown List Added 7 days ago</p> <p>markdown-list</p> <p>this is a list description for ordered:</p>	<p>(-) Never requested</p> <p>true</p>	<p>On</p>
<p>nina-markdown-test Added 7 days ago</p> <p>nina-markdown-test</p> <p>Nina creating a flag to test markdown</p> <p>Buckinghamshire</p>	<p>(-) Never requested</p> <p>true</p>	<p>On</p>

//

It just makes my life simpler and my deployments more relaxing.

Jake Freeberg, Lead Engineer, SPS Commerce

//

//

LaunchDarkly is a game changer in enabling us to rapidly experiment and receive critical feedback and keep the development pipeline moving while other departments prepare for launch activities.

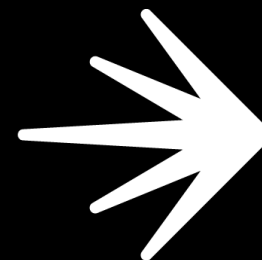
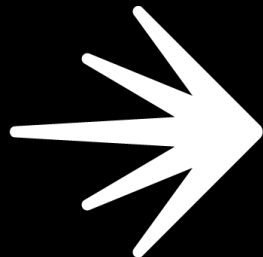
Jerel Thompson, Development Director, SPS Commerce

//

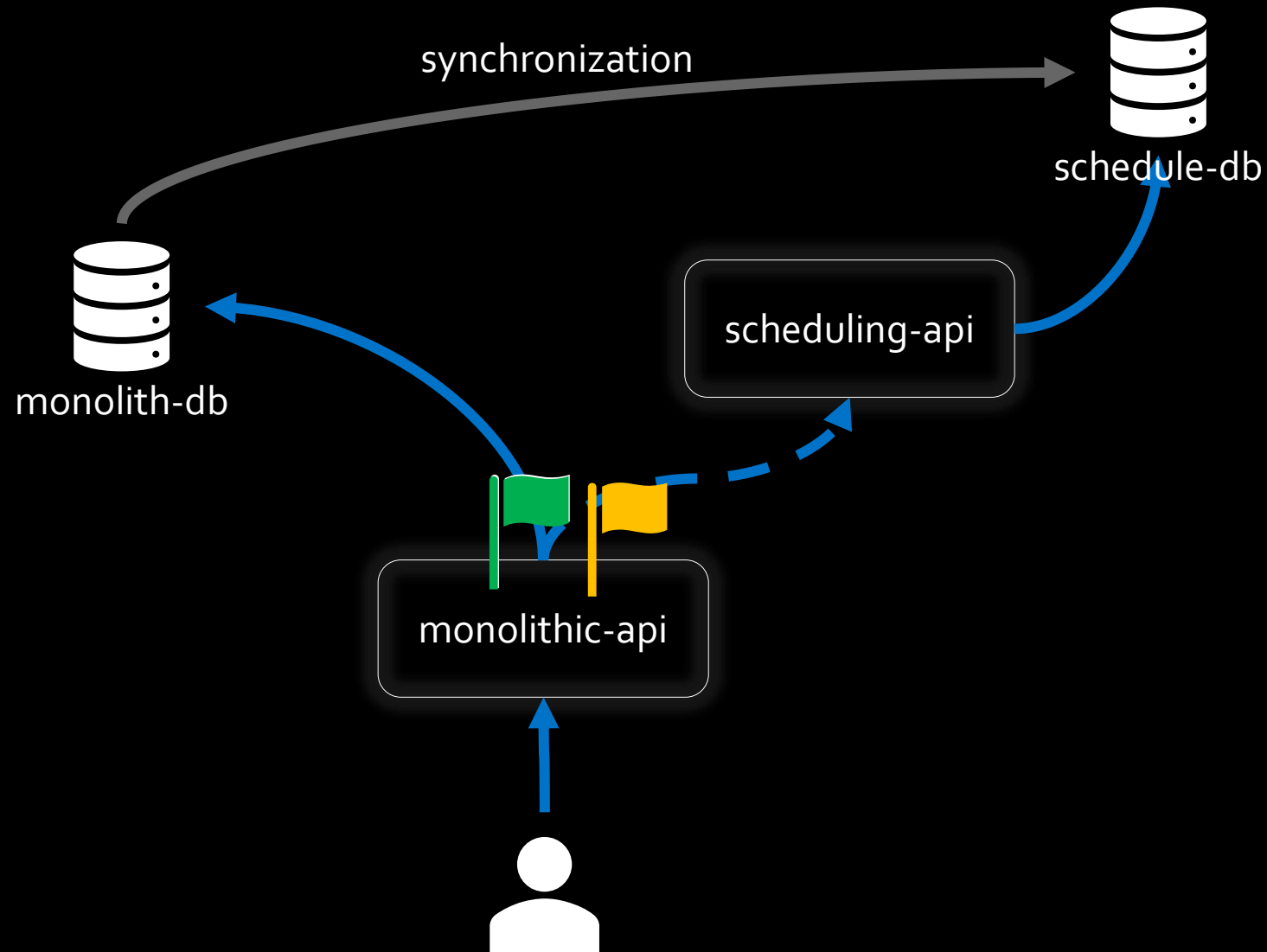


SPS COMMERCE

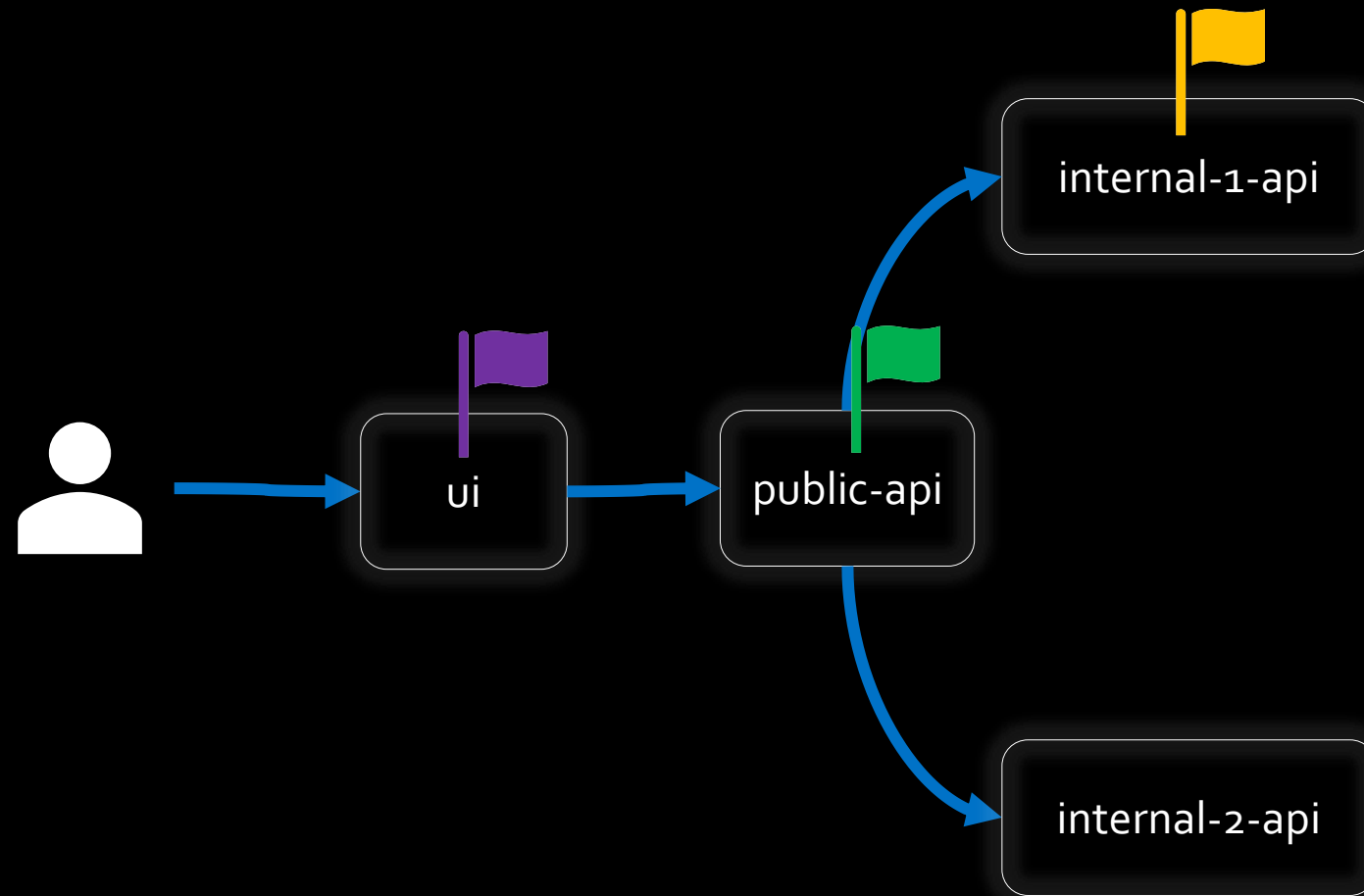
LaunchDarkly & Me!



Migrations



Coordination



User Context

User Identifier (UserID, Username, etc)				
User Details (First Name, Last Name, Email)				
Role	Group	Organization	Internal / Employee	Beta Customer

```
const user = {
  userId: '123456'
  name: 'Travis Gosselin'
};
const client = featureProvider.initialize(user);
```

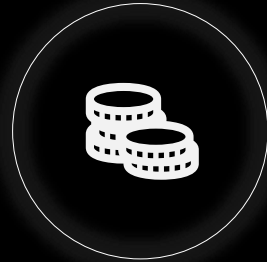


```
const user = {
  userId: '123456'
  firstName: 'Travis',
  lastName: 'Gosselin'
};
const client = featureProvider.initialize(user);
```

Other Scenarios



Log Level
Verbosity



Dynamic
Configuration



Kill Switches

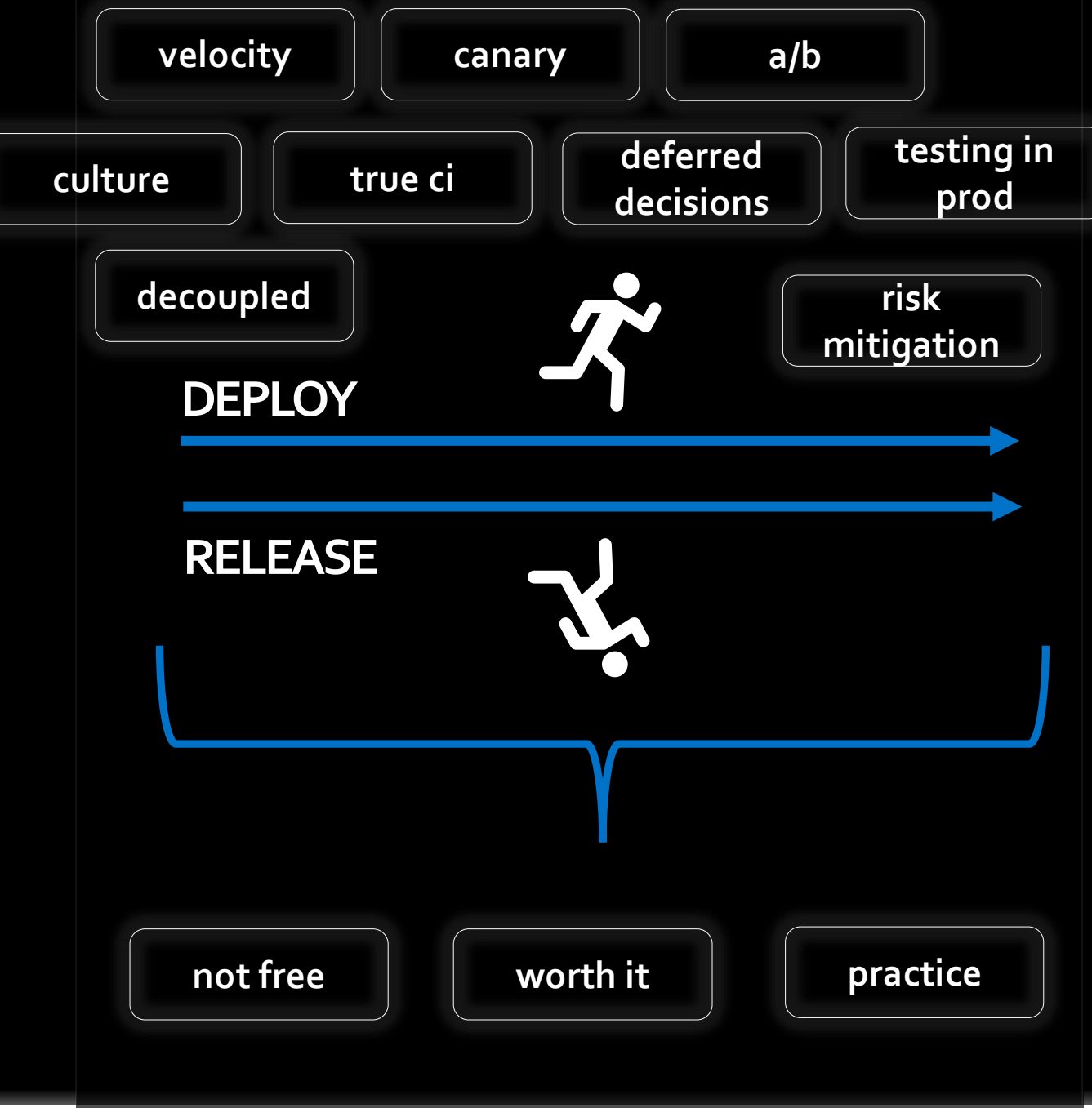


Sunset Features



Timed Features

UNLEASHING DEPLOY VELOCITY WITH FEATURE FLAGS



TRAVIS GOSSELIN 

@travisjgosselin 

www.travisgosselin.com 