



HarnesU Fall Semester

# Who are your teachers today?



Tiffany Jachja  
Evangelist / Harness

.....



Drew Oetzel  
Head of Training / Harness

.....



Ravi Lachhman  
Evangelist / Harness

.....



# What are we talking about today?

- CI and CD, Better Together!
- Kubernetes Deployments Best Practices
- Your first CI and CD Together with Harness

# CI and CD



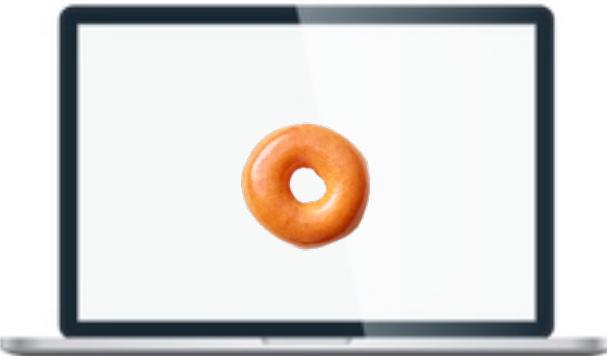
# Dream of the Artifact



=



# Because Eventually



# History of CI

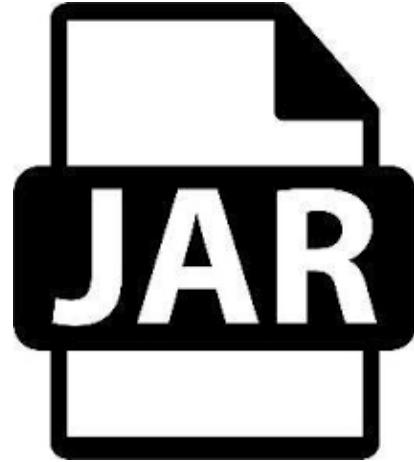
Purpose Built



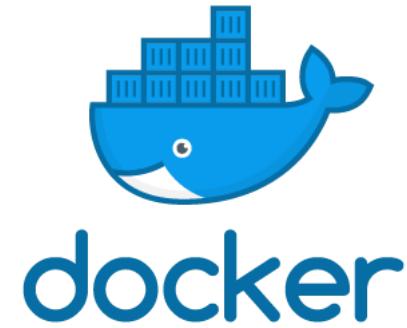
Cloud Native



# Deployable vs Distribution

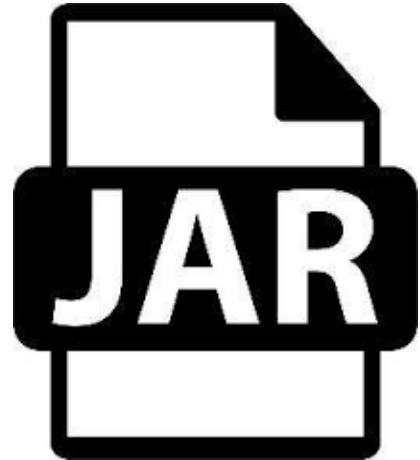


VS

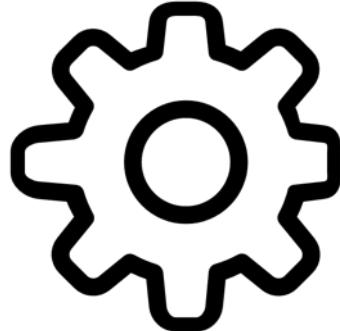


Amazon Machine  
Image (AMI)

# Build vs Release Candidate



+



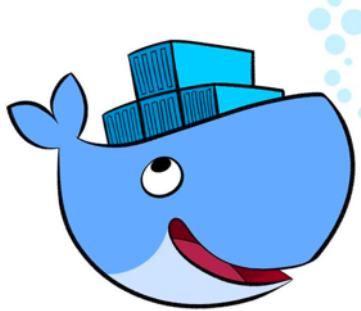
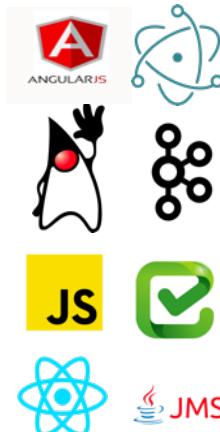
# Back in my day... 2008



# Still my day... 2020



Maven

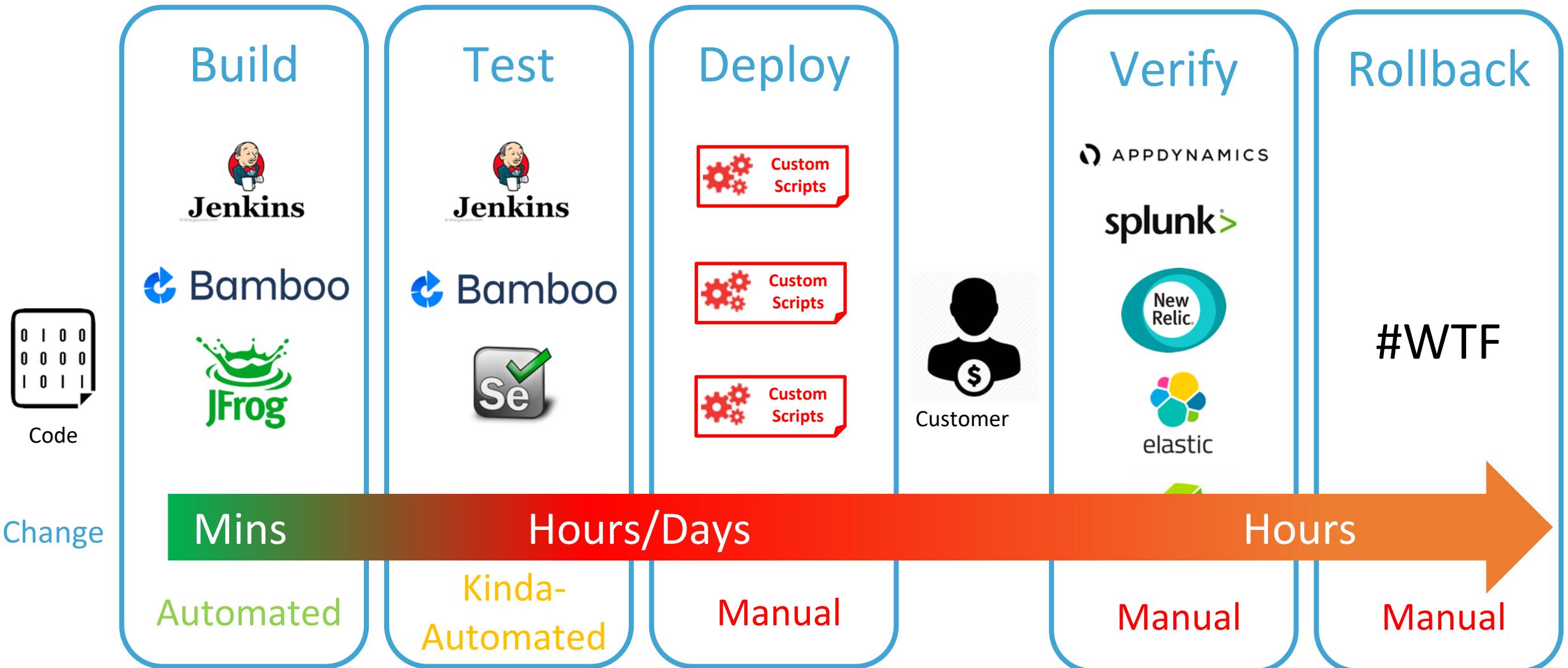


StatsD

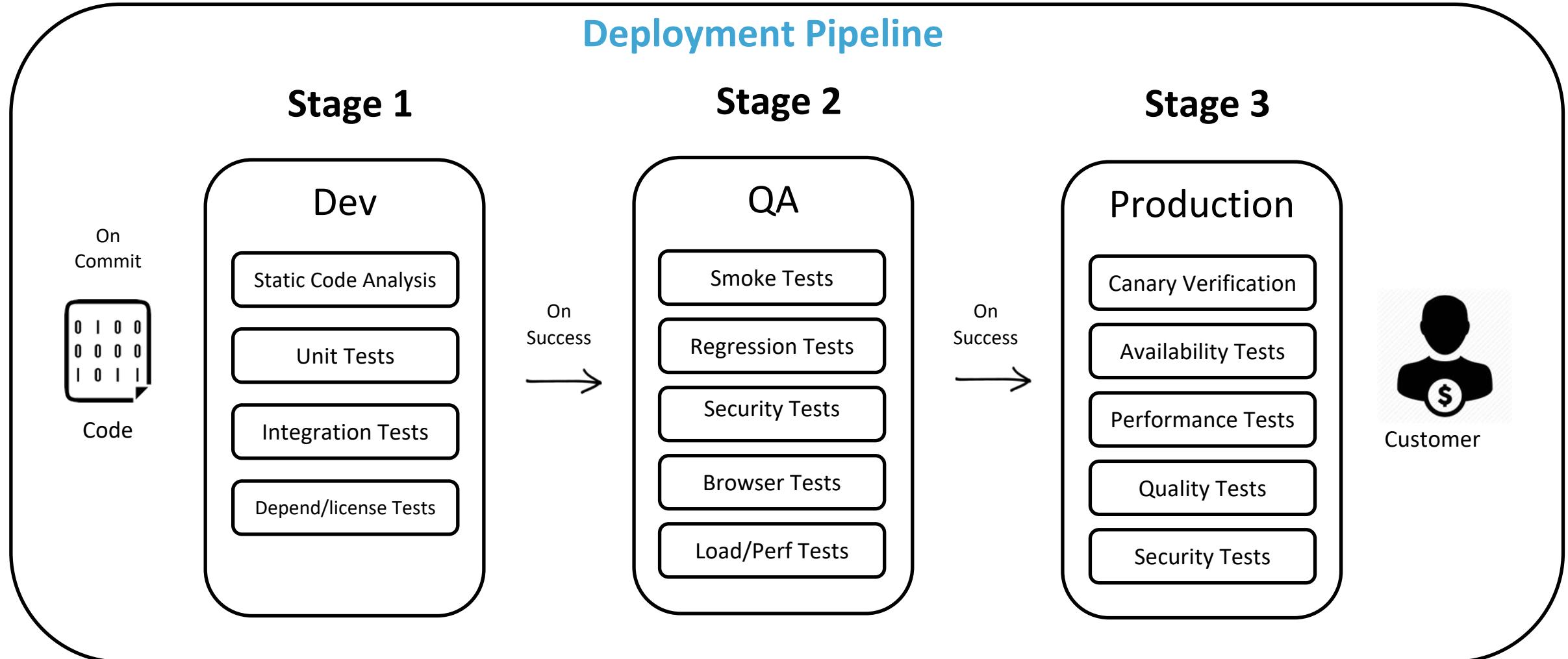
aws



# Software Deployment Pipeline



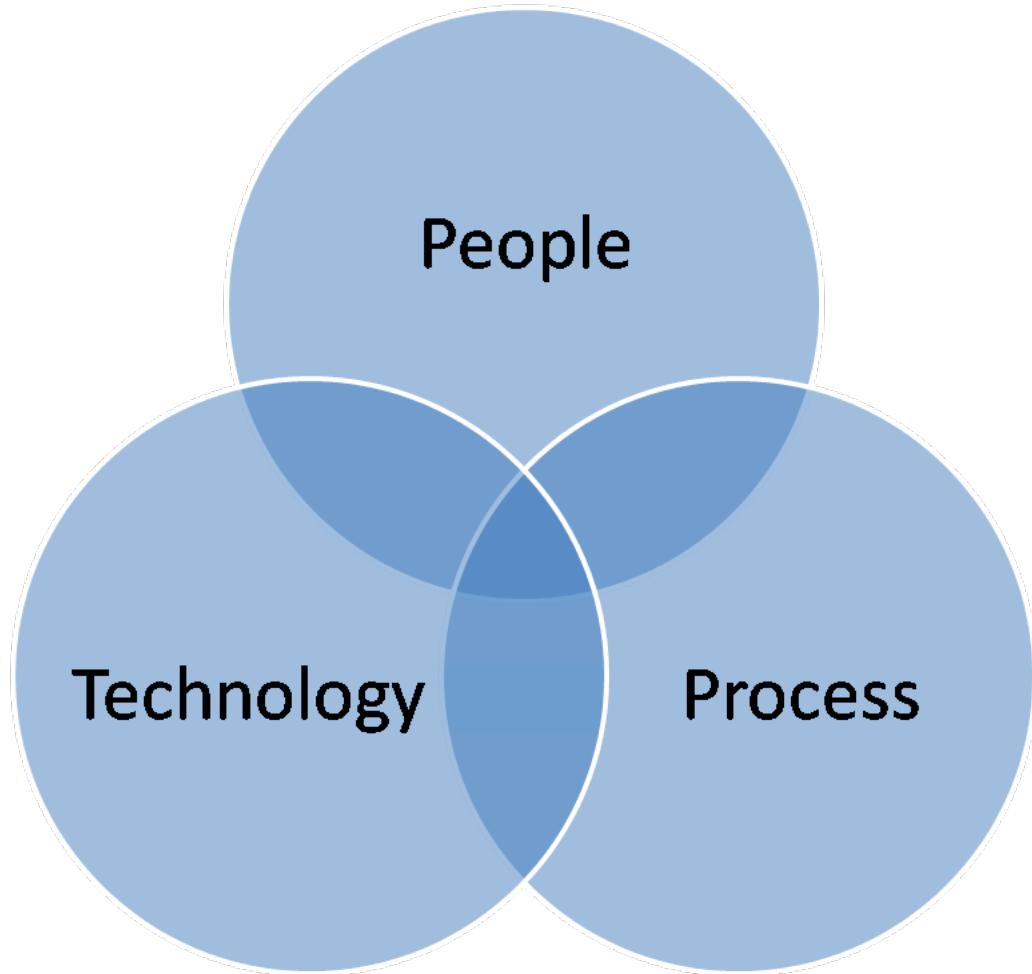
# Continuous Delivery = Automate Everything



Are you confident?



# Confidence Trifecta



Apply for a credit card in 1990



# Apply for credit card 2020



the  
**POINTS**  
↗**GUY**

Level of automation up to you



# Machines should help

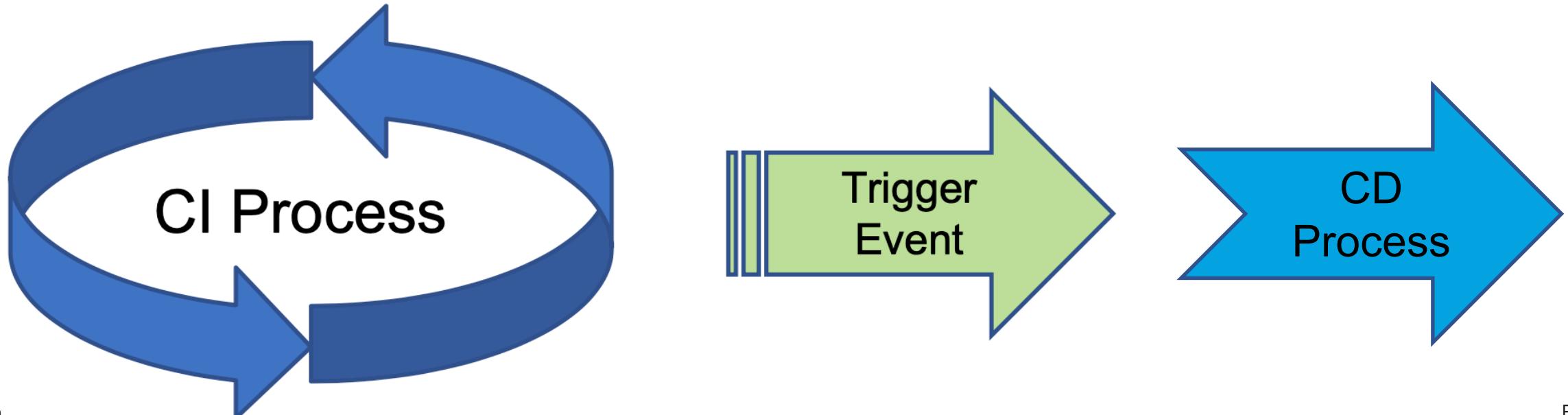


# Doing CD Right With Harness & K8s



# When Does the CI Cycle End?

- Continuous Integration ends when you have a “releasable artifact”
- A releasable artifact could be many things: Docker image, VM image, AMI, .jar file, other packaged software bundle
- Not every successful pass through the CI cycle yields a releasable artifact though
- A “trigger” event ends the CI cycle and begins the CD process



# What Causes the Trigger?

- Sometimes it's based on business needs, sometimes it's arbitrary:

We've got to get  
Penultimate Fantasy 17 out  
before Cyber Monday or  
we're toast!

Janette McGamexec  
CTO, Second To Last Games, Inc.

I say we only release  
on the full moon! The  
pack has spoken.

Michael J. Teenwolf  
Head Developer, Werewolves of London, Ltd.

- Sometimes it's just because deploying is hard, and you can't ask your team to give up all their Saturday nights. So you settle on once a month/quarter/year.
- In a perfect world you'd release as often as you could . . .

# What Is Continuous Delivery?

"Continuous Delivery is the ability to get changes of tall types — including new features, configuration changes, bug fixes and experiments — into production or the hands of users, *safely* and *quickly* in a *sustainable* way."

Jez Humble, Founder continuousdelivery.com

# How Often Is Continuous Delivery?

- Lots of factors determine this of course, but if your CI process is dialed in and your main branch is kept as bug free and "deployable" as possible you can up the frequency of your deployments
- The target is to achieve an interval that meets the business AND engineering needs
- The process is to increase your deployment frequency over time. Allow "virtuous cycles" in your CI and CD systems to constantly shrink your deployment interval

# Harness.io SaaS - A Product of CD

- Harness deploys new features and feature updates every day at 5 PM
- Bug fixes and security updates are deployed immediately once they are fully vetted – sometimes multiple times per day

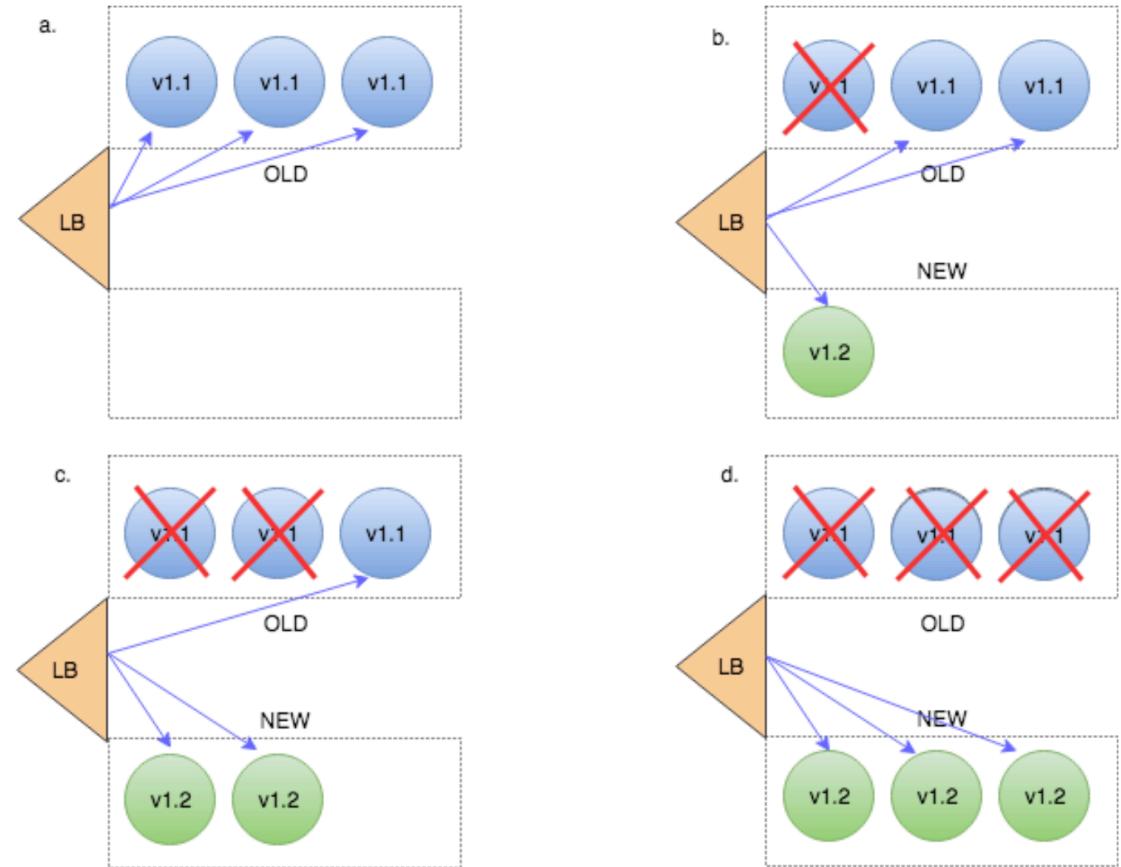


# How To Upgrade / Install New Artifacts

- **Rolling Upgrade** – for many PaaS's this is the default deployment. Slowly scale up the new version while simultaneously slowly scaling down the old version.
- **Blue/Green Deployment** – stand up a complete deployment of the new version (blue) while leaving a complete copy of the old version (green) up and running.
- **Canary Deployment** – A variation of Blue/Green. Stand up a small deployment of the new version (the canary) and direct a small portion of incoming traffic to the canary

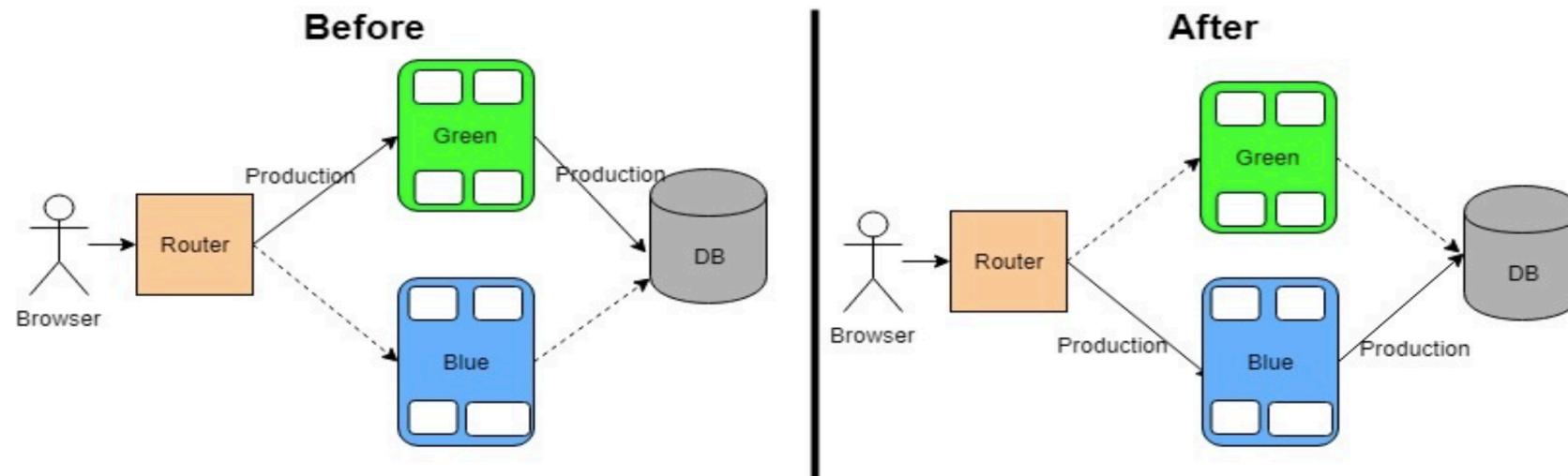
# Rolling Updates - Pros and Cons

- **Pro:** Easy to do – the default upgrade for many PaaS's
- **Pro:** Respects hardware limitations / footprints
- **Pro:** Minimally disruptive if everything goes well
- **Con:** Slow – can take a long time depending on what you're deploying
- **Con:** Rollback is quite difficult / time consuming / maybe impossible – roll forward more likely



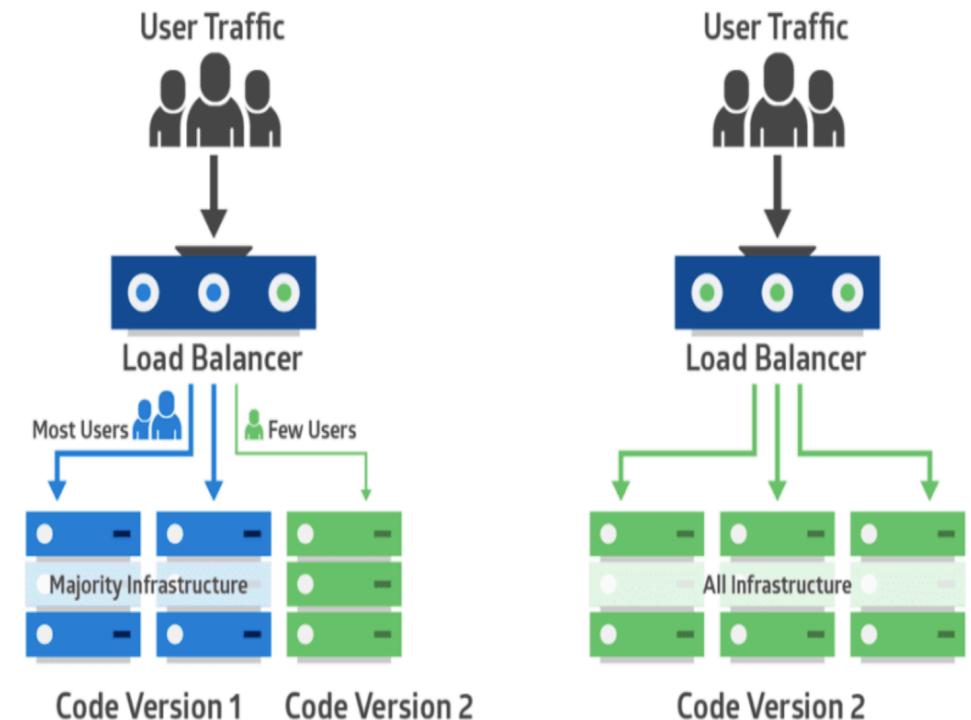
# Blue / Green Deployments - Pros and Cons

- **Pro:** Very easy rollback – just flip back to the green version
- **Pro:** Easy networking – flip a switch on the LB
- **Con:** Slow – can take a long time to get the entire Blue version up and running
- **Con:** Hardware / cloud footprint intensive – requires twice the infrastructure while both versions are up and running

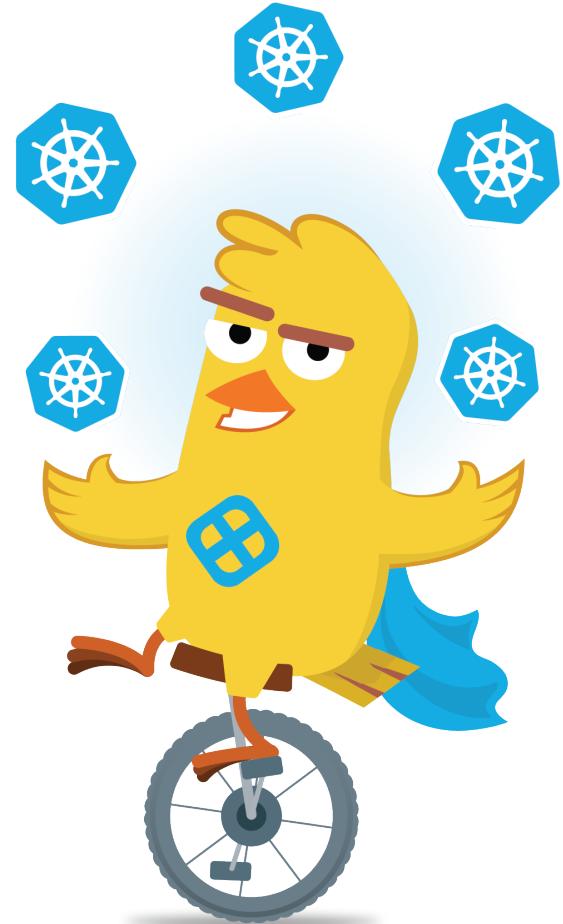
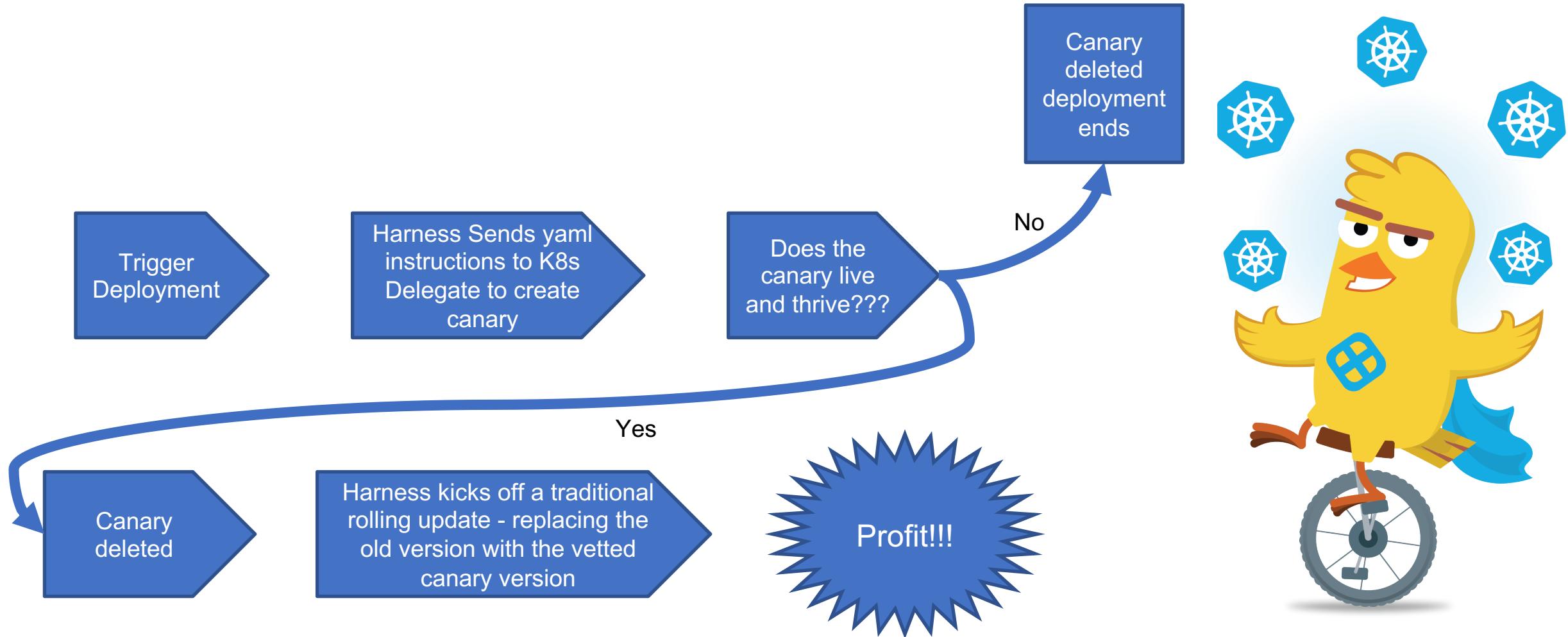


# Canary Deployment - Pros and Cons

- **Pro:** Minimal extra hardware/cloud footprint
- **Pro:** Rapid deployment and feedback – canaries come up quickly and give feedback on quality of deployment quickly
- **Pro:** Easy-ish rollback – just kill the canary and reshape the network traffic back to the original
- **Con:** Very difficult to setup – requires complex networking and orchestration deployment



# Kubernetes Canary Workflow



# Drew's: There I Said It Out Loud



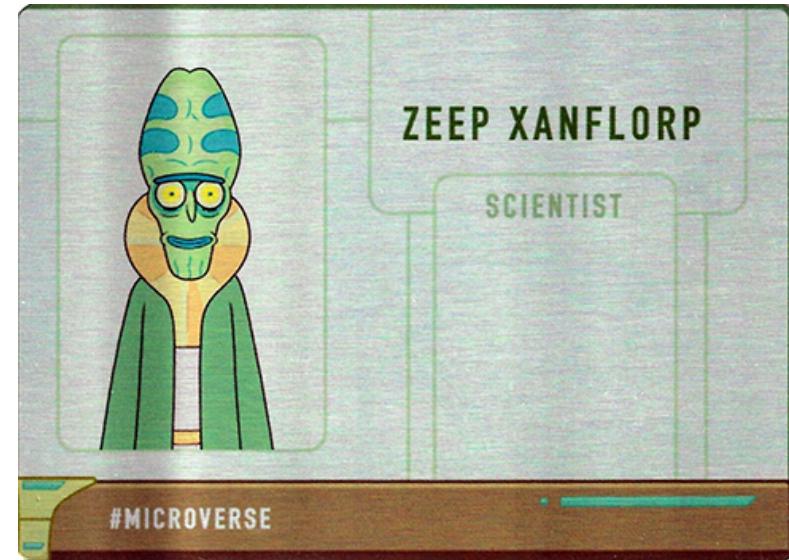
# There I Said It Out Loud - K8s and Databases



Source:

<https://cloud.google.com/blog/products/databases/to-run-or-not-to-run-a-database-on-kubernetes-what-to-consider>

# T.I.S.I.O.L. - K8s Pod Network = Network



# T.I.S.I.O.L. - Consider Support Too When Choosing K8s Distro



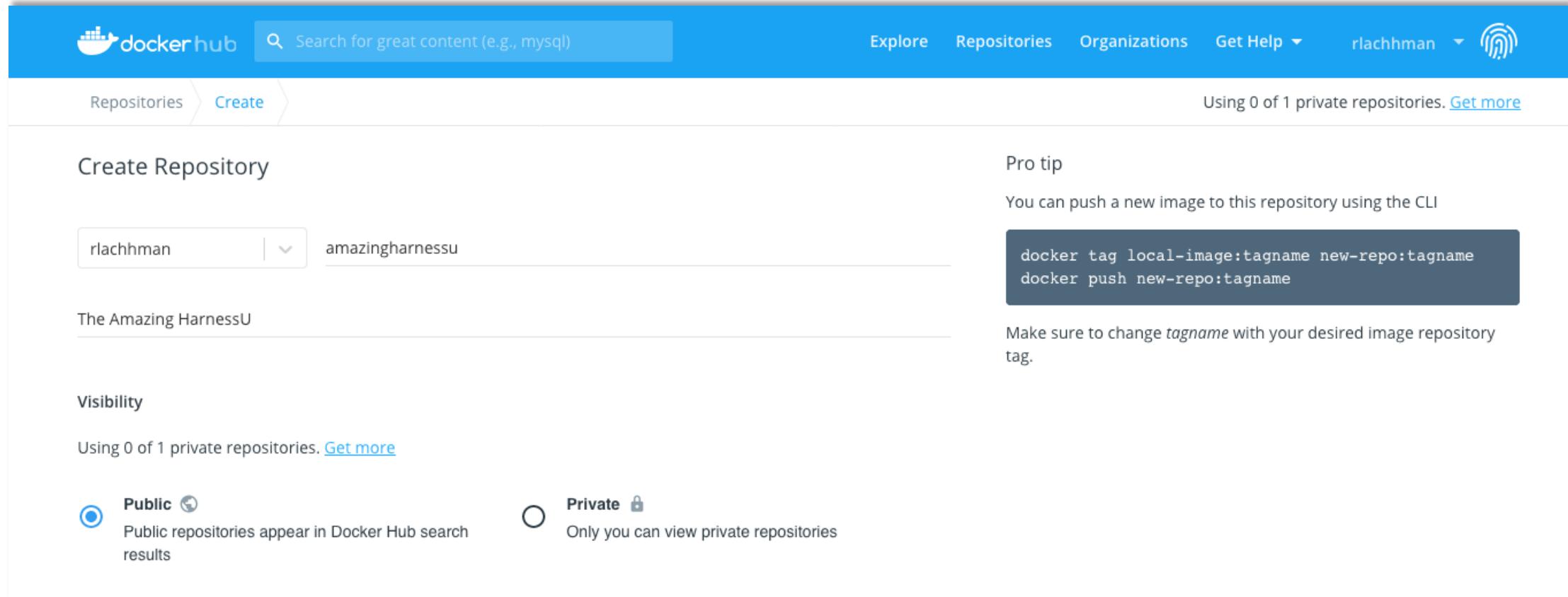
# Lab Time!



# Harness Setup Lab Steps and Demo

- <https://github.com/harness-training-dept/HarnessU-CI-CD-Workshop/blob/main/README.md>

# Docker Hub Set Up



The screenshot shows the Docker Hub interface for creating a new repository. At the top, there's a blue header bar with the Docker Hub logo, a search bar, and navigation links for Explore, Repositories, Organizations, Get Help, and a user account dropdown. Below the header, the main content area has a breadcrumb navigation showing 'Repositories' and 'Create'. It displays the message 'Using 0 of 1 private repositories.' with a 'Get more' link. The central part of the page is titled 'Create Repository' and includes fields for the owner ('rlachhman') and repository name ('amazingharnessu'). A 'Description' field contains the placeholder 'The Amazing HarnessU'. To the right, a 'Pro tip' section provides CLI commands: `docker tag local-image:tagname new-repo:tagname` and `docker push new-repo:tagname`, with a note to change `tagname`. Below this, a 'Visibility' section offers two options: 'Public' (selected) and 'Private'. The 'Public' option indicates that repositories appear in search results, while the 'Private' option ensures only the owner can view them.

Search for great content (e.g., mysql)

Explore Repositories Organizations Get Help rlachhman

Repositories Create Using 0 of 1 private repositories. [Get more](#)

### Create Repository

rlachhman amazingharnessu

The Amazing HarnessU

#### Visibility

Using 0 of 1 private repositories. [Get more](#)

**Public**  Public repositories appear in Docker Hub search results

**Private**  Only you can view private repositories

Pro tip  
You can push a new image to this repository using the CLI

```
docker tag local-image:tagname new-repo:tagname
docker push new-repo:tagname
```

Make sure to change `tagname` with your desired image repository tag.

# User/Repository:tag

 **rlachhman / amazingapp**

Sample Amazing App teaching folks how to use Drone CI. 

 Last pushed: 21 days ago

**Docker commands**

To push a new tag to this repository,

```
docker push rlachhman/amazingapp:tagname
```

[Public View](#)

# Fork Away!

The screenshot shows a GitHub repository page for the project 'firstdrone'. The URL in the address bar is <https://github.com/ravilach/firstdrone>. The repository has 2 stars and 8 forks. The 'Code' tab is selected, showing the master branch with 1 branch and 0 tags. A commit from 'ravilach' dated Aug 12 at a1008e8 is shown, which updated the Dockerfile, Drone.yaml, LICENSE, README.md, and main.go files. To the right, there's an 'About' section with a note about leveraging the sample structure for a first Drone Build, a 'Readme' link, and an Apache-2.0 License link. Below that is a 'Releases' section indicating no releases have been published, with a link to 'Create a new release'. At the bottom is a 'Packages' section indicating no packages have been published, with a link to 'Publish your first package'. A large 'Welcome to your first Drone project!' message is displayed at the bottom of the code area.

Search or jump to... Pull requests Issues Marketplace Explore

ravilach / firstdrone Unwatch 2 Star 0 Fork 8 Fork your own copy

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags Go to file Add file Code

ravilach Update Dockerfile ... a1008e8 on Aug 12 31 commits

Dockerfile	Update Dockerfile	2 months ago
Drone.yaml	Update Drone.yaml	2 months ago
LICENSE	Initial commit	3 months ago
README.md	Update README.md	2 months ago
main.go	Update main.go	2 months ago

README.md

Welcome to your first Drone project!

About

Leverage this sample structure for your first Drone Build

Readme

Apache-2.0 License

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

<https://github.com/ravilach/firstdrone>

# Thank You!

