

please

Ask questions  
through the app



*Rate Session*

Thank you!



A photograph of a modern, multi-story glass building, likely a corporate headquarters, with a blue overlay. The building has large windows and a prominent corner. The text is overlaid on the left side of the image.

# Modern Continuous Delivery

A journey in four acts

**ThoughtWorks®**



A man with short dark hair and a beard is sitting on a light-colored corduroy couch. A large black and white speckled dog is lying on the couch in front of him, looking towards the camera. The background is slightly blurred, showing a bookshelf and a wooden door.

# WHO AM I?

## KEN MUGRAGE

THOUGHTWORKS OFFICE OF THE CTO

10 YEARS WORKING WITH  
THOUGHTWORKS CD CLIENTS

GLOBAL ORGANIZER FOR  
DEVOPSDAYS

IN DENIAL ABOUT PEOPLE USING  
DEVOPS ENGINEER AS A JOB TITLE



ThoughtWorks®

# ACT I

UNITED KINGDOM – 2005



# We knew everything

- Signed Agile Manifesto
- Defined Continuous Integration
- Created the first (or second) CI server
- Created Selenium

# Java – Write once, run anywhere

- Developed a system on Windows laptops to be deployed to a Solaris cluster
- Did all the right Continuous Integration things
- One small issue...

It didn't work in production



# Conan The Deployer

- Massive shell script
- Automated deployment to a cluster after every successful CI run
- Deployment became a non-issue

# The Output

- Jez Humble, Dan North and Chris Read write “The Deployment Product Line”
- Jez Humble and David Farley release the book “Continuous Delivery”
- Continuous Delivery becomes an expectation



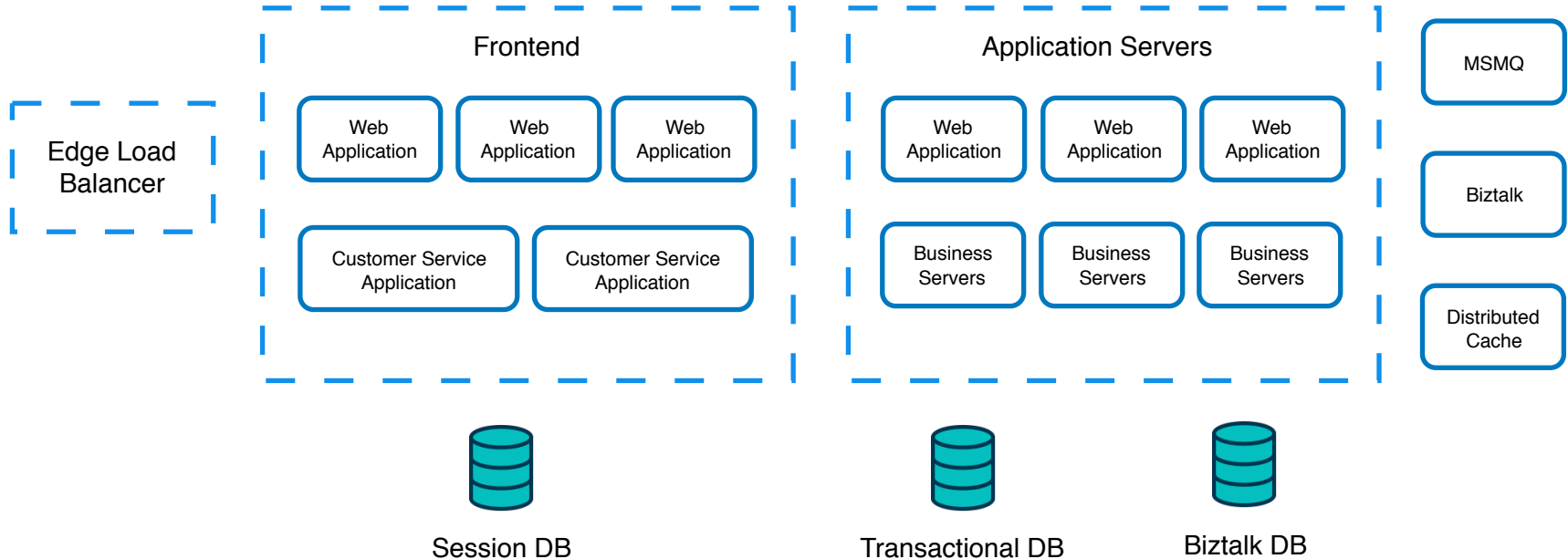
ThoughtWorks®

# ACT II

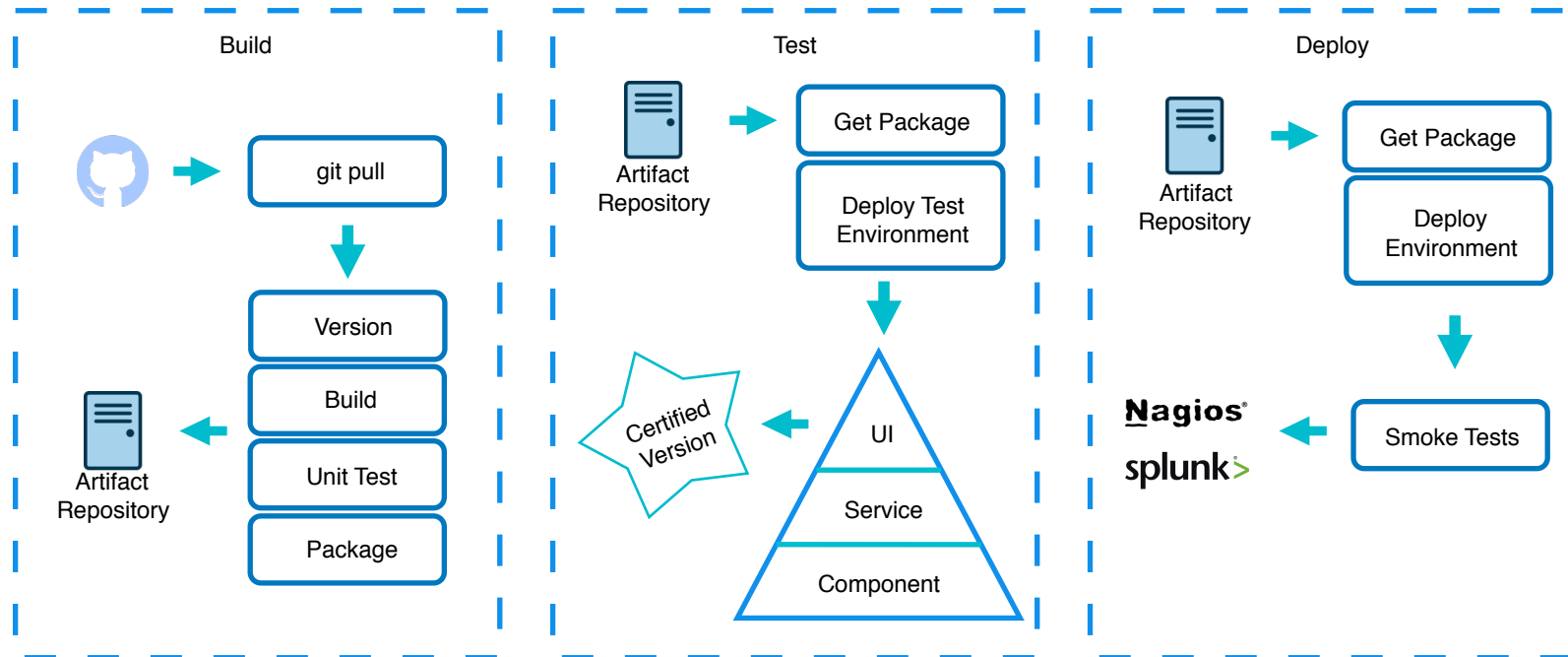
UNITED KINGDOM – 2010



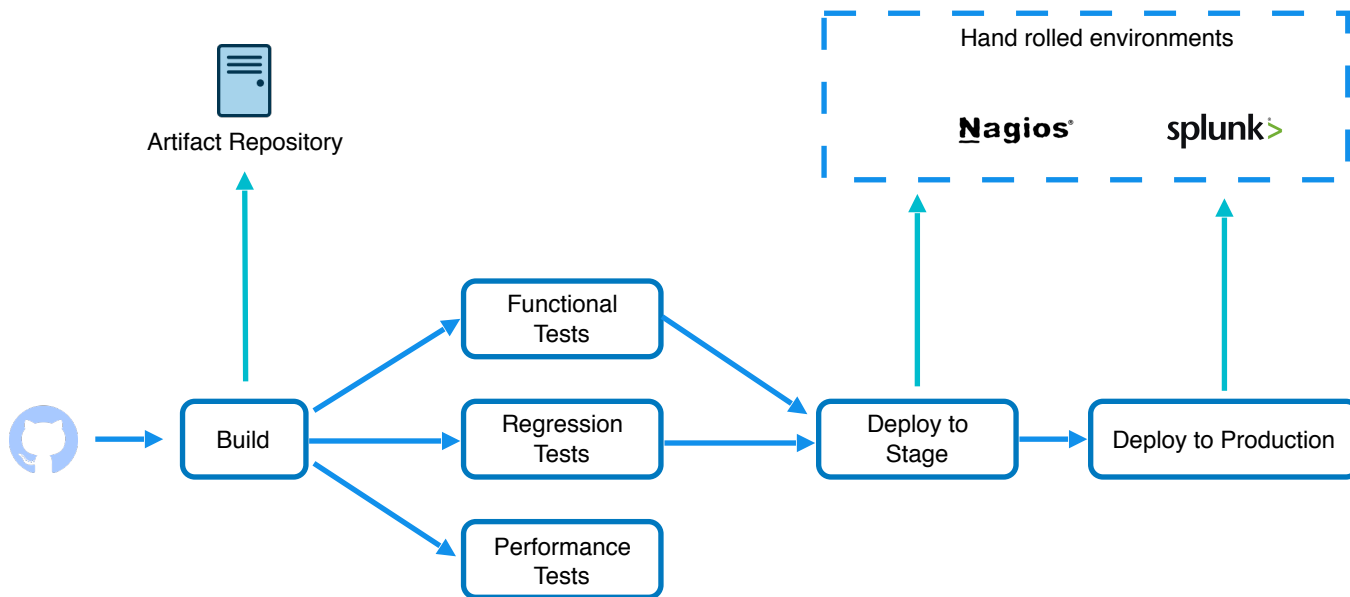
# Fairly typical architecture



# Traditional Continuous Delivery



# Traditional Continuous Delivery





# The Good

- Reproducible builds
- Generate package once
- High level of automation
- Safety net with automated test stages

# The Bad

- Entire system deployed at once
- Releases were large
- Browser tests were flaky
- Rollbacks were hard

# The Output

- Sam Newman releases the book “Building Microservices”
- Kief Morris releases the book “Infrastructure as Code”
- Created Gauge open source testing framework
- Created Taiko to combat flaky browser tests

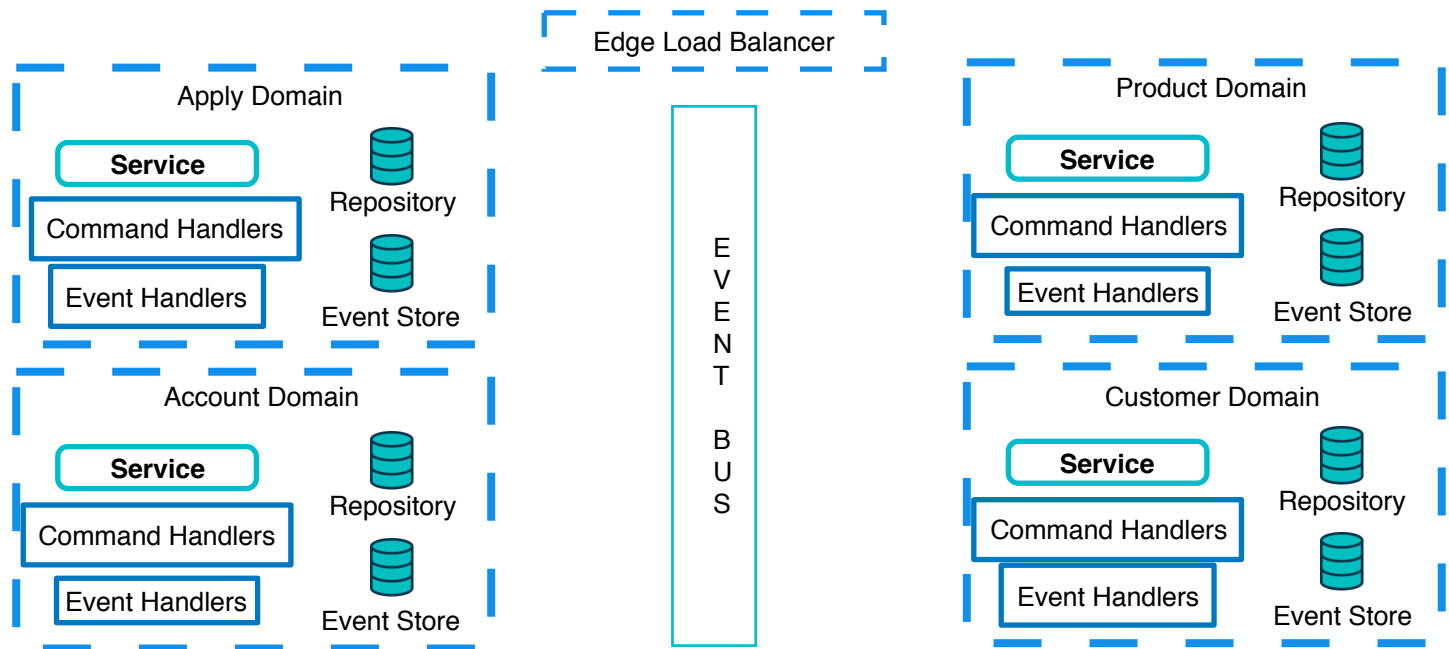
ThoughtWorks®

# ACT III

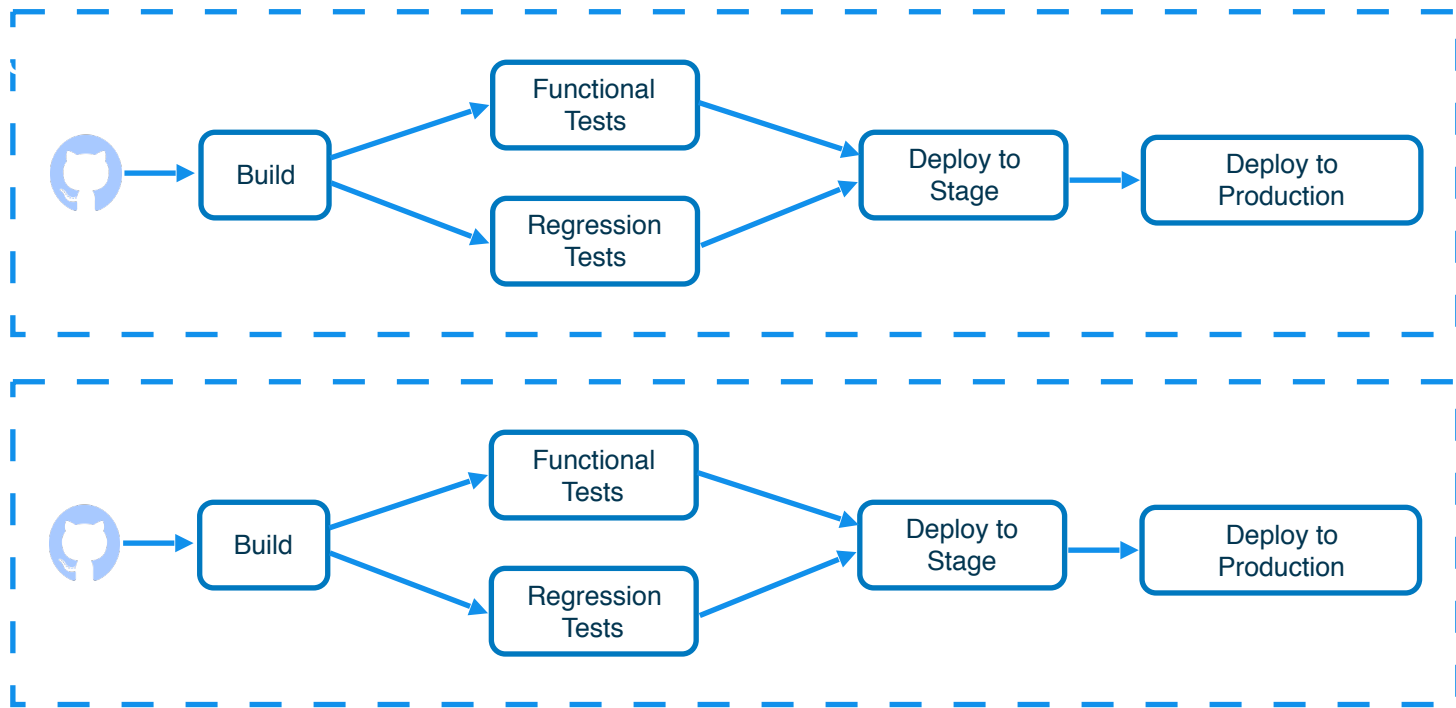
UNITED STATES – 2019



# Financial Services Platform



# Modern Pipelines





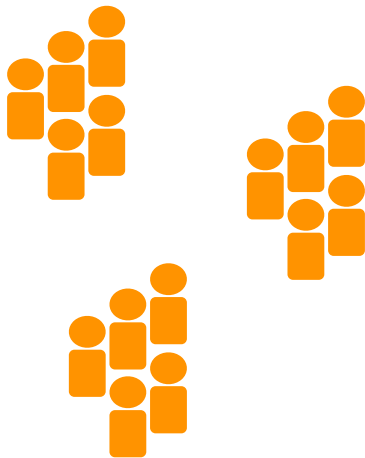
ThoughtWorks®

# ACT IV

THE DETAILS

# Traditional Structure

Development Teams



QA Team

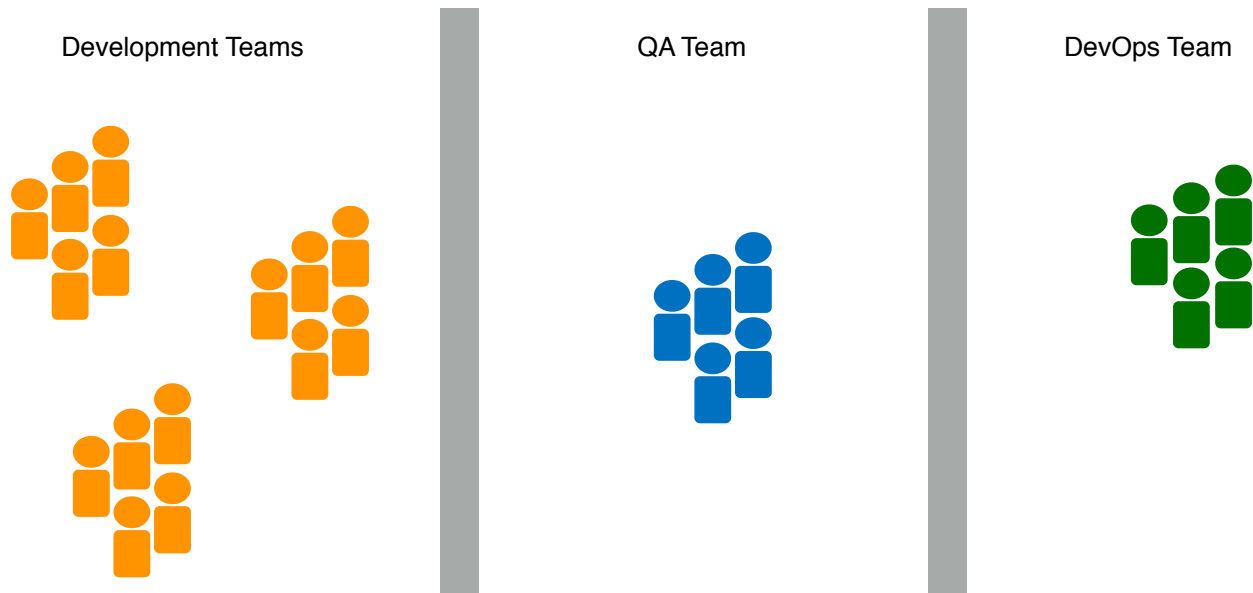


Operations Team

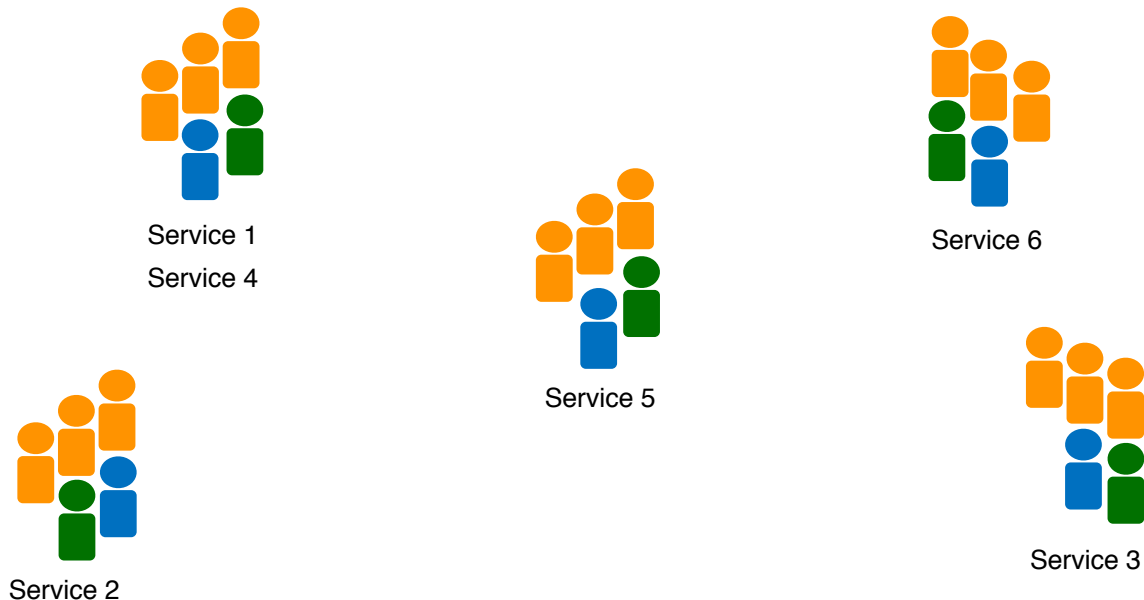




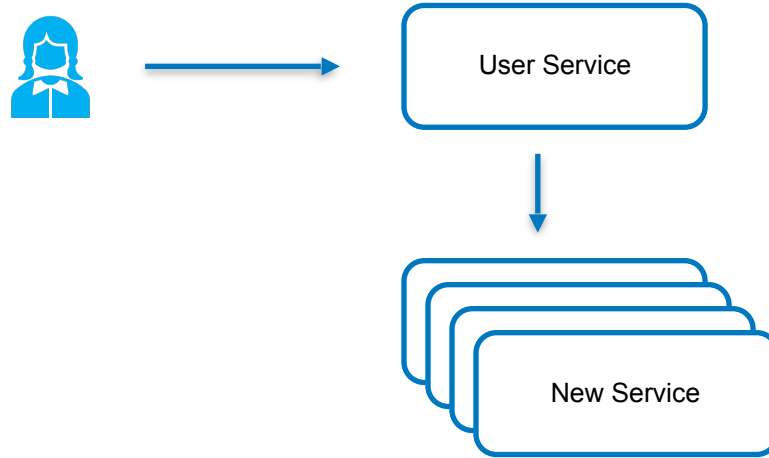
# Renaming Ops isn't a solution



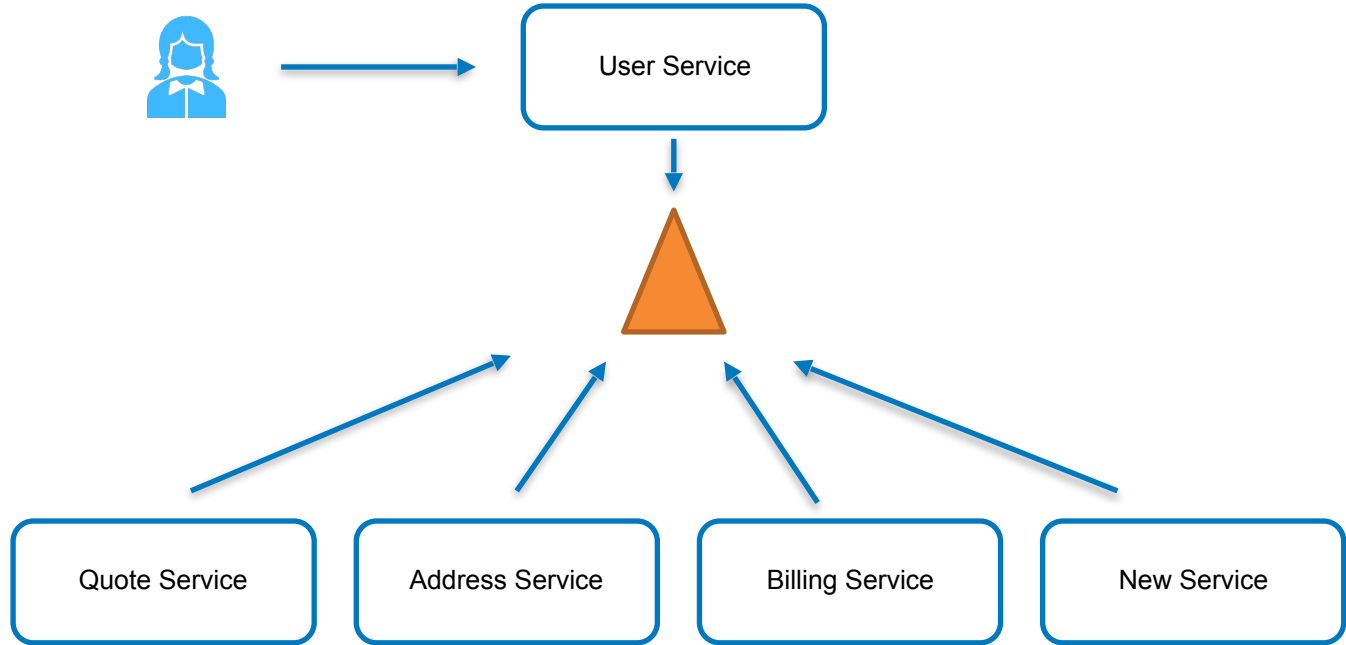
# Products over Projects (do the DevOps)

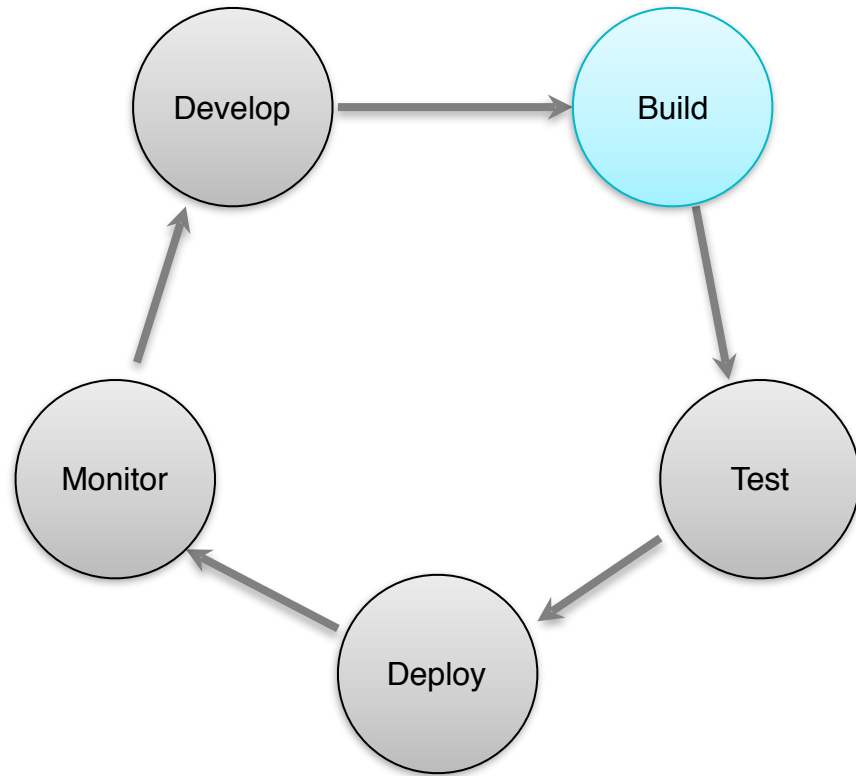


# WITHOUT EVENTS

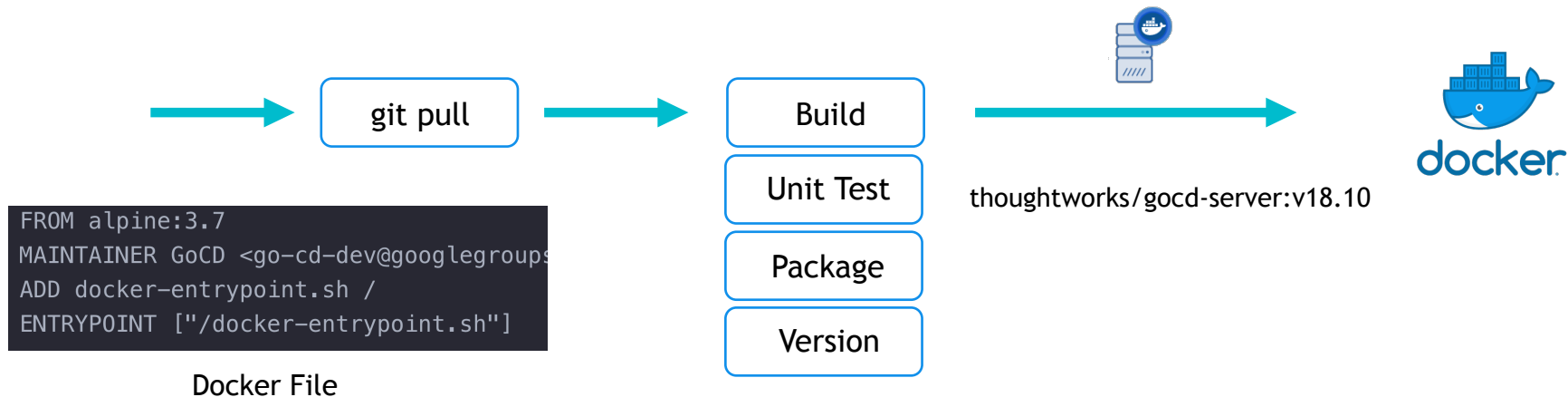


# EVENT NOTIFICATION





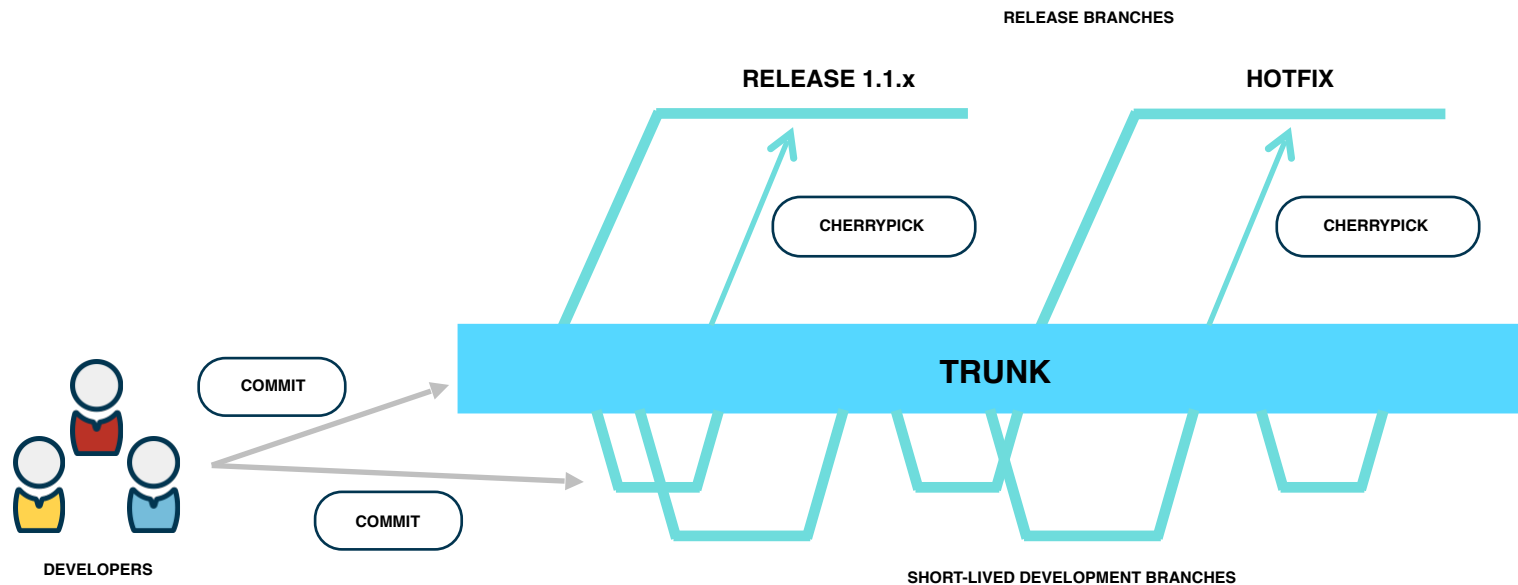
# The New Build Artifact



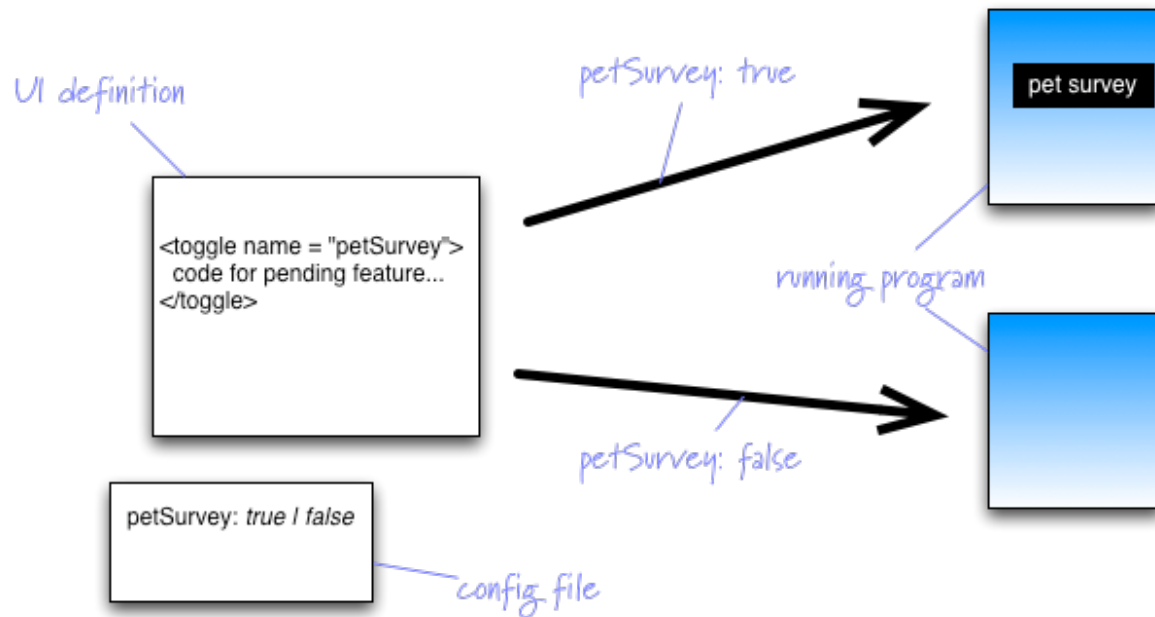
# A modern Continuous delivery pipeline

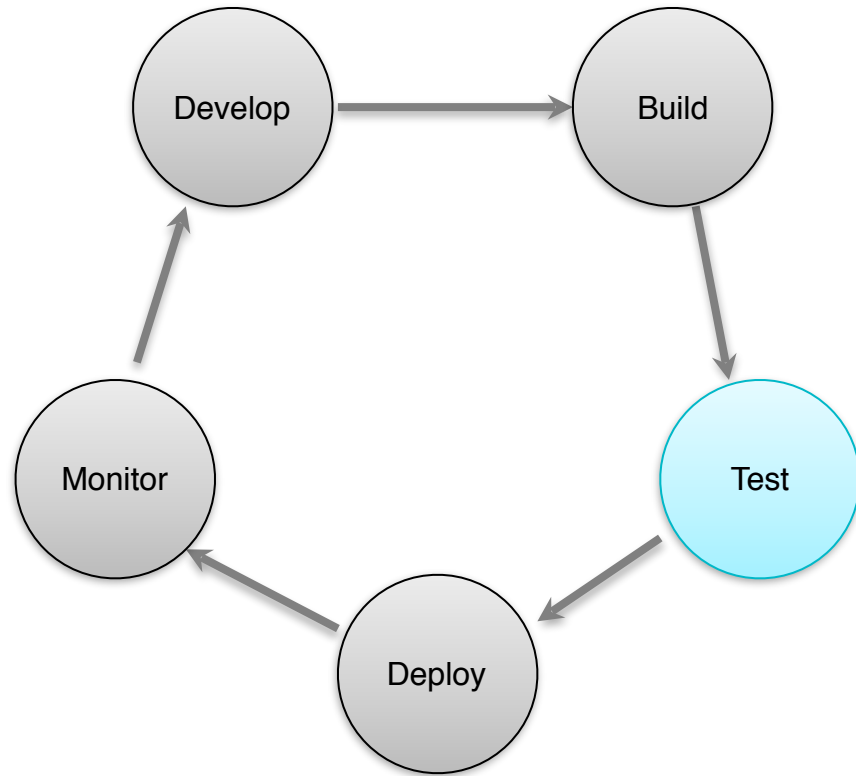


# Trunk Based Development

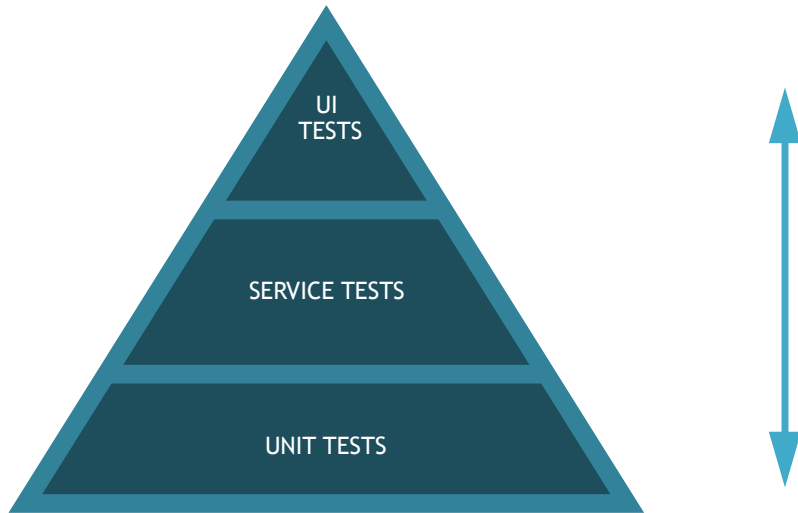


# Feature Toggles

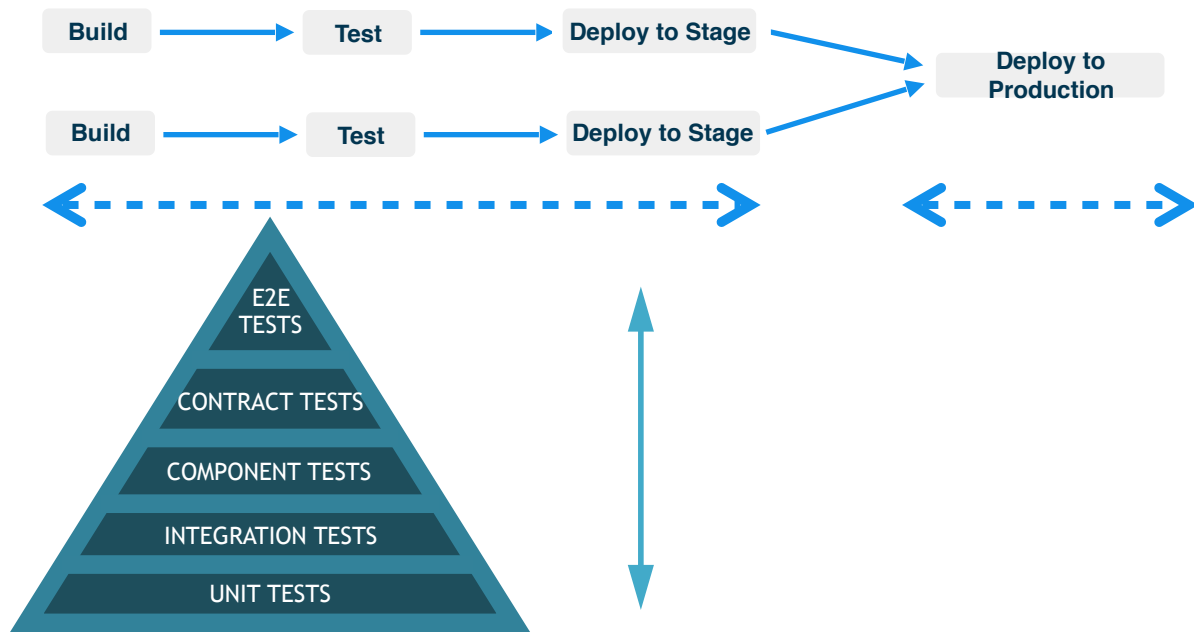




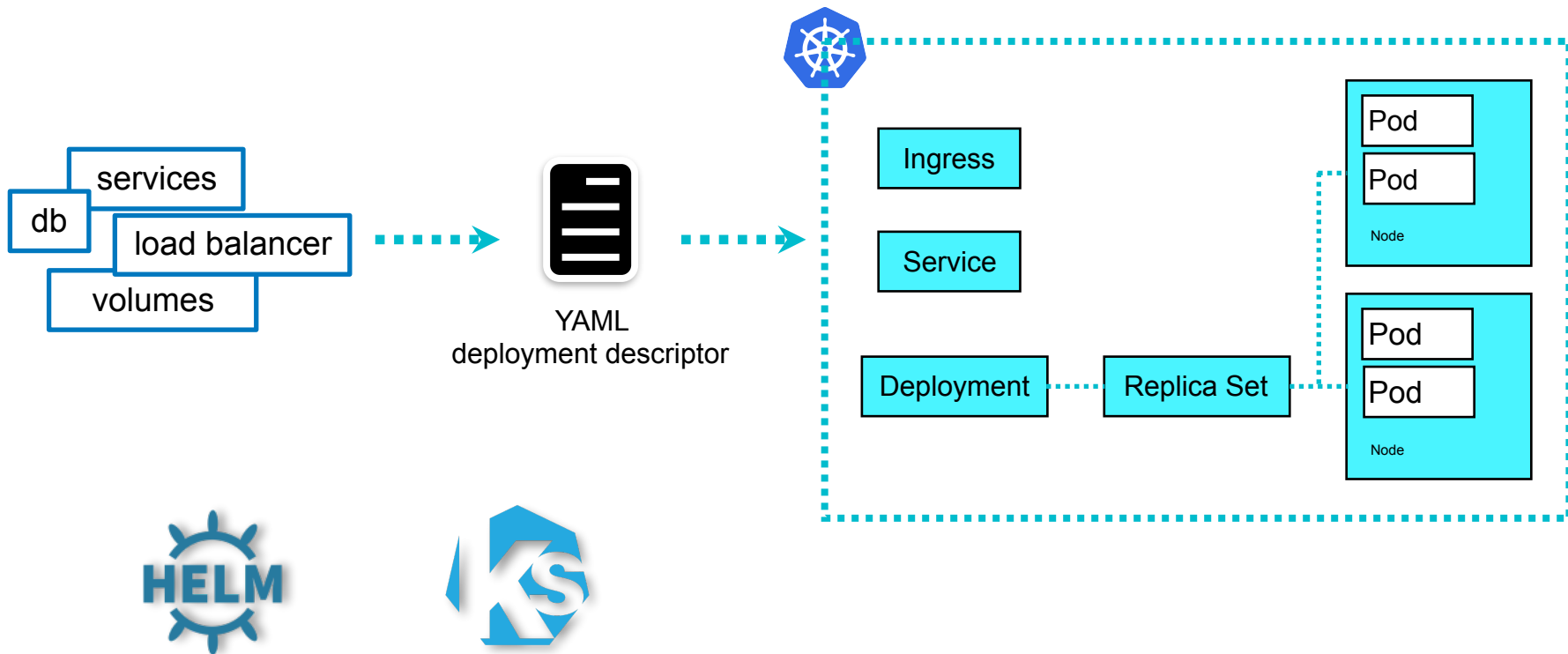
# The Test Pyramid



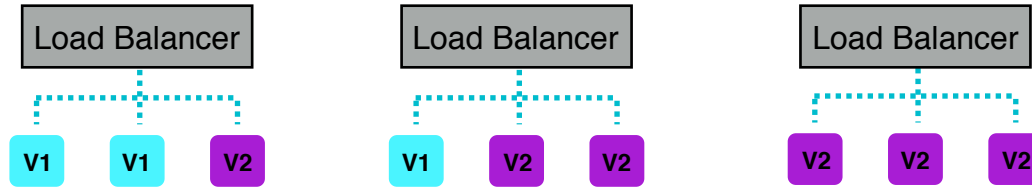
# The Test Pyramid In Context



# Declarative Deployments



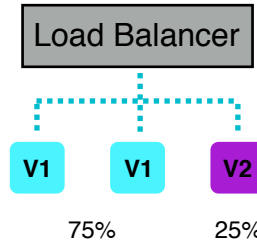
# Deployment Strategies



Rolling Update



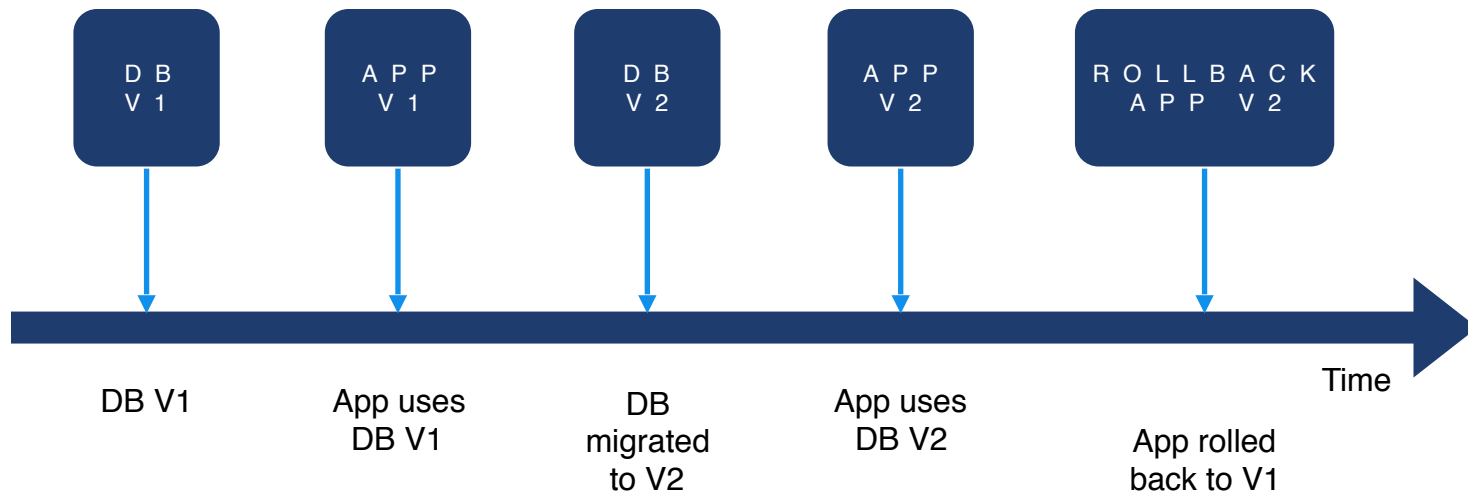
Blue Green Deployment



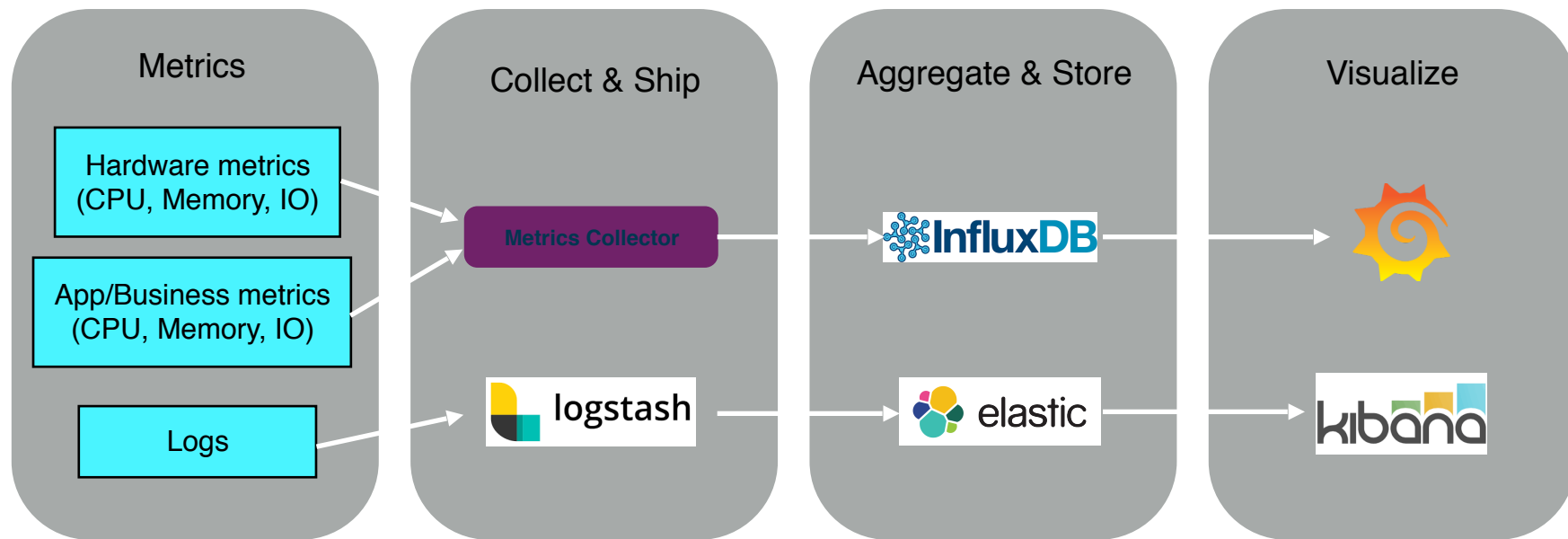
Canary Deployment



# Release DB Changes Out Of Band



# Monitoring and observability



# Monitoring and observability

honeycomb.io

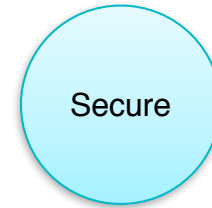
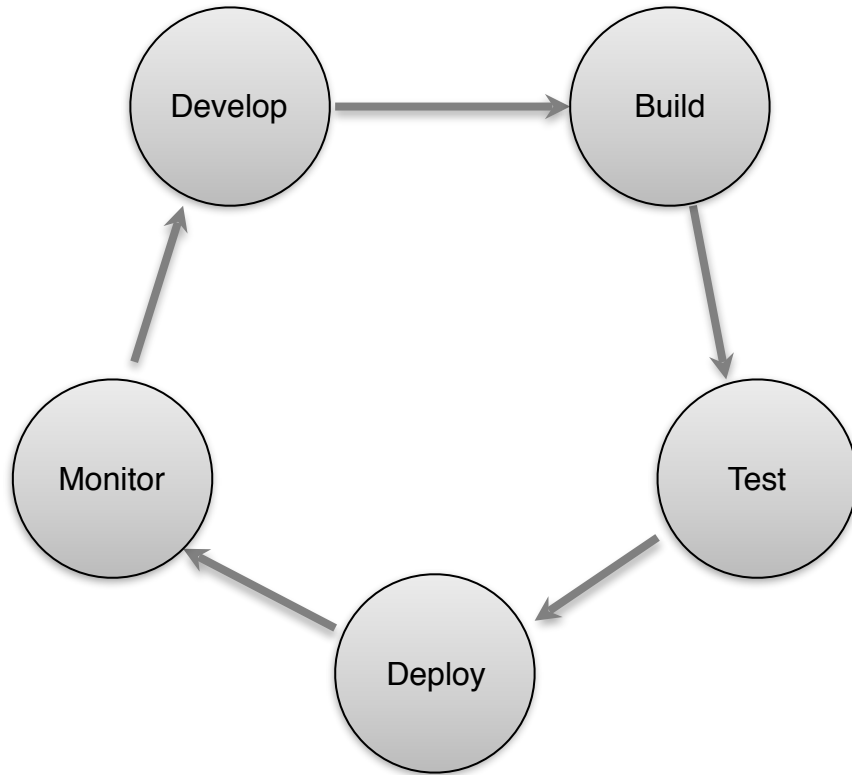
```
"time_total_seconds": %{{time_total}},  
"time_namelookup_seconds": %{{time_name}},  
"time_connect_seconds": %{{time_connect}},  
"content_type": "%{{content_type}}",  
"http_code": %{{http_code}},  
"speed_download": %{{speed_download}},  
"url": "%{{url_effective}}"
```

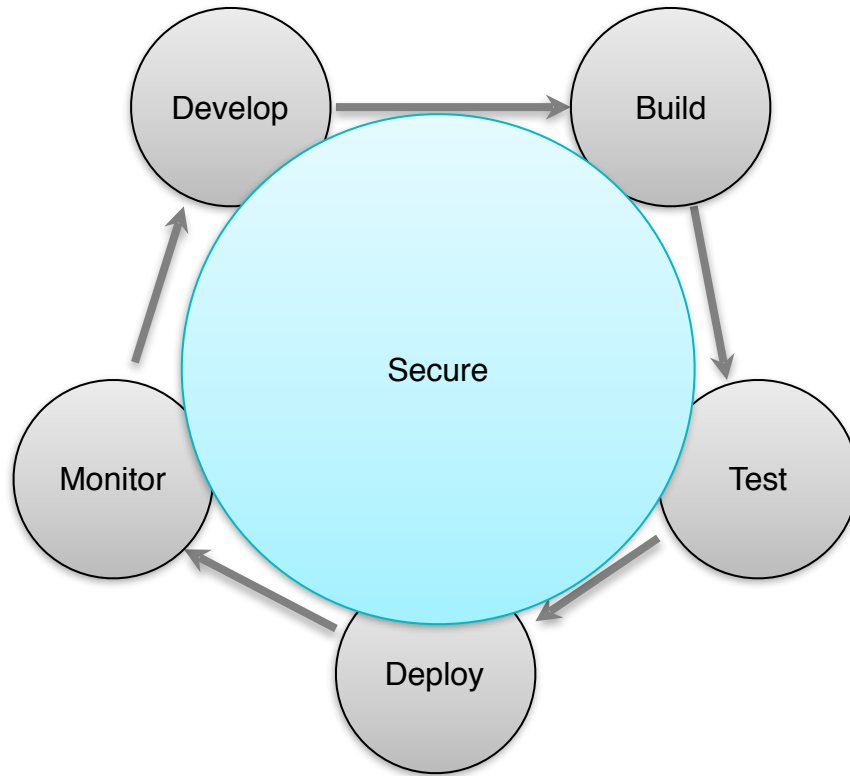
High cardinality events



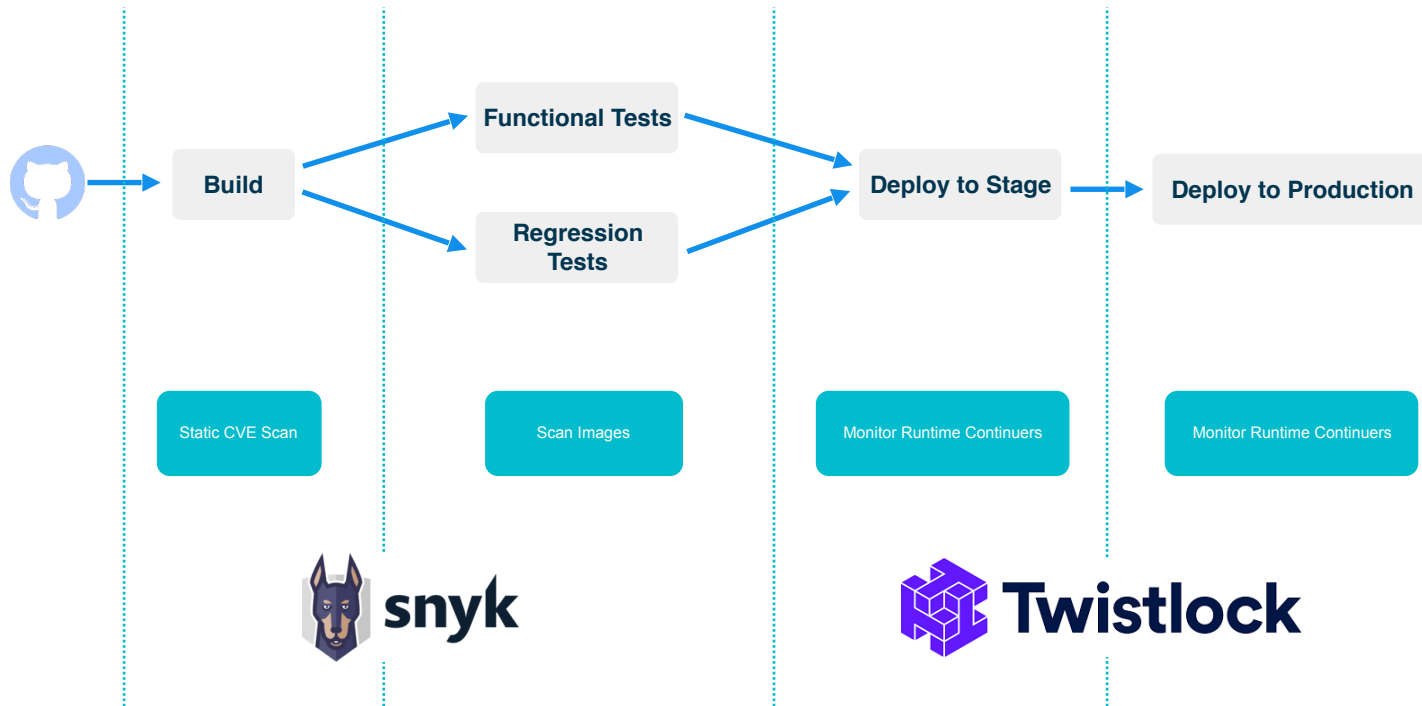
Observability

@kmugrage

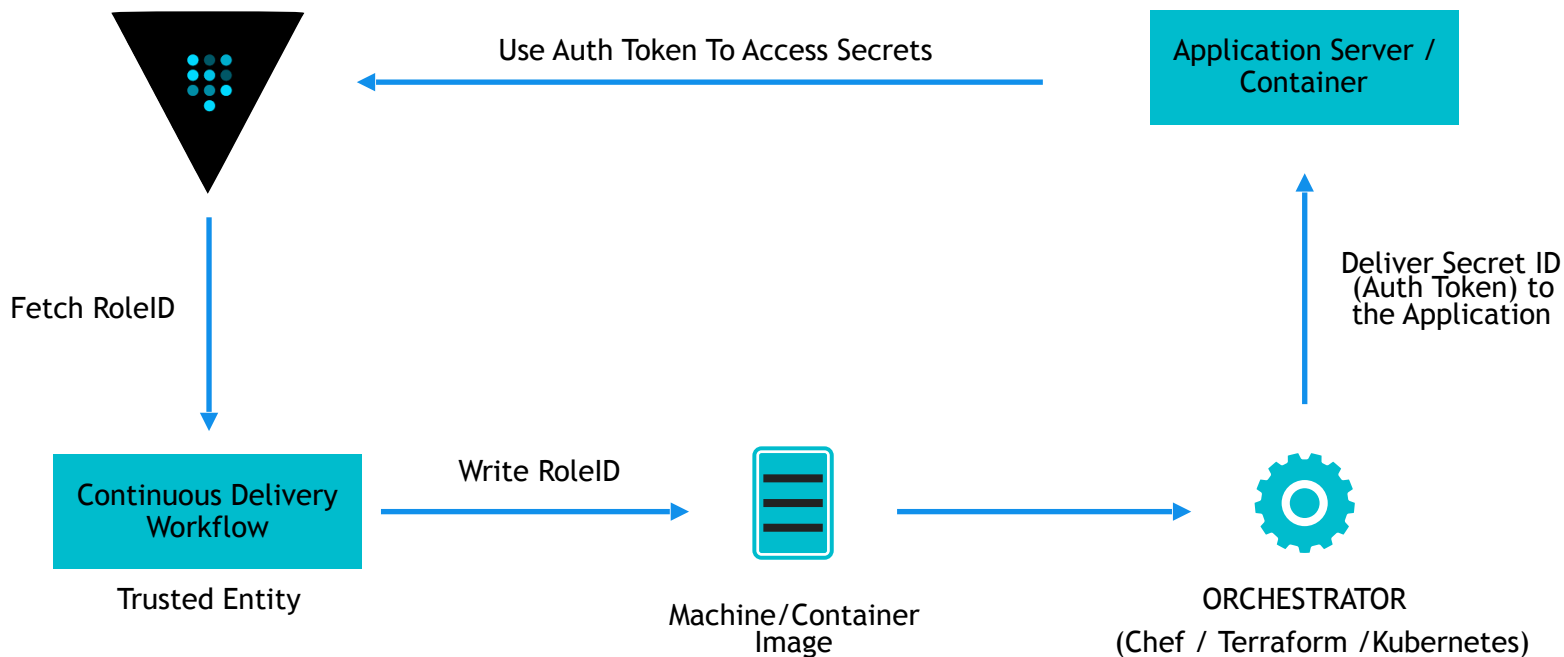




# Vulnerability Planning



# Secrets Management



ThoughtWorks®

# EPILOGUE



# In the future

- Teams must be structured to own small pieces
- Docker and Kubernetes are here to stay... until they aren't
- You must test in production. Own it and architect for it
- Security is everybody's job

# Resources & Citations

- <http://agilemanifesto.org/>
- <https://martinfowler.com/articles/originalContinuousIntegration.html>
- [https://continuousdelivery.com/wp-content/uploads/2011/04/deployment\\_production\\_line.pdf](https://continuousdelivery.com/wp-content/uploads/2011/04/deployment_production_line.pdf)
- <https://martinfowler.com/articles/microservice-trade-offs.html>
- <https://martinfowler.com/articles/practical-test-pyramid.html>
- <https://trunkbaseddevelopment.com/>
- <https://martinfowler.com/articles/feature-toggles.html>
- <https://docs.honeycomb.io/learning-about-observability/intro-to-observability/>
- <https://martinfowler.com/articles/201701-event-driven.html>

# Thank You

Ken Mugrage - @kmugrage

**ThoughtWorks®**

*Please*

**Remember to  
rate this session**

*Thank you!*

