

Production Deployments

The easiest part of the day

Stephen Shary

stephen.shary@gmail.com

<https://www.linkedin.com/in/stephenshary/>

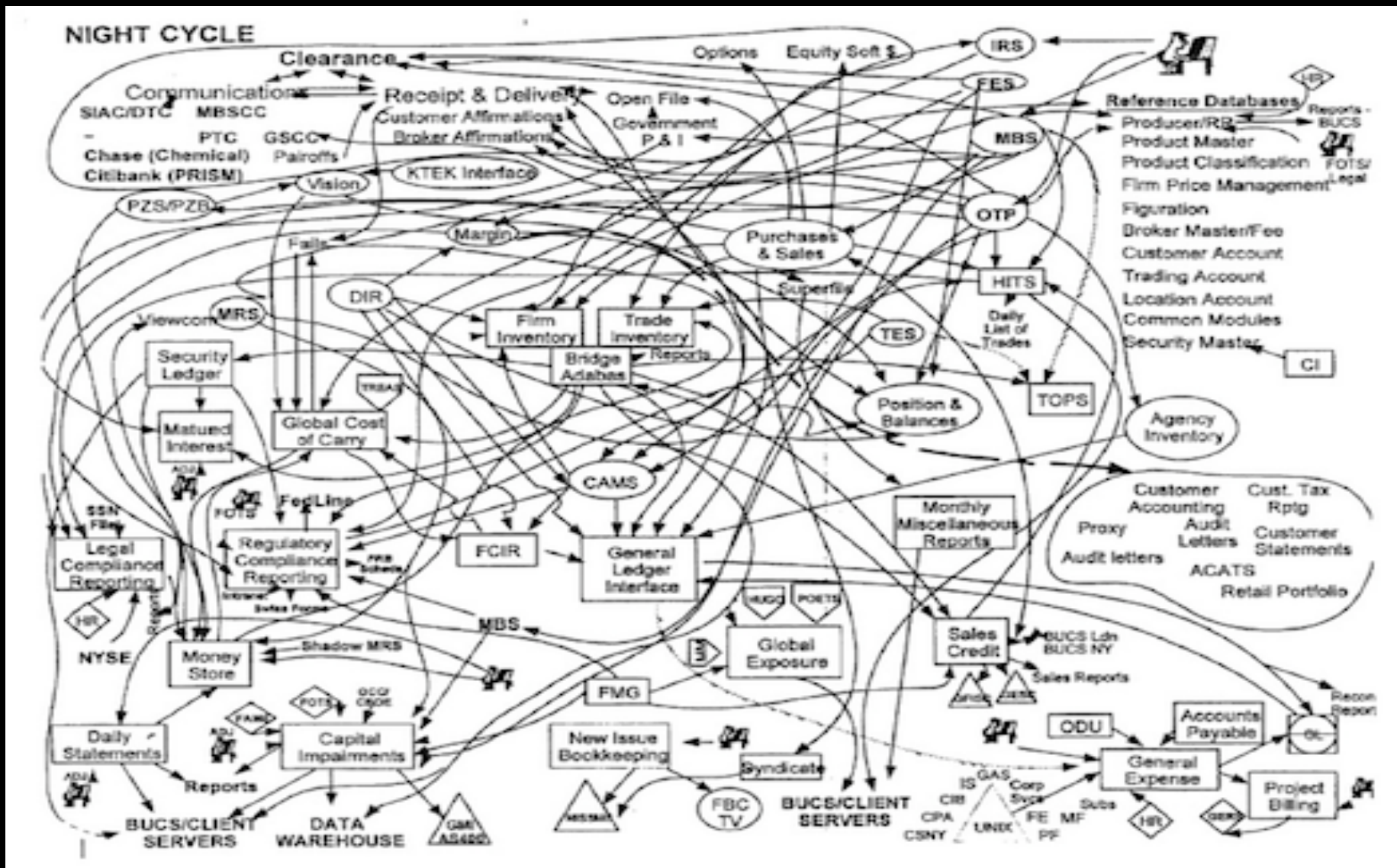
A Hypothetic Release

- 3 – 24 months of work.
- Still feeling rushed.
- You're scared...
- (I'm scared for you)...

Lets Begin



The Deployment Plan



Let's Do a Go-No-Go!

Devs: Go-no-go?



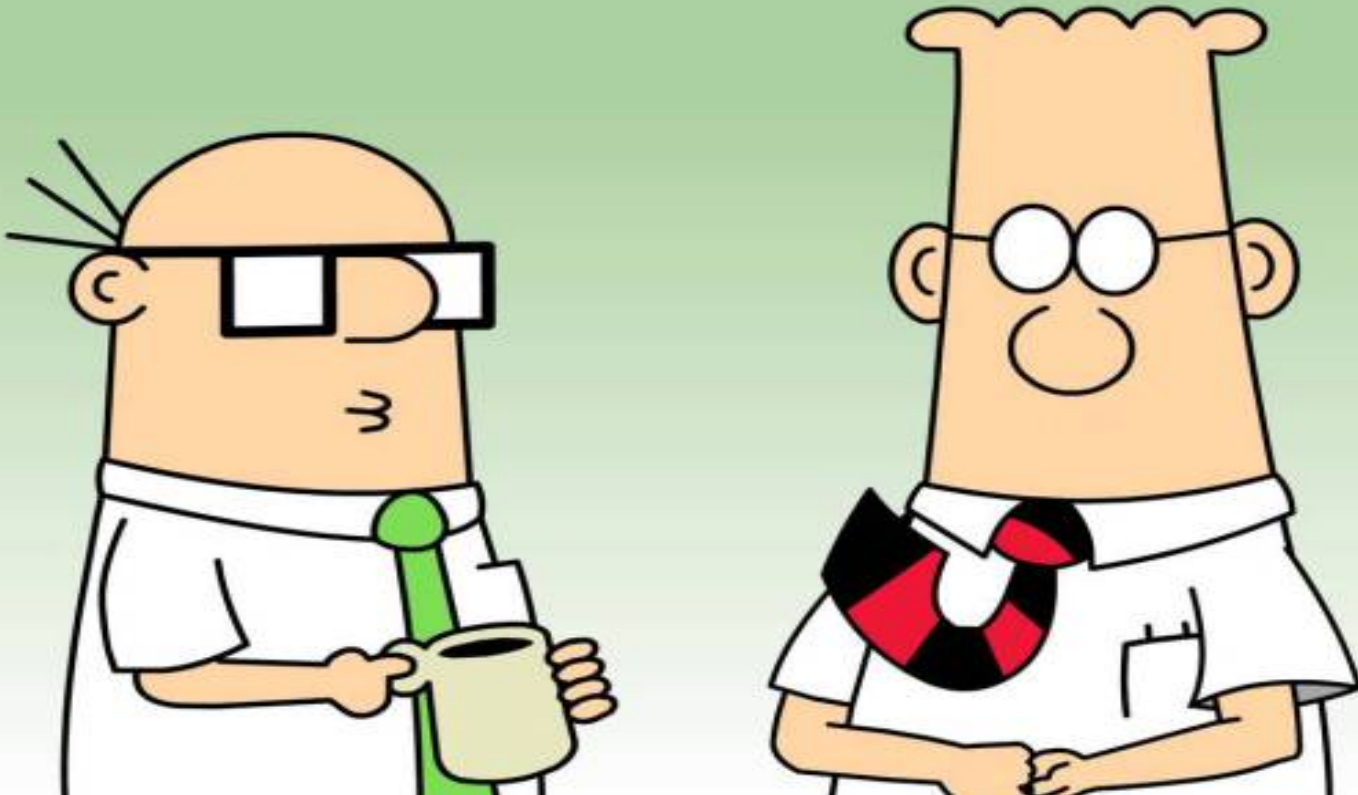
We forgot to call in sick today!

QA: Go-no-go?



Ready to start testing!

Ops: Go-no-go?



I really hate my life...

Business: Go-no-go?



I love trips!

Begin Step 1.

Later...

Initiate Step 49 of 50!

There's an Error

Operating system deployment did not complete successfully.

Please review the log files to determine the cause of the problem.

During the deployment process, 10 errors and 3 warnings were reported.

⌆ Details ...

```
FAILURE ( 5616 ): 1: Verify BCDBootEx
Litetouch deployment failed, Return Code = -2147467259 0x80004005
Failed to save environment to (80070057)
Failed to run the action: Install Operating System.
Unknown error (Error: 000015F0; Source: Unknown)
The execution of the group (Install) has failed and the execution has been aborted. An action failed.
Operation aborted (Error: 80004004; Source: Windows)
Failed to run the last action: Install Operating System. Execution of task sequence failed.
Unknown error (Error: 000015F0; Source: Unknown)
Task Sequence Engine failed! Code: enExecutionFail
Task sequence execution failed with error code 80004005
SetNamedSecurityInfo0 failed.
```


What Do you see in Prod!?!

Are you sure it's
running?

Do you see anything?

I hate everyone....

Later....



DEVELOPING STORY

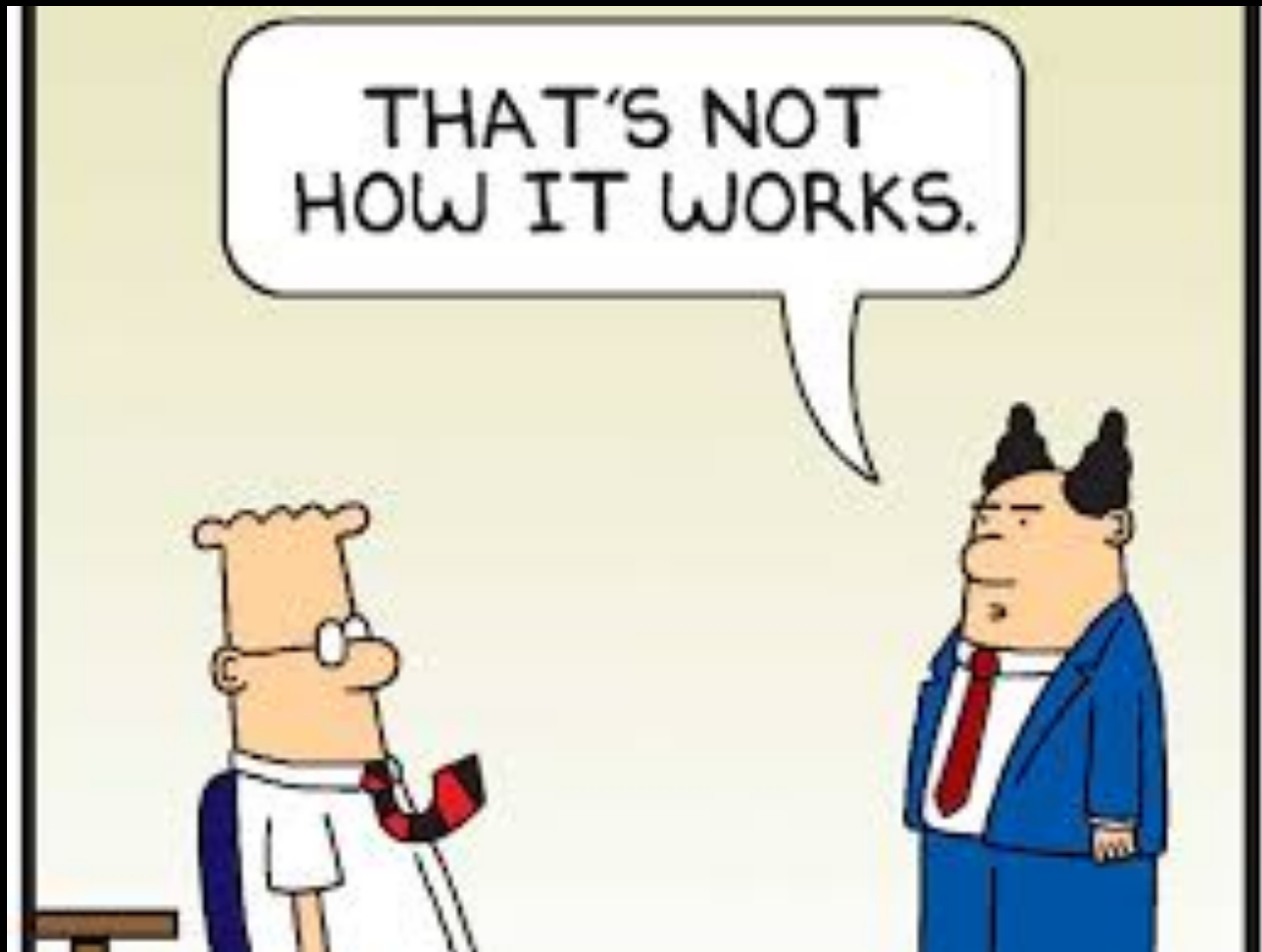
**DEPLOYMENT IGNITES MASSIVE FIREBALL
HUNDREDS FORCED TO EVACUATE**

7:08
18°



@GMA

Hyperbole?



What a Deployment Shouldn't Be

- Shuttle Launches
- Deployments to Mars.
- An irreversible surgery.
- Stop with the metaphors!

What could go wrong?

- Outage windows
- Performance issues
- Complicated deployment failure
- Security breaches
- Functionality failure
- Component incompatibility
- Looking incompetent
- Panic
- Being fired
- Heart attack
- Death
- Cats and dogs living together... total chaos.

Could it be different?

Probably not...
I'm just kidding
(maybe...)

Three Progressive Philosophies

- Always ready to go to prod.
- Use Software Defined Networking (SDN)
- Production feedback

NOTE: All require discipline.

Always ready to go to prod

- No more snapshots
- Separate state as much as you can.
- State services must be backwards compatible.
- Trunk based development
- Your code is:
 - Ready for prod
 - WIP

WIP Code in PROD

- Dead code.
- Isn't exposed.
- New api.
- Toggled off.

Toggles

- Simplest example:

```
if(config.isEnabled("newFeature")) {  
    return runNewFeature();  
} else {  
    return runOldJunk();  
}
```

- Requires maturity.
- Make a card/task/ticket to remove it.
- Make them HOT!

Toggle frameworks

- Spring Cloud
- Etcd
- Consul
- Zookeeper
- Shared mount (not recommended)

Small and Often

- This is super important.
- Bring the pain forward.
- Separate deployment risk from functionality risk.
- Make it a Kata

By practicing in a repetitive manner the learner develops the ability to execute those techniques and movements in a natural, reflex-like manner. Systematic practice does not mean permanently rigid. The goal is to internalize the movements and techniques of a kata so they can be executed and adapted under different circumstances, without thought or hesitation. A novice's actions will look uneven and difficult, while a master's appear simple and smooth. [wikipedia "kata"]

END ACT 1

Deployment = Downtime

- My releases cause outages.
- Outages = bad.
- Therefore: Releases = bad.

The Solution

- Software Defined Networking
- Routes traffic to your app.
- Can be manipulated live.
- Popular Solutions:
 - Nginx
 - HA Proxy
 - Varnish
 - Httpd

Who does this?

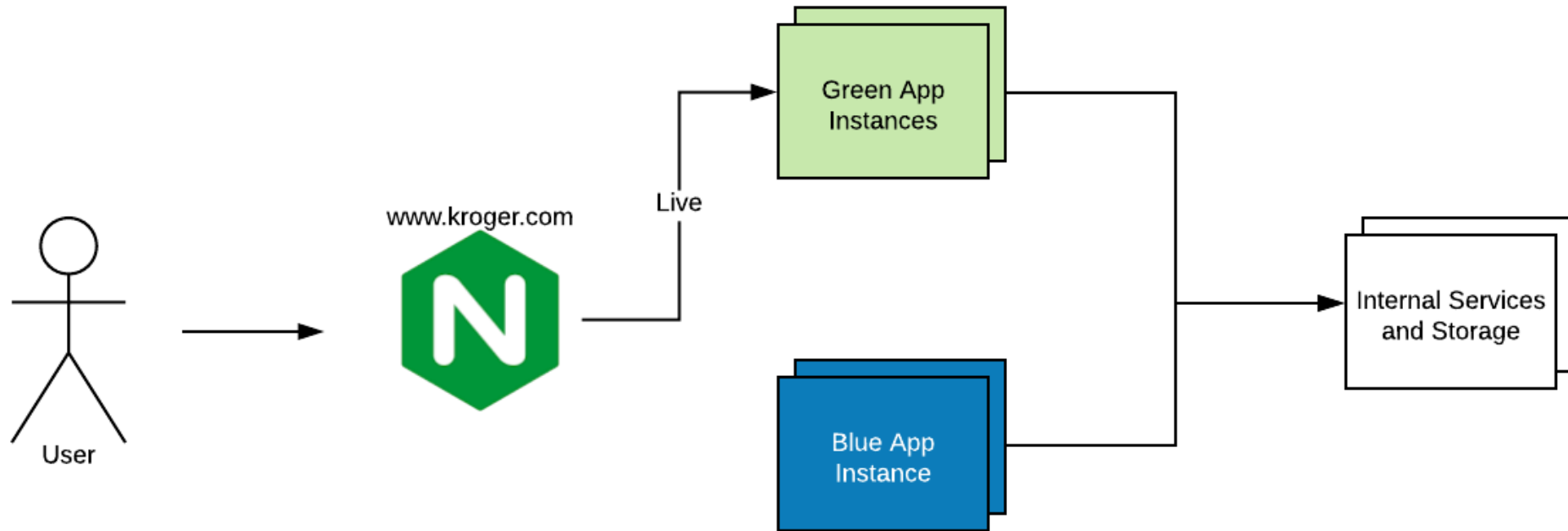
- Kroger
- Google (Kubernetes ingress)
- Stack Overflow
- Netflix
- Spotify
- AirBnB
- Instagram
- Shopify
- Instacart
- Digital Ocean
- Reddit
- Yelp
- Quora
- Asana
- Foursquare
- Coinbase
- CircleCI
- Intuit
- Pandora
- TripAdvisor
- Sony
- EverNote
- Pintrest
- Bloomberg
- Webmd
- Apple

... like everybody.

Blue/Green Deploy

- Two Groups: Blue and Green
- Live to one color.
- New Version to other color.
- Change routing to new version

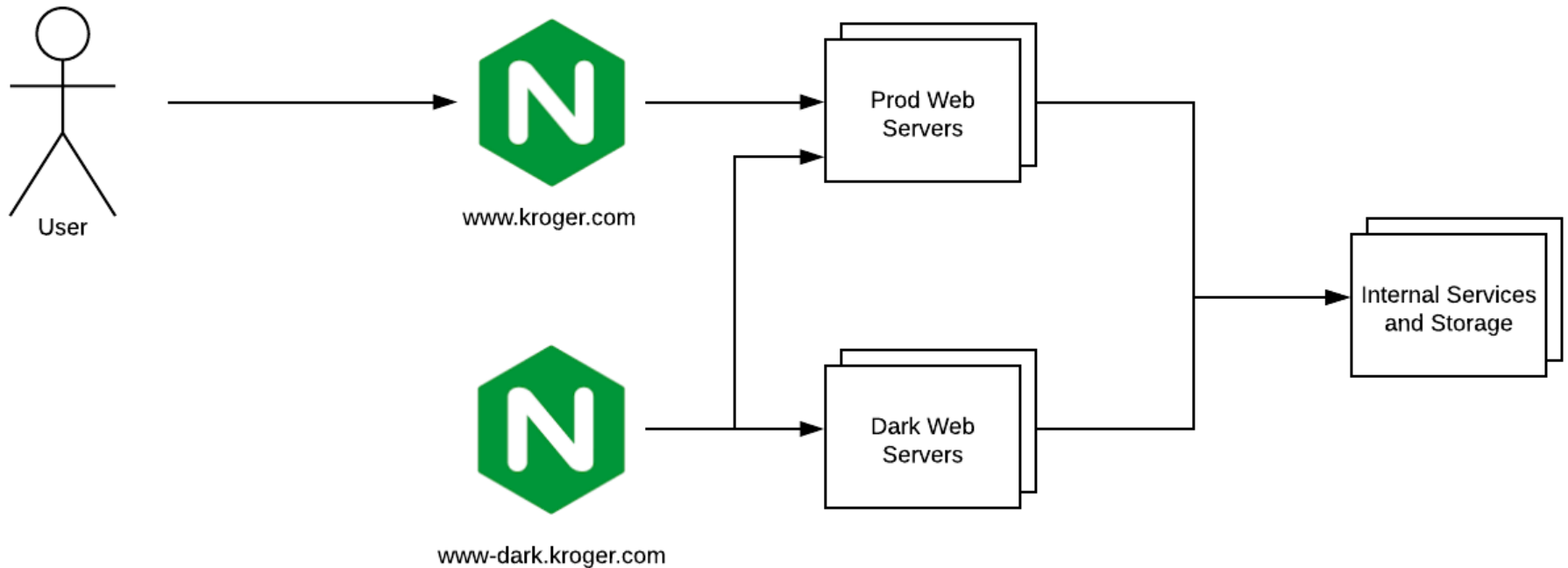
Blue/Green Diagram



Dark Deployment

- Not sure if other people do this.
- Used on the web tier.
- Create a separate group of servers
- Servers are configured to run as prod.
- Only accessible internally.
- We use <https://www-dark.kroger.com>

Dark Diagram Setup



End Act 2

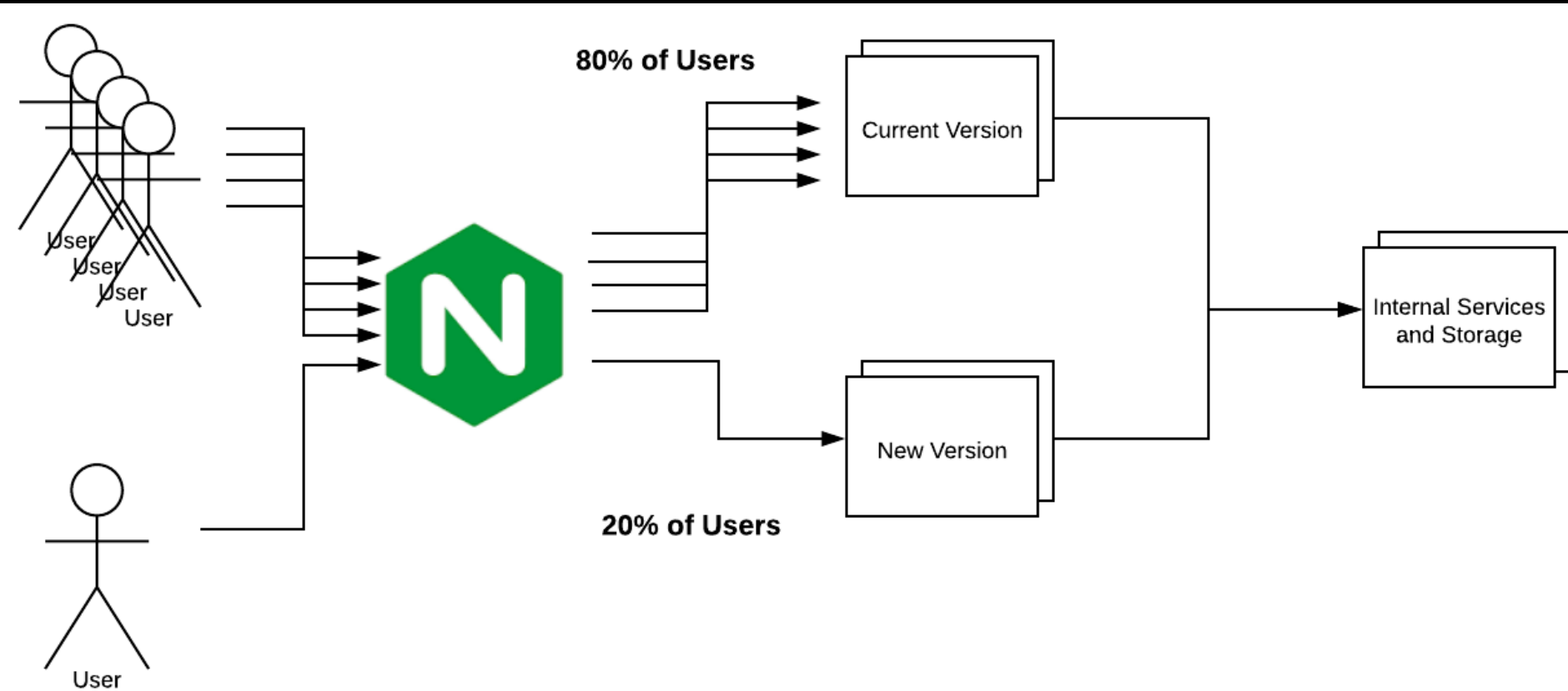
Production Feedback

- Coined as “QA in Prod”
- Expand focus to Production
- Monitor
 - Logs
 - Metrics
 - KPIs
 - Customer usage
- Use alerts
- Create dashboards
- Kroger uses an internal Pipeline
 - Gathers over 1.2TB data per day.
 - 10 second lag off real-time

Incremental Deployments

- Common or Closely related terms
 - Red/Black Deployments
 - Rolling Deployments
 - Canary Deployments
 - A/B deployments.
- A percentage of users are set to use new version.
- Users are incrementally moved.
- 100% traffic to new app is deployment complete

Incremental Deployments



So Why Metrics?

- How do you know the new version is working?
- Start manually.
- Decide on objective analysis
- Automate it.
- Don't forget rollback.

A Mature Example

- Spinnaker.io
- OSS by Netflix
- Deployment Types:
 - Blue/Green
 - Rolling Red/Black
 - Rolling Canary
- Providers:
 - GCP App Engine
 - AWS
 - Azure
 - DC/OS
 - GCP Conainter Engine
 - Kubernetes
 - Openstack
 - Oracle
 - Cloud Foundry (Maybe)

Closing Thoughts

- Make every commit ready for production
- Make deployments small
- Automate
- Practice as much as you can
- Measure

Questions?

Stephen Shary

stephen.shary@gmail.com

<https://www.linkedin.com/in/stephenshary/>

Please provide feedback: rate the talk