DEVOPS ON DAY 1

WITH OPERATIONS FIRST DELIVERY

Lean TECHniques

LEAN/AGILE/XP/DEVOPS COACH

FOUNDER OF AGILE IOWA

FAILED ENTREPRENEUR

SUMMER OF TIM (2001)

FUTURE SKYDIVER

TIM GIFFORD

a NETFLIX

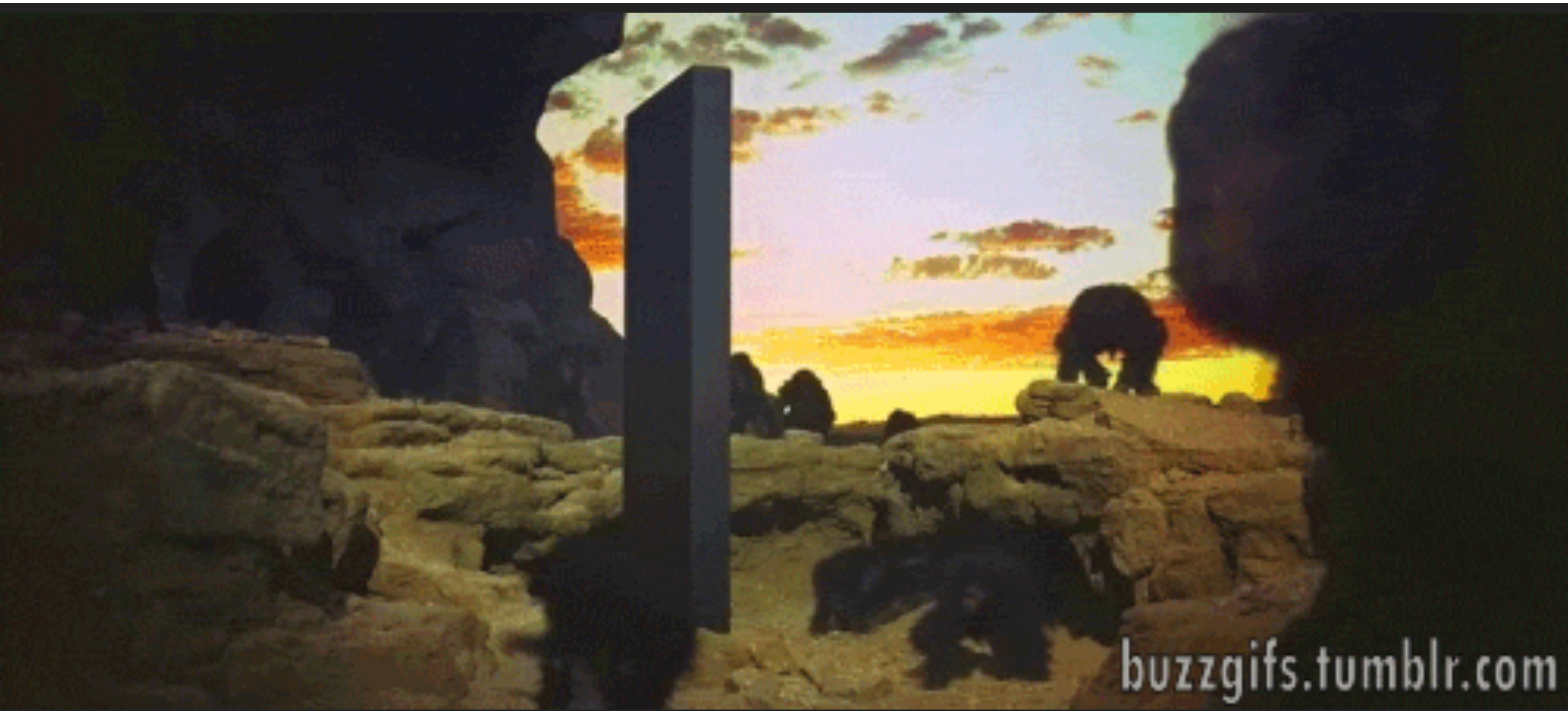
Etsy Google

UNICORN THEISM

MYTH

A WIDELY HELD BUT FALSE BELIEF OR IDEA

BUZZWORDS ARE MYTHS



MICROSERVICES

DEVOPS

LONG WINTER FARM
lip balm
UNICORN FARTS
Net Wt .15 ounces

THE CLOUD

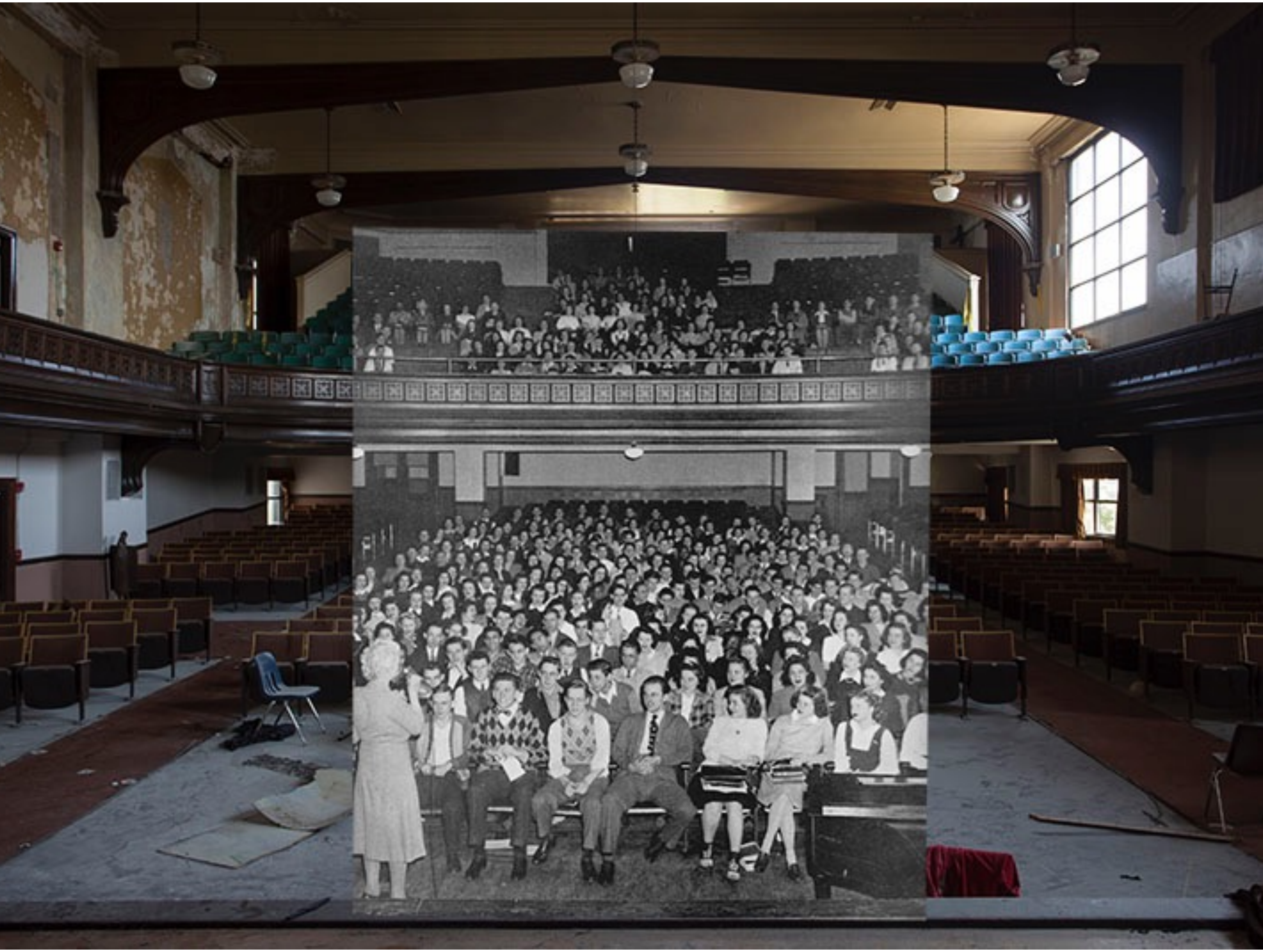
BEHAVIORS BASED ON MYTHS

SLOW == SAFE

PLAN OR BUSINESS VALUE

ORGANIZATIONS NEED TO DECIDE WHETHER THEIR PRIMARY OBJECTIVE IS TO DELIVER LONG-TERM ACCURATE PLANS TO ITS EXECUTIVES
OR
IF IT IS TO DELIVER BUSINESS VALUE TO ITS CUSTOMERS.

Gruver, Gary; Mouser, Tommy
Leading the Transformation: Applying Agile and DevOps Principles at Scale

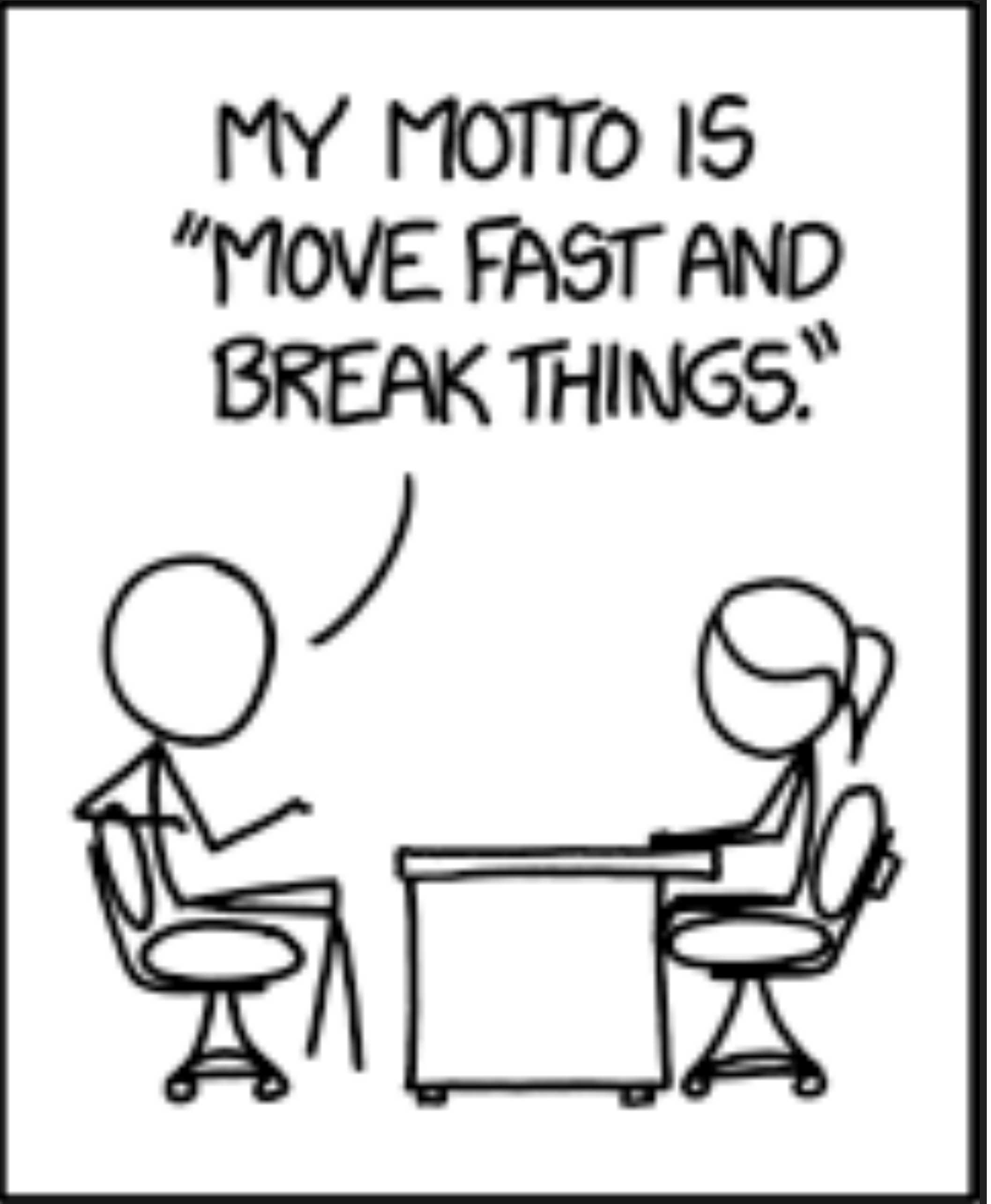


WARNING!

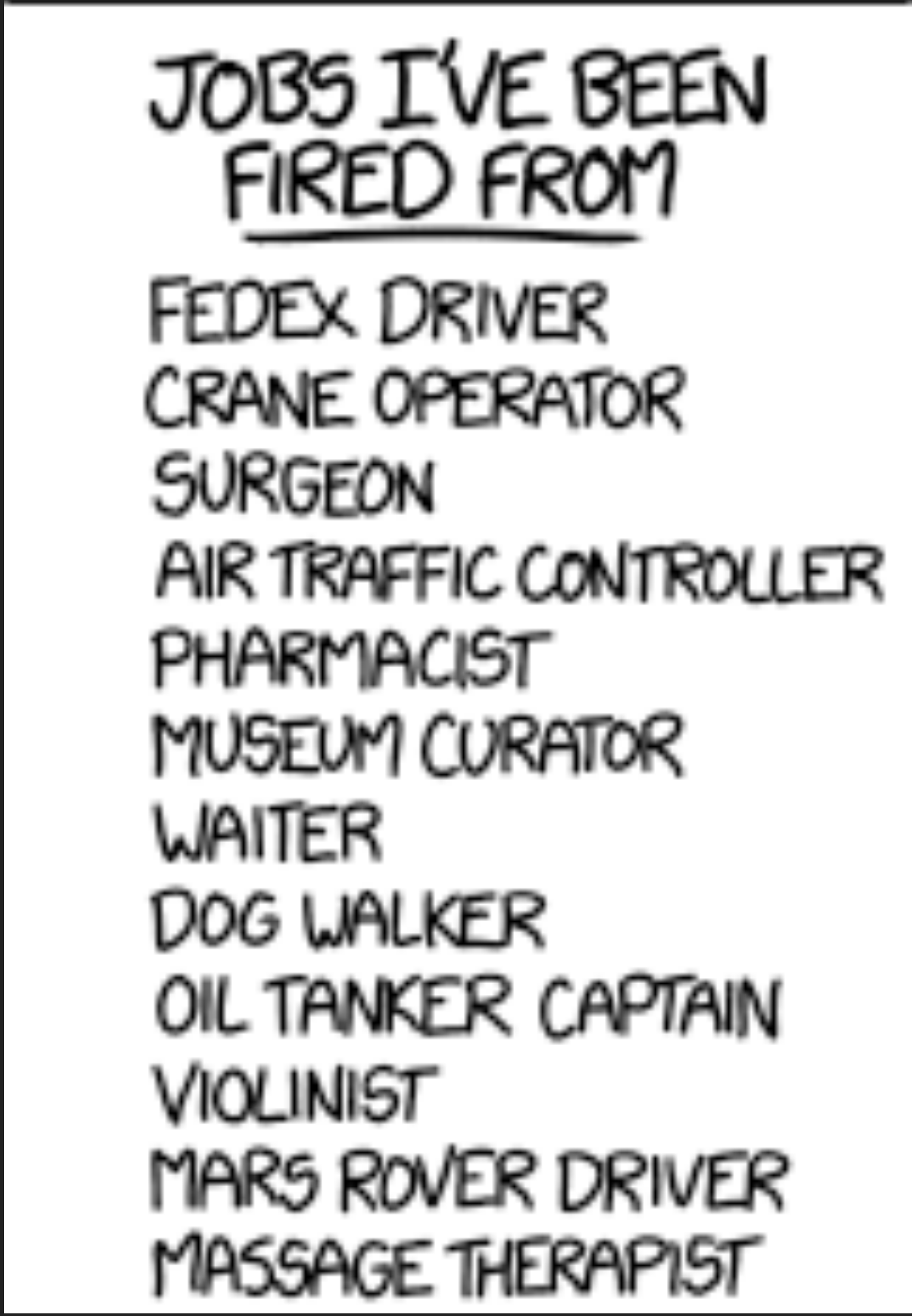
DETROIT OF FINANCIAL SERVICES

FAST == SAFE

MOVE
FAST AND
BREAK
THINGS



https://xkcd.com/1428/



FAST && SAFE

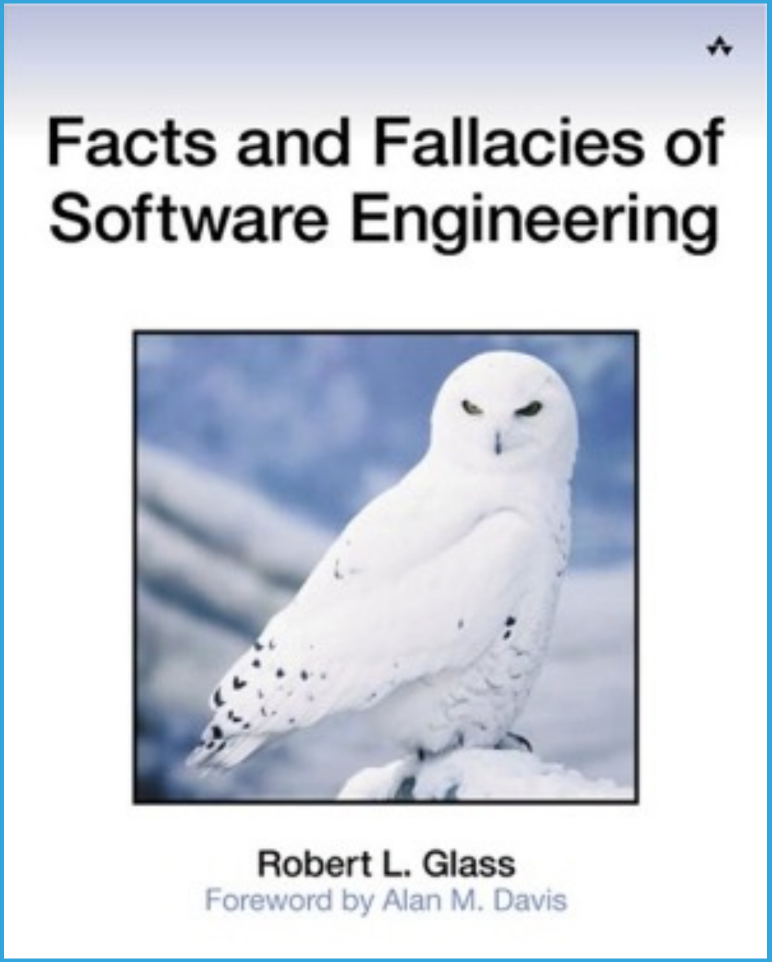
MOVE FAST WITH STABLE INFRA

The traditional model is that you take your software to the wall that separates development and operations, and throw it over and then forget about it. Not at Amazon.

You build it, you run it.

This brings developers into contact with the day-to-day operation of their software. It also brings them into day-to-day contact with the customer. This customer feedback loop is essential for improving the quality of the service.

Werner Vogels, Amazon CTO (2006)


Facts and Fallacies of Software Engineering

Robert L. Glass
Foreword by Alan M. Davis

MAINTENANCE TYPICALLY CONSUMES **40 TO 80 PERCENT** OF SOFTWARE COSTS. IT IS PROBABLY THE MOST IMPORTANT SOFTWARE LIFECYCLE PHASE.

Robert L. Glass

FAST

&&

SAFE

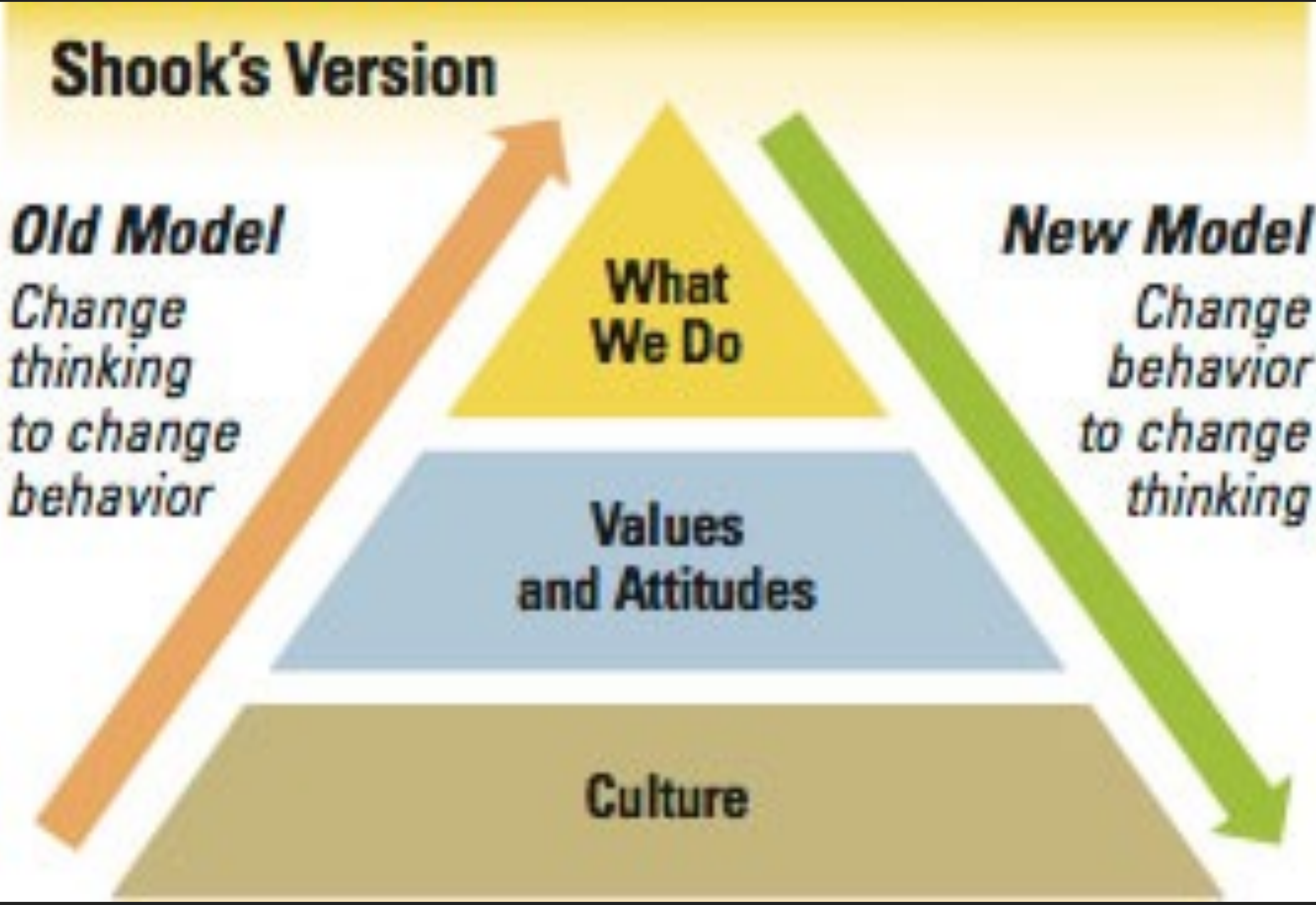
&&

SECURE

DEVOPSEC ™

KNOWLEDGE DOESN'T CHANGE BEHAVIOR

TELL ME WHAT TO DO…
I'LL TELL YOU WHY I CAN'T


Shook's Version
Old Model
Change thinking to change behavior
What We Do
Values and Attitudes
Culture
New Model
Change behavior to change thinking

IT'S NOT KNOWING WHAT TO DO; IT'S DOING WHAT YOU KNOW.

Tony Robbins

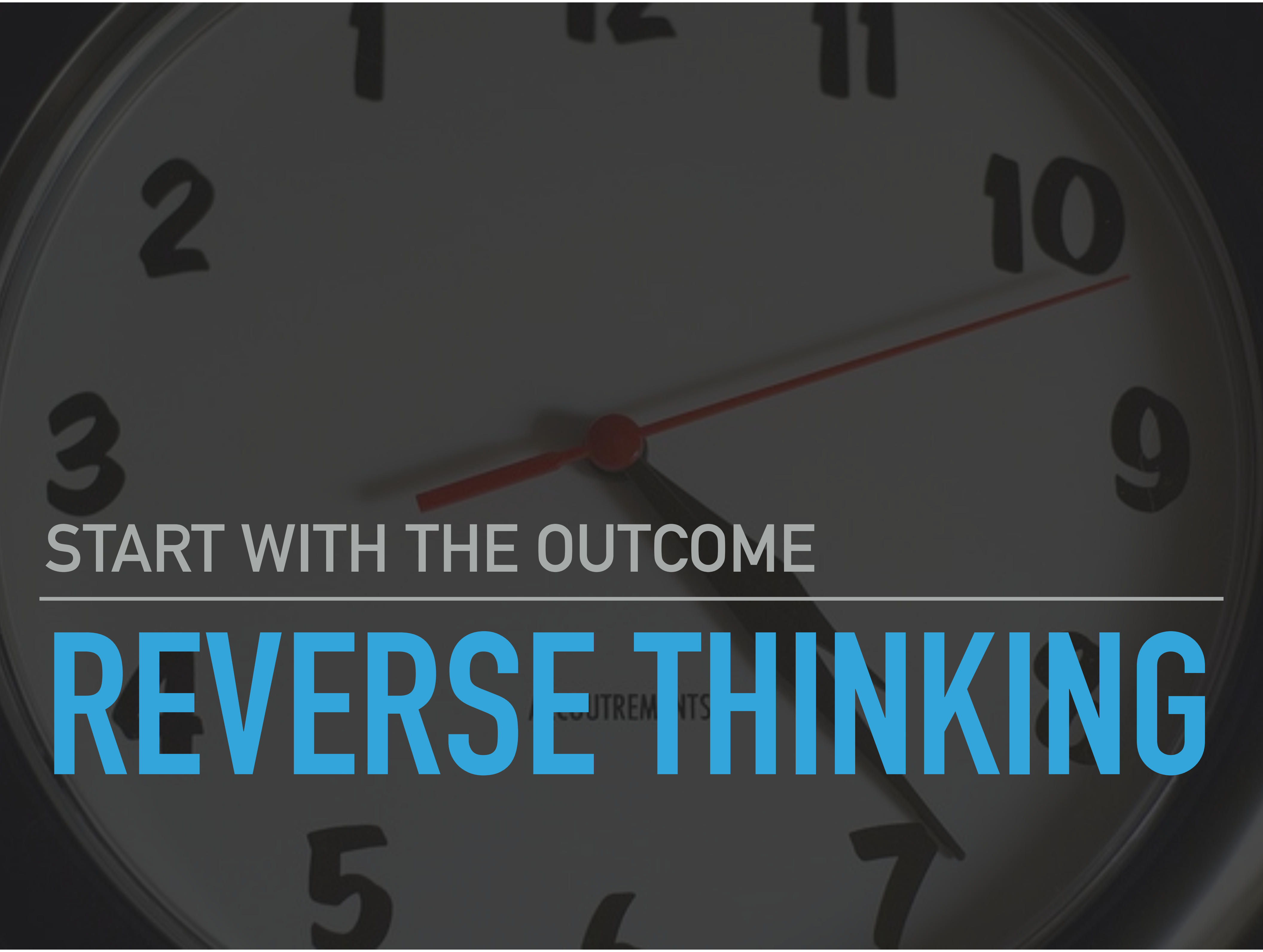
OPERATIONS FIRST DELIVERY

Start with PRODUCTION and work backwards!

DAY 1:

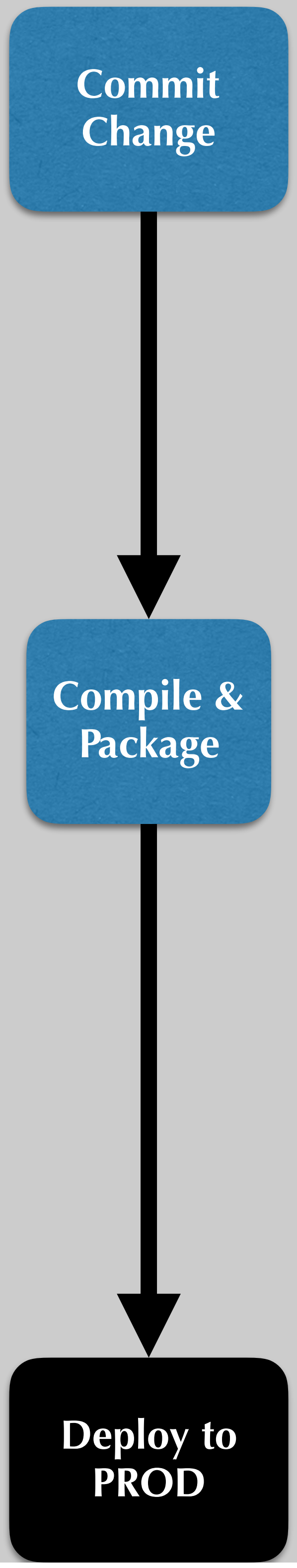
EVERY VALIDATED CHANGE DEPLOYED TO PRODUCTION

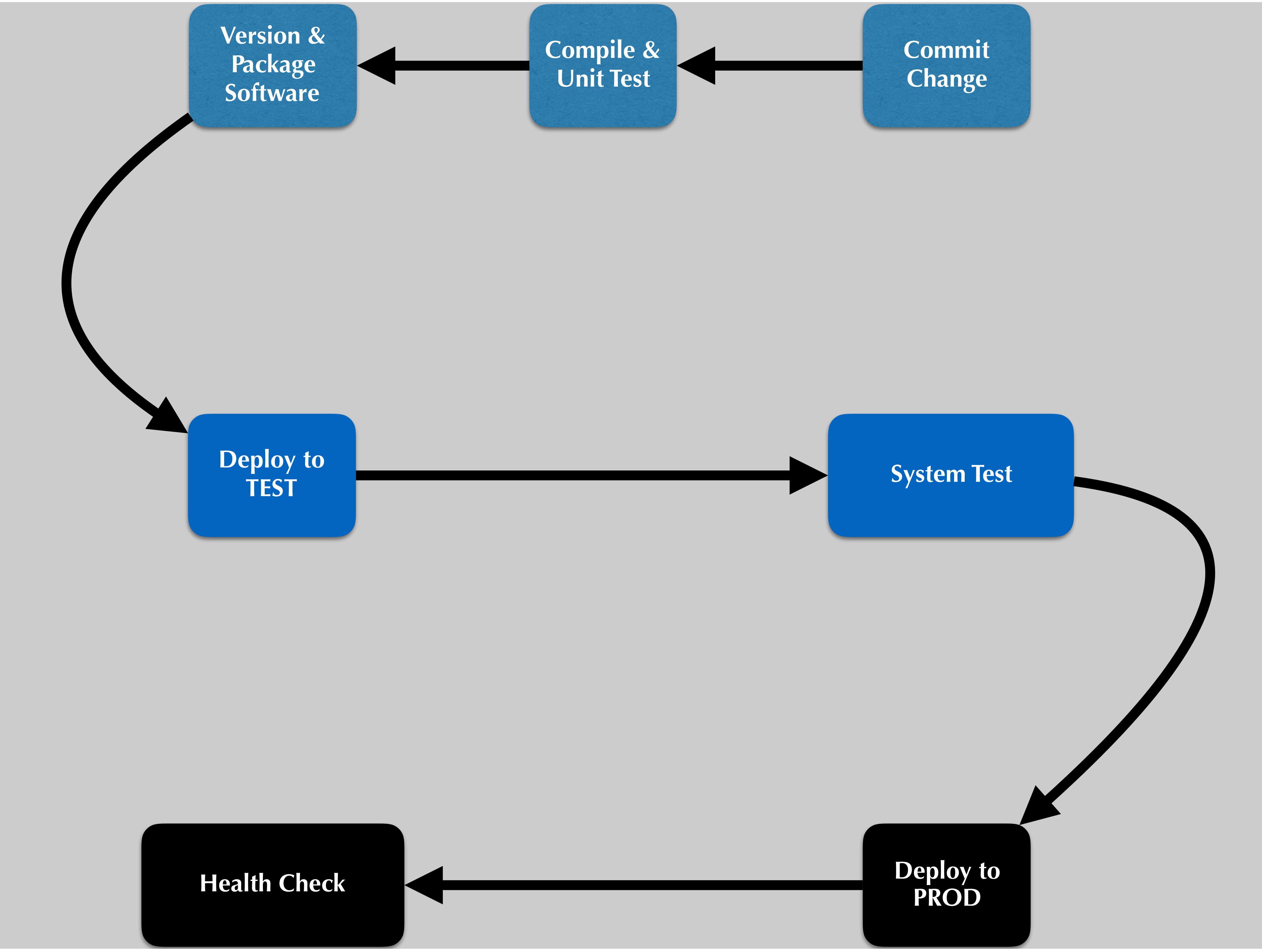
NO USERS == NO RISK

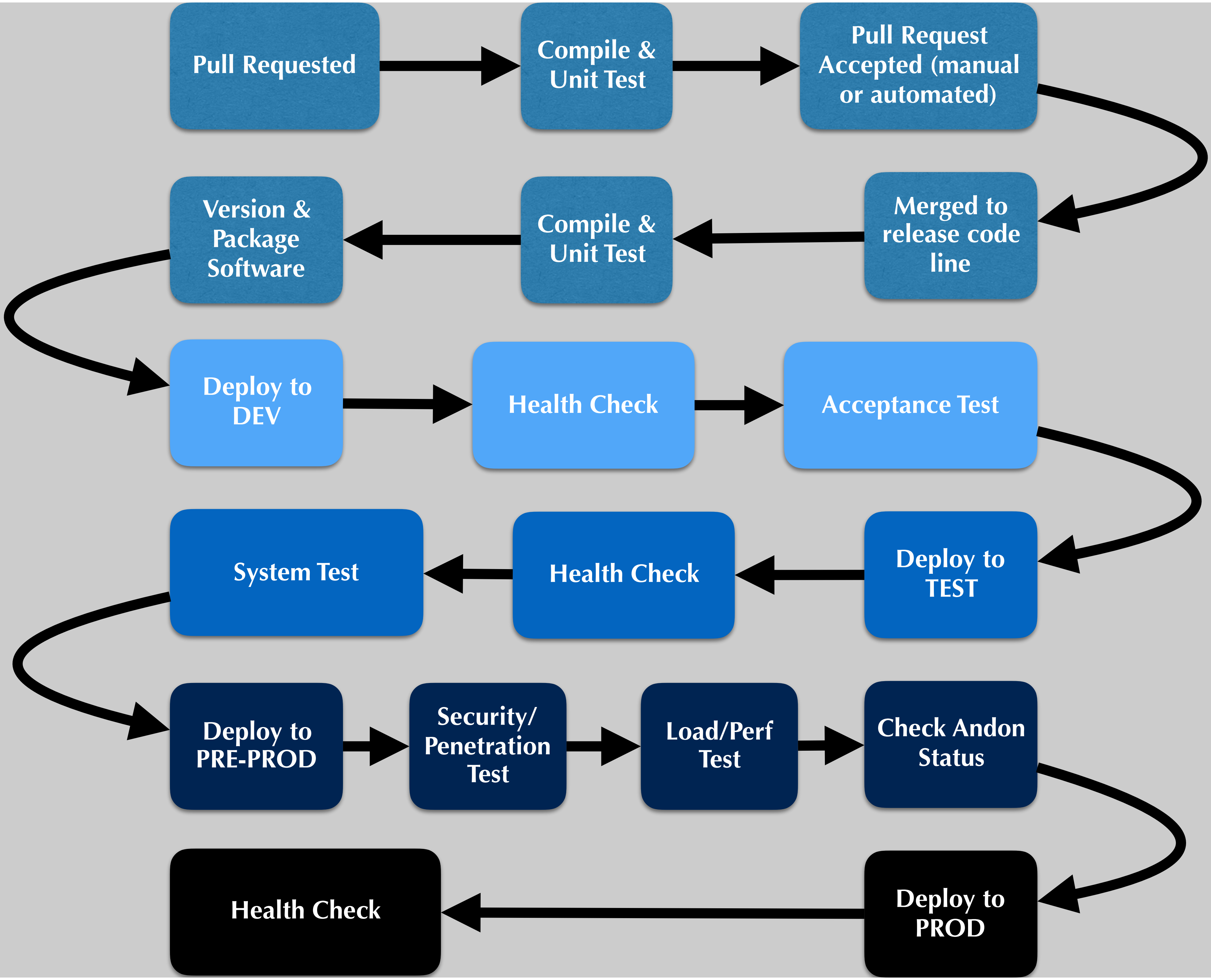


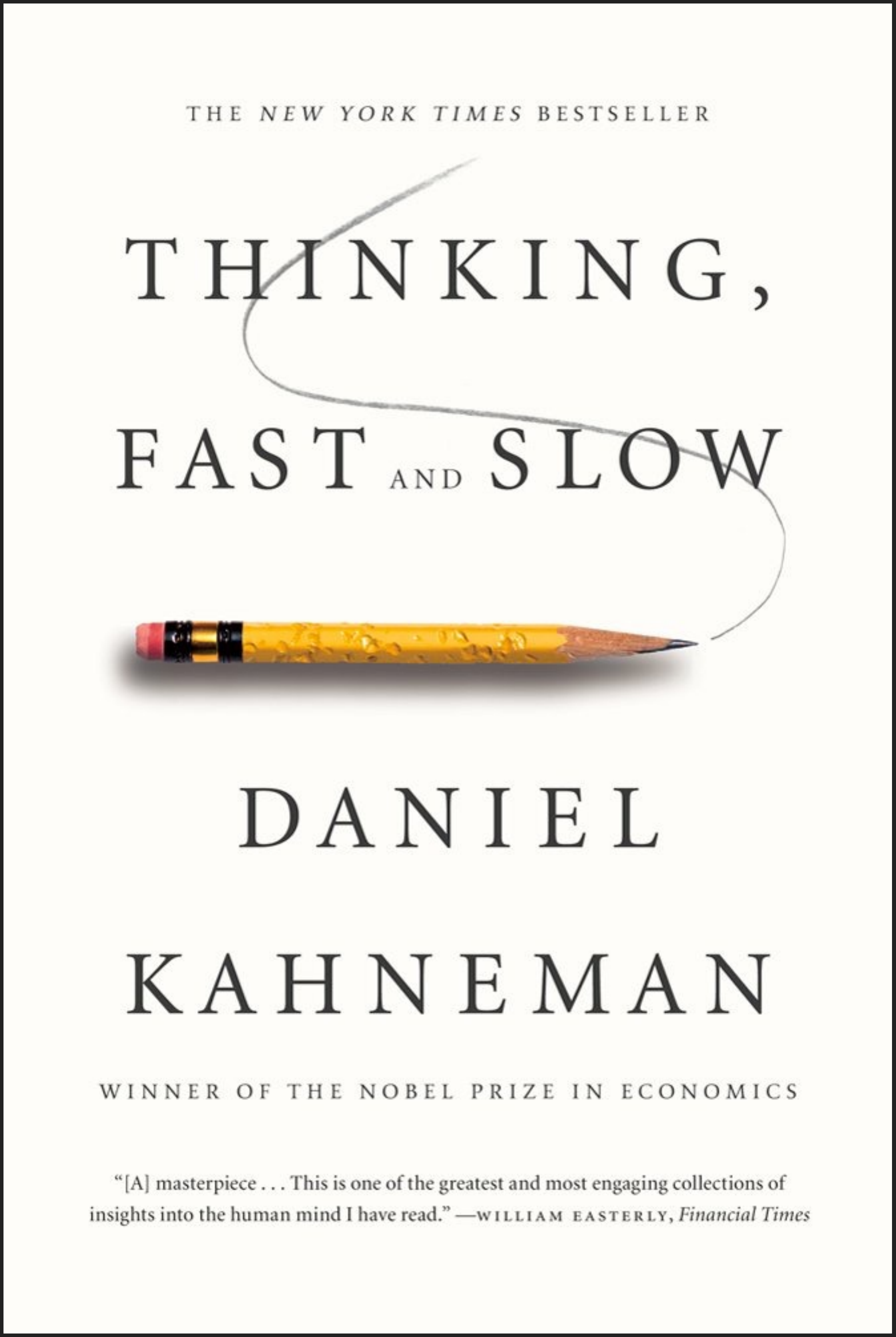
START WITH THE OUTCOME

REVERSE THINKING


Commit Change
Compile & Package
Deploy to PROD


Commit Change
Compile & Unit Test
Version & Package Software
Deploy to TEST
System Test
Deploy to PROD
Health Check


Pull Requested
Compile & Unit Test
Pull Request Accepted (manual or automated)
Merged to release code line
Compile & Unit Test
Version & Package Software
Deploy to DEV
Health Check
Acceptance Test
Deploy to TEST
Health Check
System Test
Deploy to PRE-PROD
Security/ Penetration Test
Load/Perf Test
Check Andon Status
Deploy to PROD
Health Check


THE *NEW YORK TIMES* BESTSELLER

THINKING, FAST AND SLOW

DANIEL KAHNEMAN

WINNER OF THE NOBEL PRIZE IN ECONOMICS

"[A] masterpiece . . . This is one of the greatest and most engaging collections of insights into the human mind I have read." —WILLIAM EASTERLY, *Financial Times*

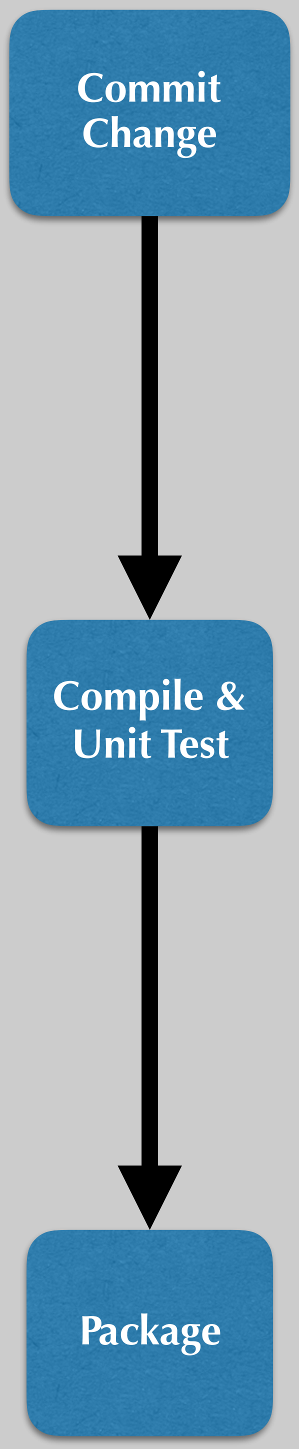
LOSSES ARE TWICE AS POWERFUL

LOSS AVERSION

TOOLS & TECHNIQUES

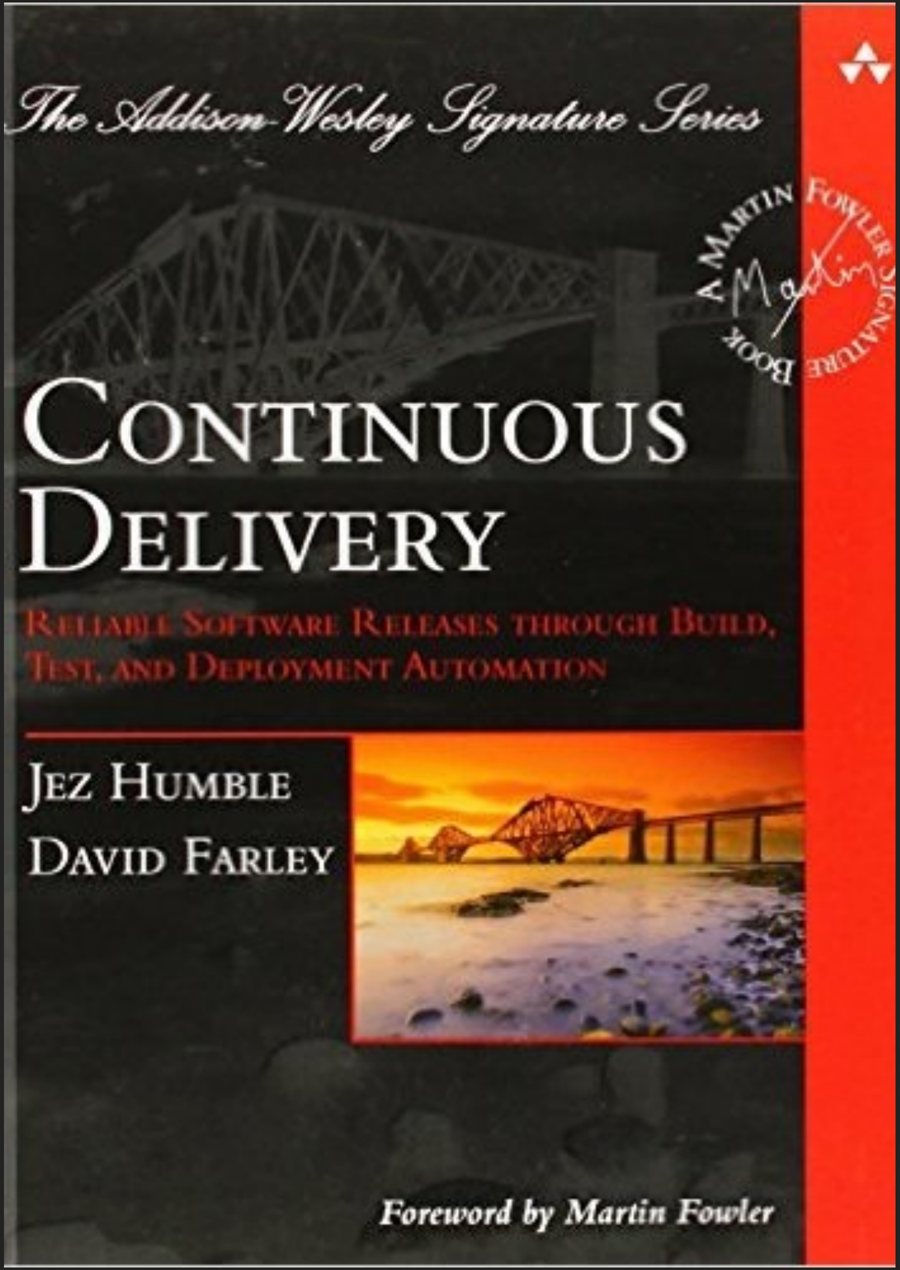
- CONTINUOUS INTEGRATION => CONTINUOUS DELIVERY
- ARCHITECTURE => 12 FACTOR APPS
- APP VERSIONING => SYSTEM VERSIONING
- RELEASES => DEPLOY WITH FEATURE TOGGLES
- SOFTWARE ENGINEERING => RESILIENCE ENGINEERING

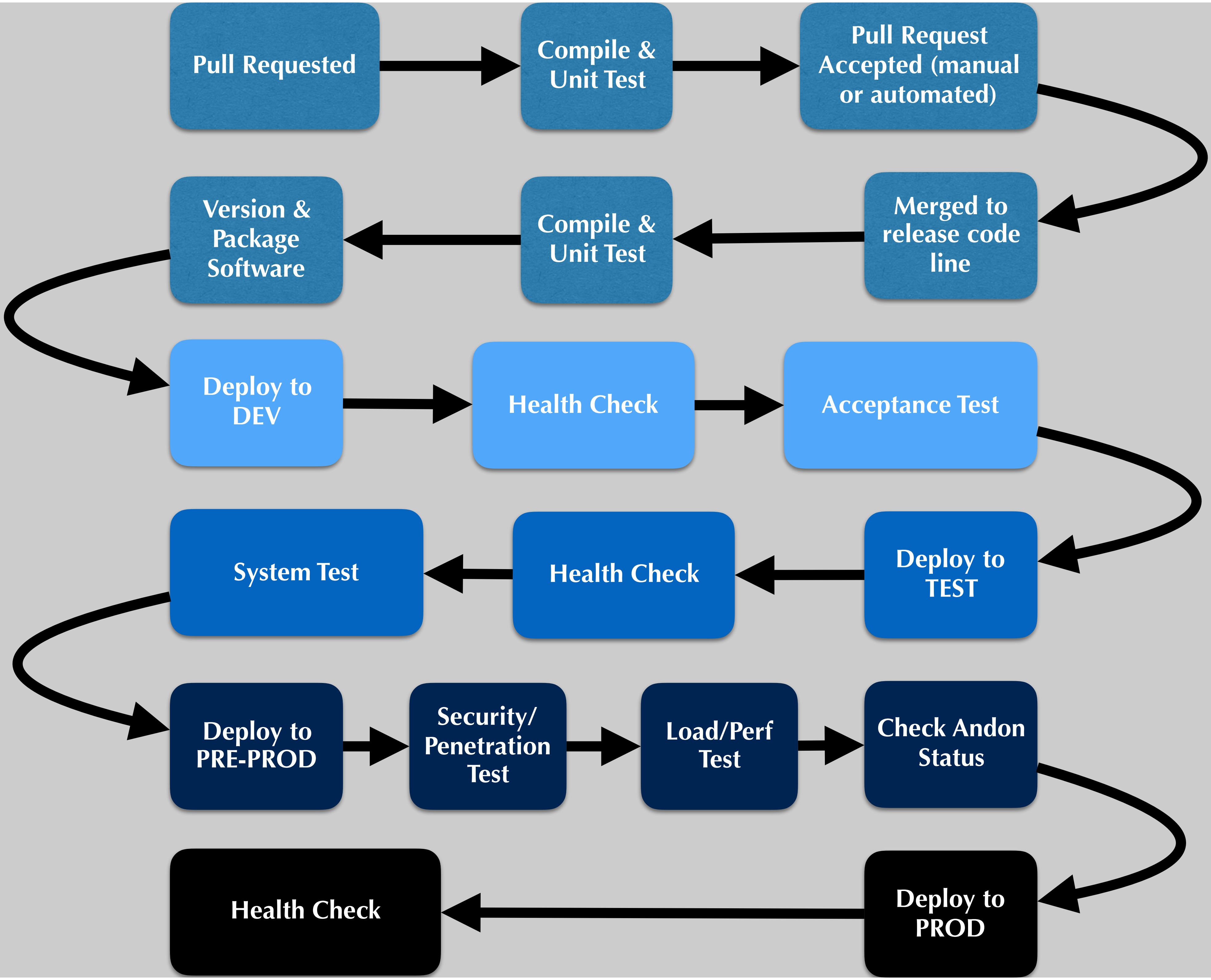
CONTINUOUS INTEGRATION


Commit Change
Compile & Unit Test
Package

CONTINUOUS DELIVERY

- Deployment Pipeline
- Every change set is a release candidate
- Integrate FIRST


The Addison-Wesley Signature Series
A Martin Fowler Signature Book
Continuous Delivery
Reliable Software Releases through Build, Test, and Deployment Automation
Jez Humble
David Farley
Foreword by Martin Fowler


Pull Requested
Compile & Unit Test
Pull Request Accepted (manual or automated)
Merged to release code line
Compile & Unit Test
Version & Package Software
Deploy to DEV
Health Check
Acceptance Test
Deploy to TEST
Health Check
System Test
Deploy to PRE-PROD
Security/ Penetration Test
Load/Perf Test
Check Andon Status
Deploy to PROD
Health Check

The Twelve Factors

I. Codebase
One codebase tracked in revision control, many deploys

II. Dependencies
Explicitly declare and isolate dependencies

III. Config
Store config in the environment

IV. Backing Services
Treat backing services as attached resources

V. Build, release, run
Strictly separate build and run stages

VI. Processes
Execute the app as one or more stateless processes

VII. Port binding
Export services via port binding

VIII. Concurrency
Scale out via the process model

IX. Disposability
Maximize robustness with fast startup and graceful shutdown

X. Dev/prod parity
Keep development, staging, and production as similar as possible

XI. Logs
Treat logs as event streams

XII. Admin processes
Run admin/management tasks as one-off processes

http://12factor.net/

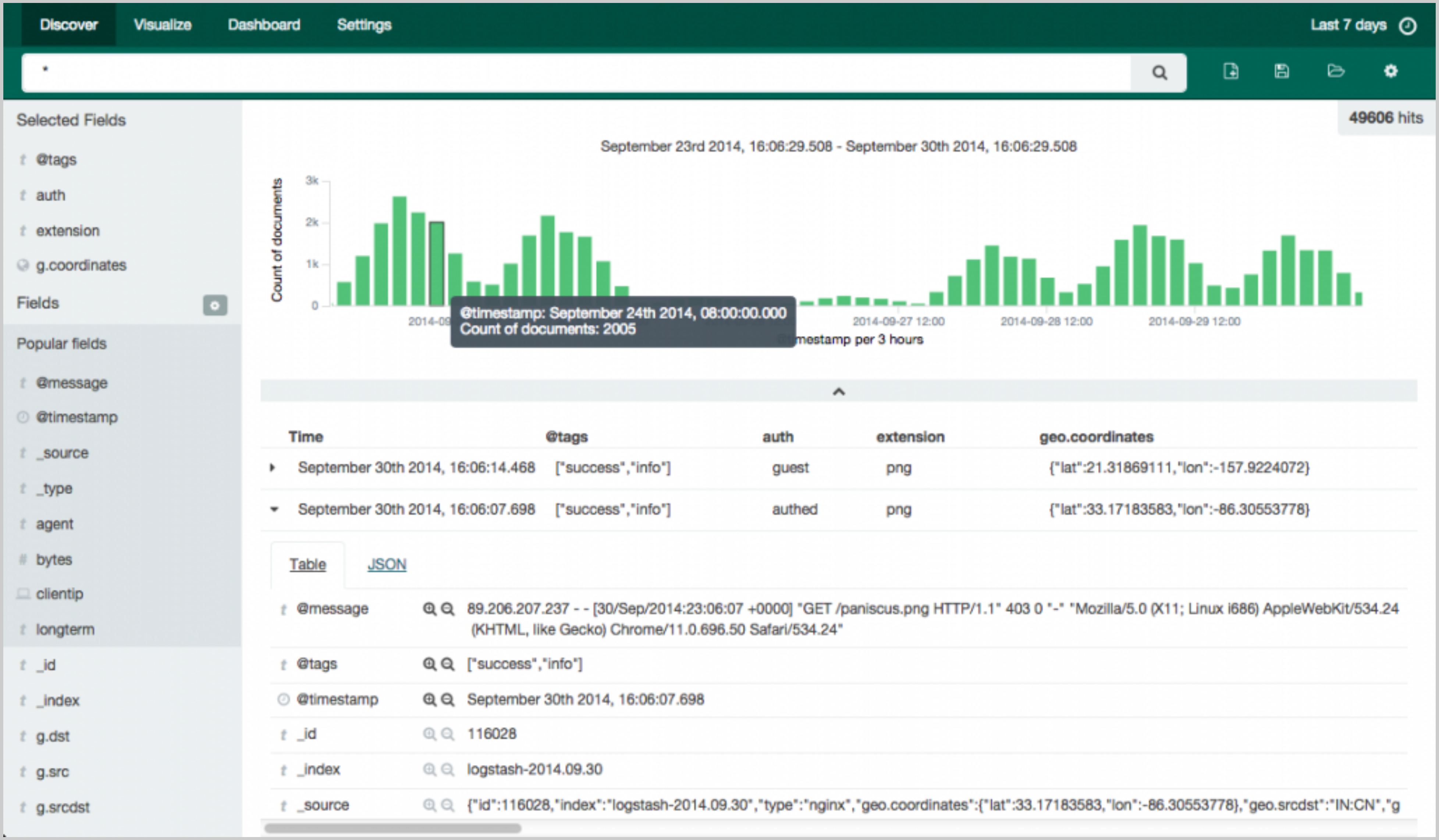
10. DEV/PROD PARITY

	Traditional App	12 Factor App
Time between deploys	Weeks	Hours
Code author vs code deployers	Different	Same
Dev vs prod environment	Divergent	Similar

11. LOGS

- Time ordered Event Stream
- App not responsible for routing or storage
- Aggregate across all backing service

LOGS - ELK STACK - ELASTICSEARCH/LOGSTASH/KIBANA



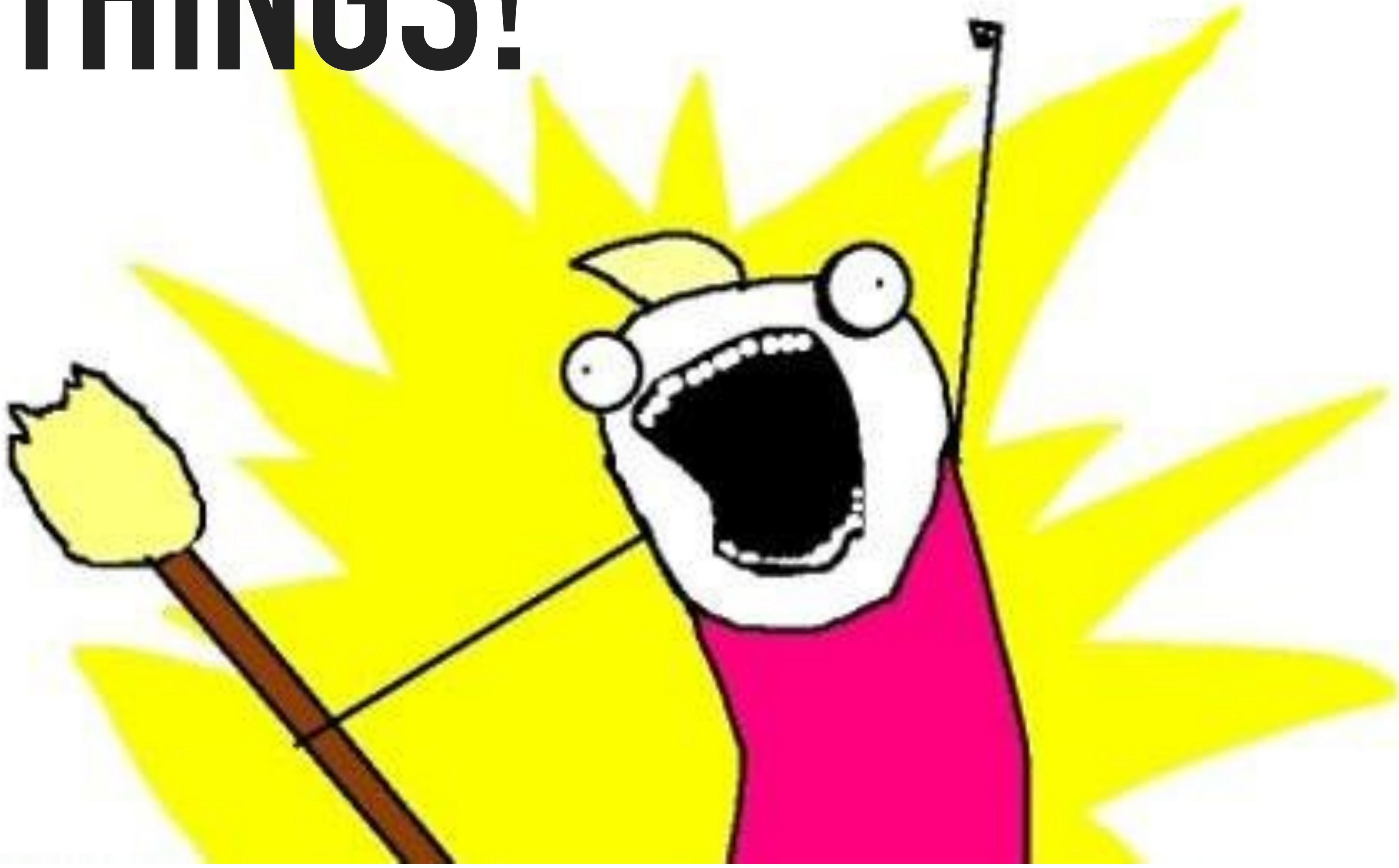
Time	@tags	auth	extension	geo.coordinates
▸ September 30th 2014, 16:06:14.468	["success","info"]	guest	png	{"lat":21.31869111,"lon":-157.9224072}
▾ September 30th 2014, 16:06:07.698	["success","info"]	authed	png	{"lat":33.17183583,"lon":-86.30553778}

@message	89.206.207.237 - - [30/Sep/2014:23:06:07 +0000] "GET /paniscus.png HTTP/1.1" 403 0 "-" "Mozilla/5.0 (X11; Linux i686) AppleWebKit/534.24 (KHTML, like Gecko) Chrome/11.0.696.50 Safari/534.24"
@tags	["success","info"]
@timestamp	September 30th 2014, 16:06:07.698
_id	116028
_index	logstash-2014.09.30
_source	{"id":116028,"index":"logstash-2014.09.30","type":"nginx","geo.coordinates":{"lat":33.17183583,"lon":-86.30553778},"geo.srcdst":"IN:CN","g

12+1. MONITORING

- Collecting
  - Collectd (CPU, Memory, NIC)
  - Coda Hale (App, Biz)
- Routing/Storage
- Visualizing
- Alerting

VERSION ALL THE THINGS!



WHAT VERSION IS YOUR...

CODE?

WHAT VERSION ARE YOUR...

DEPENDENCIES?

```
"dependencies": {
  "chalk": "^1.0.0",
  "date-time": "^1.0.0",
  "figures": "^1.0.0",
  "hooker": "^0.2.3",
  "pretty-ms": "^1.0.0",
  "text-table": "~0.2.0"
},
```

WHAT VERSION IS YOUR...

DATABASE?

WHAT VERSION IS YOUR…

INFRASTRUCTURE?

IMMUTABLE INFRASTRUCURE

Randomize and discard admin credentials

WHAT VERSION IS YOUR…

SECURITY POLICY?

WHAT VERSION ARE YOUR…

FEATURES?

SEMANTIC VERSIONING

- Given a version MAJOR.MINOR.PATCH, increment the:
  - MAJOR when you make **incompatible** API changes
  - MINOR when you **add functionality** in a backwards-compatible manner
  - PATCH when you make backwards compatible **bug fixes**

DON'T USE SEMANTIC VERSIONING...

FOR CONTINUOUS DEPLOYED APPLICATIONS

Tim Gifford (2015)

NEVER QUOTE YOURSELF.

Tim Gifford

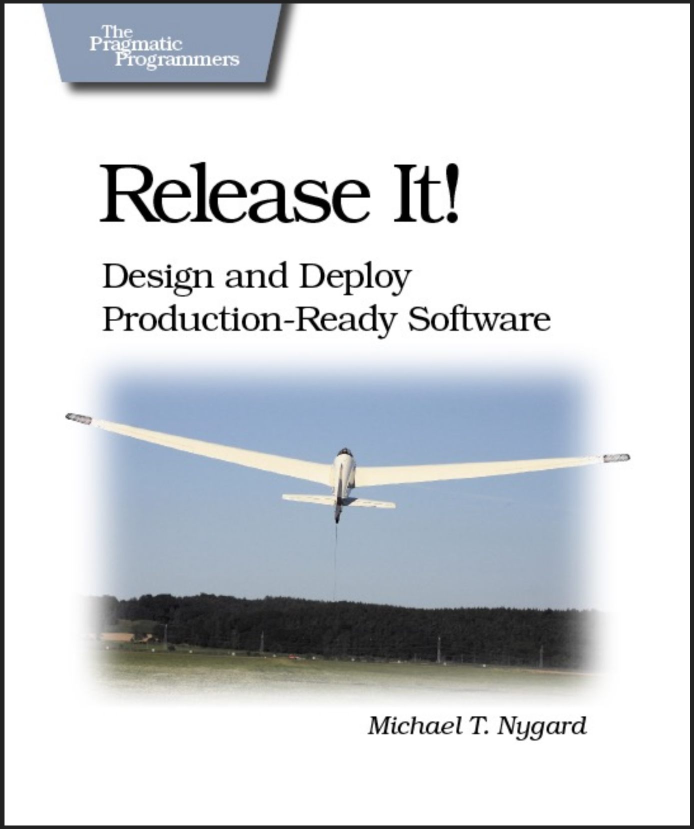
FEATURE TOGGLES

YO DAWG I HEARD YOU LIKE FEATURE TOGGLES
SO I GOT A FEATURE TOGGLE FOR YOUR FEATURE TOGGLE
memegenerator.net

YOU FELL FOR ONE OF THE CLASSIC BLUNDERS
NEVER ADD A FEATURE TOGGLE UNLESS YOU PLAN TO REMOVE IT!

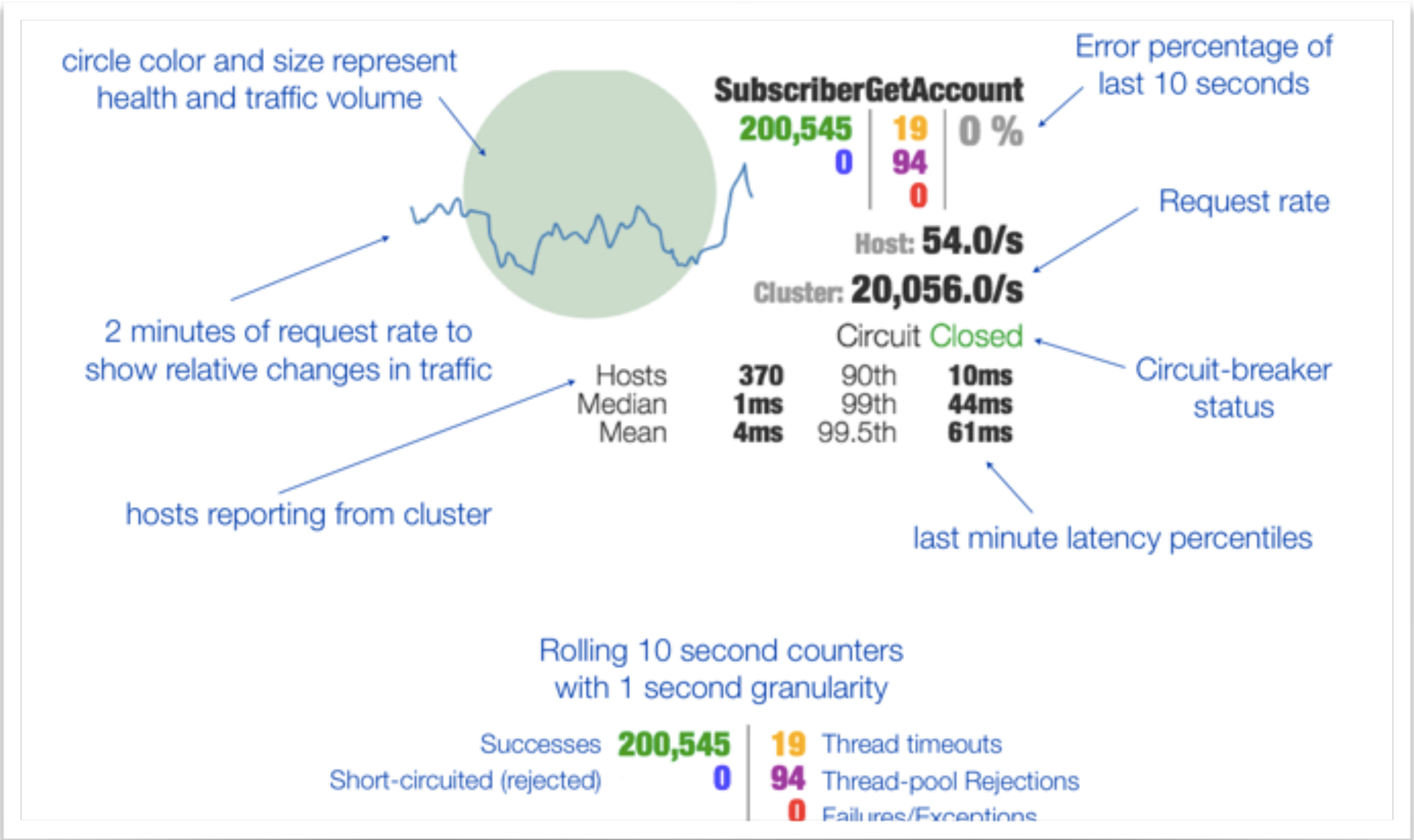
RESILIENCE ENGINEERING

- Timeouts
- Handshakes
- Circuit Breaker
  - Hystrix (Netflix)


The Pragmatic Programmers
Release It!
Design and Deploy Production-Ready Software
Michael T. Nygard

BACK IN MY DAY
A MANS WORD & A HANDSHAKE WERE AS GOOD AS A CONTRACT
imgflip.com

HYSTRIX DASHBOARD


circle color and size represent health and traffic volume
SubscriberGetAccount
200,545
0
19
94
0
0 %
Error percentage of last 10 seconds
Host: 54.0/s
Cluster: 20,056.0/s
Request rate
Circuit Closed
Circuit-breaker status
2 minutes of request rate to show relative changes in traffic
Hosts 370
Median 1ms
Mean 4ms
90th 10ms
99th 44ms
99.5th 61ms
hosts reporting from cluster
last minute latency percentiles
Rolling 10 second counters with 1 second granularity
Successes 200,545
Short-circuited (rejected) 0
19 Thread timeouts
94 Thread-pool Rejections
0 Failures/Exceptions

WHEN CALLING THIRD PARTIES, SERVICE LEVELS ONLY DECREASE.

Michael Nygard

59%

90%

90%

90%

90%

90%

SCIENTIST

```
Experiment<Integer> e = new Experiment("foo");
e.run(this::controlFunction, this::candidateFunction);
```

REVIEW

- Behaviors based on beliefs
- Change actions to change behaviors
- Start with the customer and work backwards
- Leverage Loss Aversion
- Tools & Techniques

- Final Thoughts….

PERFORMANCE METRICS

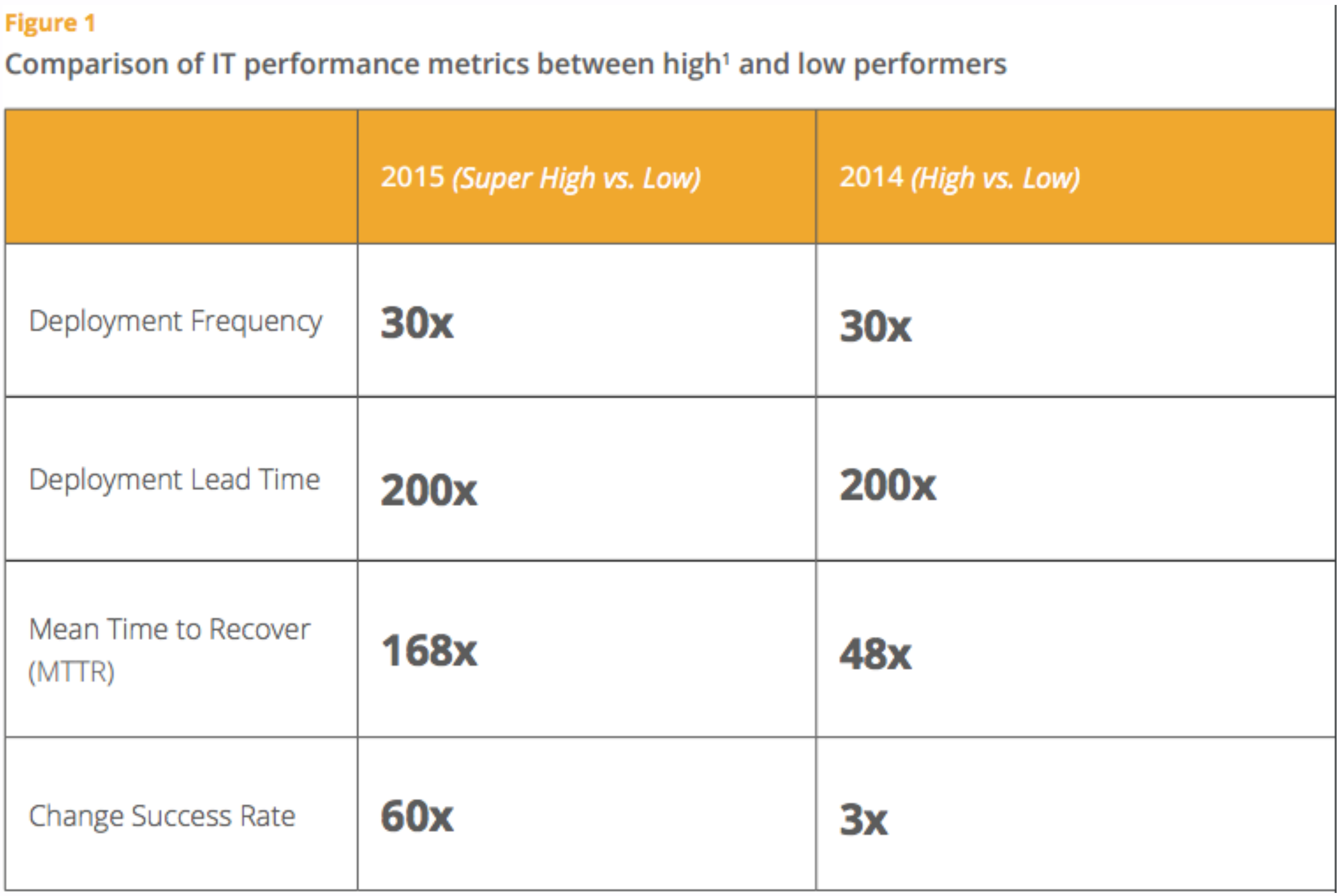


Figure 1

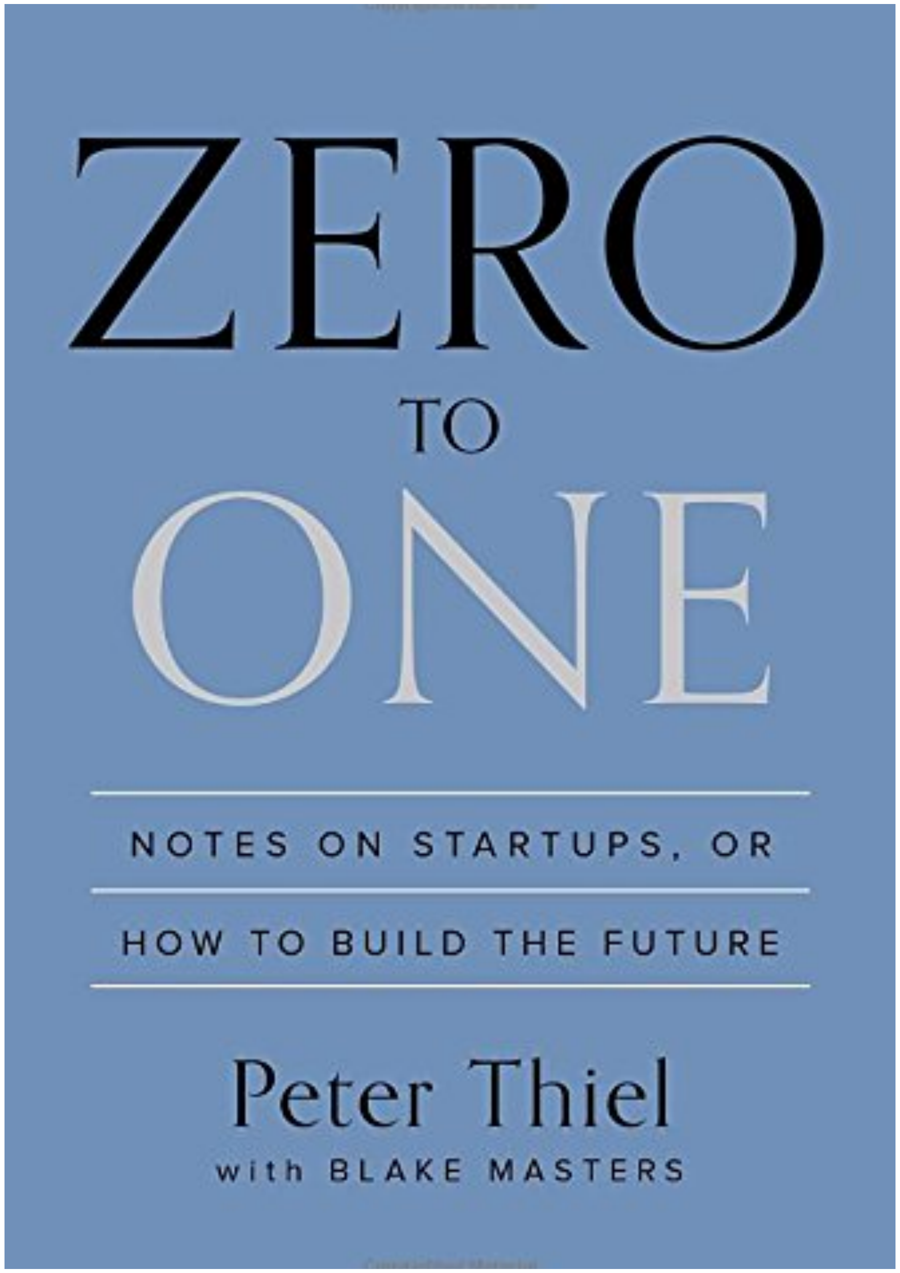
Comparison of IT performance metrics between high[1] and low performers

	2015 *(Super High vs. Low)*	2014 *(High vs. Low)*
Deployment Frequency	**30x**	**30x**
Deployment Lead Time	**200x**	**200x**
Mean Time to Recover (MTTR)	**168x**	**48x**
Change Success Rate	**60x**	**3x**

Puppet Labs 2015 State of DevOps Report (https://puppetlabs.com/2015-devops-report)

ZERO TO ONE

- Technology is going from nothing to something
- Create something new is the greatest challenge
- Current culture doesn't support big ideas (Cults, Space)
- 1 to N is globalization


ZERO TO ONE

NOTES ON STARTUPS, OR HOW TO BUILD THE FUTURE

Peter Thiel with BLAKE MASTERS

UNICORN CHALLENGE

OPERATIONS FIRST DELIVERY

QUESTIONS?

Lean TECHniques

TGIFFORD@LEANTECHNIQUES.CO

@TIMGIFFORD

HTTPS://WWW.LINKEDIN.COM/IN/TIMGIFFORD

THANKS