

Learning From Chaos

“Architecting for Resilience”

or ... “How to be Great at Being Wrong”

Russ Miles, CEO, ChaosIQ - @russmiles











What Chaos is *NOT*



Also ... a confession.

What is “*Wrong*”?

And why are we ***scared*** of it?

“not correct or true; incorrect.”

**“an injurious, unfair, or unjust
act”...**

...“action or conduct *inflicting harm without due provocation or just cause*”

“a *violation* or *invasion* of the
legal rights of another”

When are **we ever** wrong?

“the state of being mistaken or
incorrect”

A close-up shot of a male doctor with a beard and short dark hair, wearing a white lab coat over a blue and white striped shirt. He has a stethoscope around his neck. He is looking directly at the camera with a serious, slightly distressed expression. His hands are raised in front of him, palms facing forward, as if he is explaining something or gesturing during a conversation. The background is a blurred clinical setting with light-colored walls. The overall lighting is soft and professional.

I've got some bad news for you...

A woman in a dark dress with a white lace collar and a man in a white ruffled shirt and dark vest looking at each other in a room with wood paneling.

We're wrong all the time.

Why is wrong ***scary***?

Risk?

Consequences.

Why *us*?!

Two factors

Feature Velocity

Striving for Reliability

Feature Velocity **VS.** Reliability

Good news!

No conflict!

Feature Velocity **VS.** Reliability

Feature Velocity + Reliability

But ... *Microservices!*?

What about *tests*? *gates*?
pipelines? *isolation*?



We're covered...

A Story...







But business is easier...

*“One Hour of Downtime Costs >
\$100K For 95% of Enterprises”*

*“lost revenue and lost end user
productivity”*

“not take into account the cost of additional penalties for **regulatory non-compliance** or **“good will” gestures** made to the organization’s customers and business partners that were negatively impacted by a system or network failure. In fact, these two conditions can **cause downtime costs to skyrocket even further”**

Feature Velocity + Reliability

Can the problem
be designed out?

Complex

Probe
Sense
Respond

Emergent

Complicated

Sense
Analyze
Respond

Good Practice

Disorder

Chaotic

Act
Sense
Respond

Novel

Simple

Sense
Categorize
Respond

Best Practice

You are
NOT here



Complex

Probe
Sense
Respond

Emergent

Complicated

Sense
Analyze
Respond

Good Practice

Disorder

Chaotic

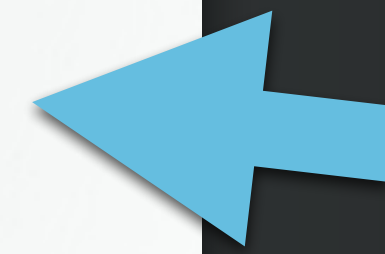
Act
Sense
Respond

Novel

Simple

Sense
Categorize
Respond

Best Practice



You are here?

Distributed Systems?
External Dependencies?

You are here?



Complex

Probe
Sense
Respond

Emergent

Complicated

Sense
Analyze
Respond

Good Practice

Disorder

Chaotic

Act
Sense
Respond

Novel

Simple

Sense
Categorize
Respond

Best Practice

Systems that evolve quickly?

Complex

Probe
Sense
Respond

Emergent

Complicated

Sense
Analyze
Respond

Good Practice

Disorder

Chaotic

Act
Sense
Respond

Novel

Simple

Sense
Categorize
Respond

Best Practice

You are
here.



Can't this be *prevented*?

Essential.

Dark Debt...

♥ Aaron Rinehart and sueallspaw liked



John Allspaw @allspaw · 4h

Reminder that "dark debt" *cannot* be anticipated or prevented. It is a natural byproduct of complexity.

I understand this is uncomfortable, especially for engineers. But that is the concept. If it can be prevented, it's not "dark debt" - it's something else.

Dark debt is found in complex systems and the anomalies it generates are complex system failures. Dark debt is not recognizable at the time of creation. Its impact is not to foil development but to generate anomalies. It arises from the unforeseen interactions of hardware or software with other parts of the framework. There is no specific countermeasure that can be used against dark debt because it is invisible until an anomaly reveals its presence.

💬 4

↻ 19

♥ 50



What about *tests*? *gates*?
pipelines? *isolation*?

Bad news...

A close-up photograph of Donald Trump, wearing a dark suit, white shirt, and red tie. He is covering his eyes with both hands, with his fingers spread. His mouth is slightly open, and he has a distressed or frustrated expression. The background is dark and out of focus. A semi-transparent dark blue diagonal overlay is present in the top right corner.

You're not covered

Microservices-based systems
tend to look like...

“To be fully described, there are
many details, not few”

“The rate of change is high; the systems change before a full description (and therefore understanding) can be completed.”

“How components function is partly unknown, as they resonate with each other across varying conditions.”

“Processes are heterogeneous
and possibly irregular.”

Reactions?

Ugly Risk Avoidance!

A silhouette of a man on the left and a woman on the right, both pointing their right hands towards each other. The man is wearing a suit jacket, and the woman is wearing a dress with a beaded shoulder detail. The background is a light blue-grey gradient with a diagonal line separating a darker blue-grey upper section from a lighter grey lower section.

Blame.

A better reaction?

*Being wrong is a
key software skill*

Get Better At Being Wrong™

Make it Safe(r) to be wrong.

Technical Robustness

Zero Blame

Go *Beyond Blame*

Remember “Dark Debt”

Deliberately Practice
Being Wrong

“prepare for undesirable
circumstances” - John Allspaw

Deliberately Practice
Being Wrong

=

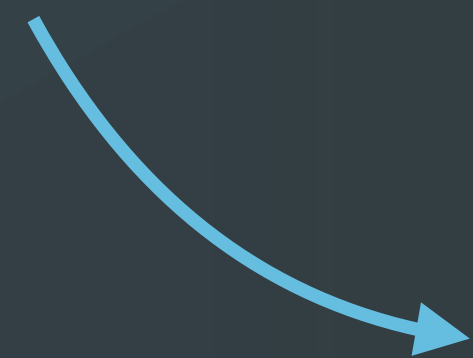
Chaos Engineering

Invest in **Resilience**

Resilience is a
Learning Loop

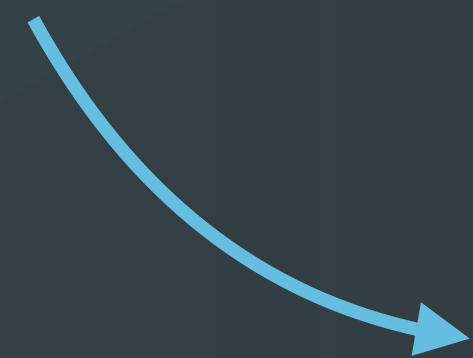
“Normal”

“Normal”

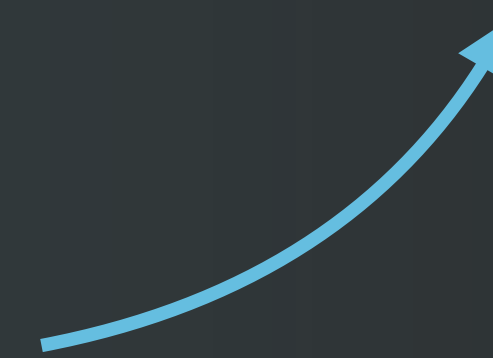


Outage

“Normal”



Outage



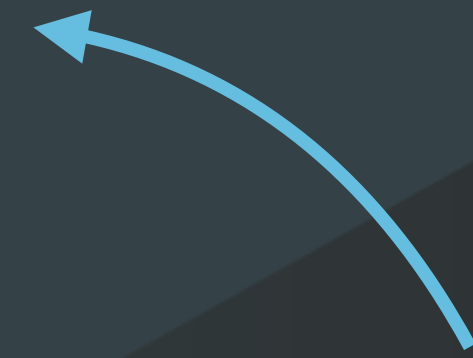
Detection

Diagnosis

Fix



Learning

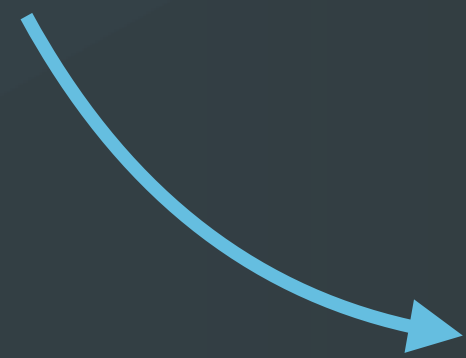


Fix

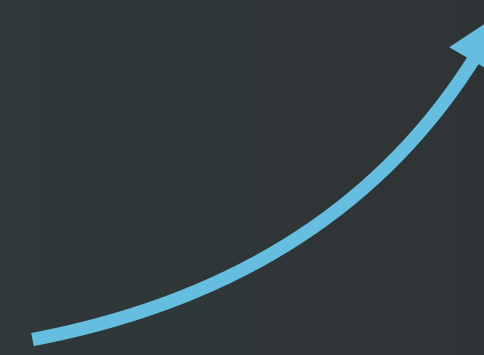
Diagnosis

Detection

“Normal”



Outage





Learning

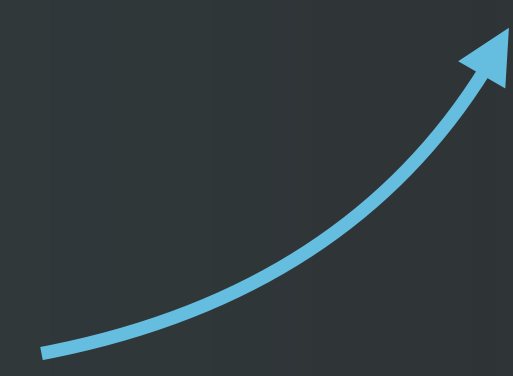
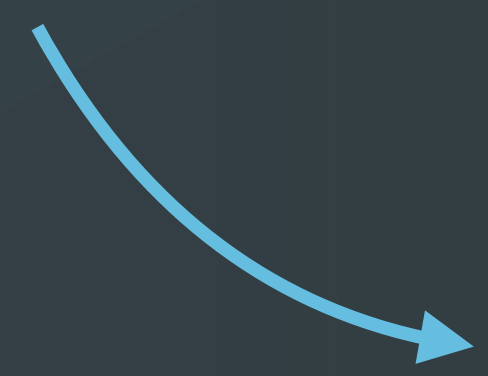
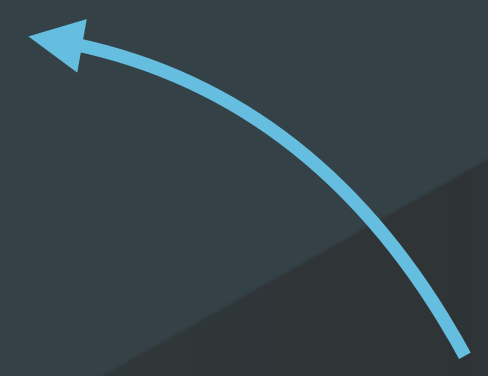
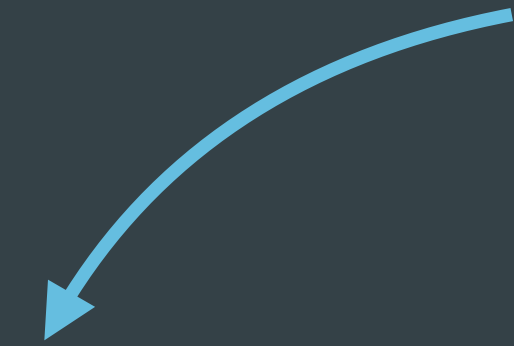
Improvement
(Robustness)

“Normal”



Outage

Fix
Diagnosis
Detection



“Never Let an Outage Go To Waste” - Casey Rosenthal

Post-mortem Learning
is Good

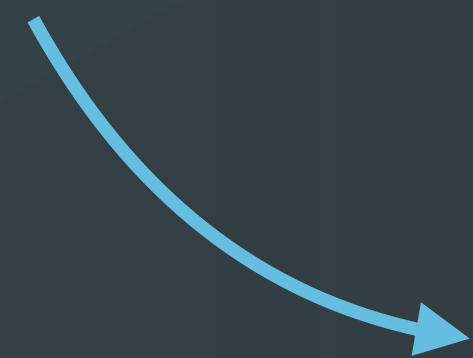
Pre-mortem Learning
is Better!

“Chaos Engineering is the discipline of ***experimenting*** on a distributed ***system*** in order to build ***confidence*** in the system’s ***capability*** to ***withstand turbulent conditions*** in production.”

- principlesofchaos.org

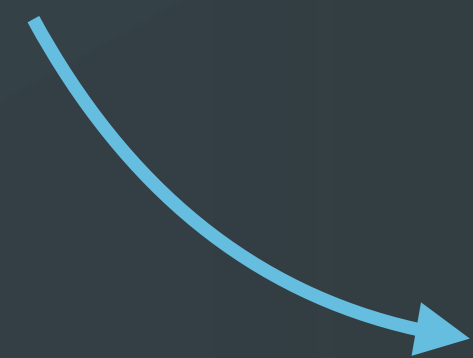
“Normal”

“Normal”



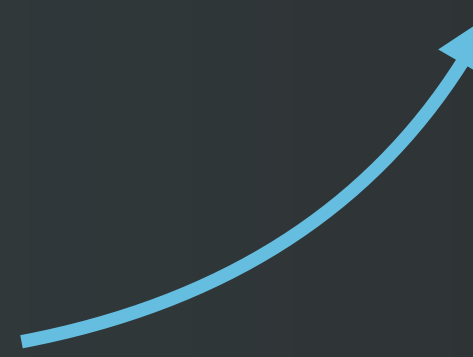
Game Day /
Automated Chaos Experiment

“Normal”



Game Day /

Automated Chaos Experiment



Detection

Diagnosis

Fix



Learning

Improvement
(Robustness)

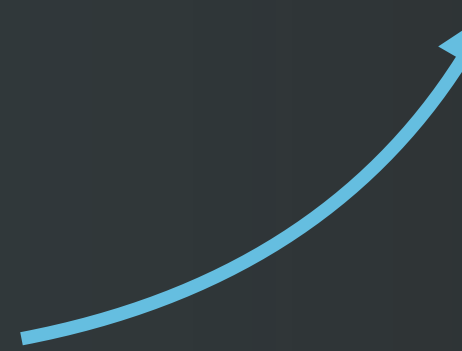
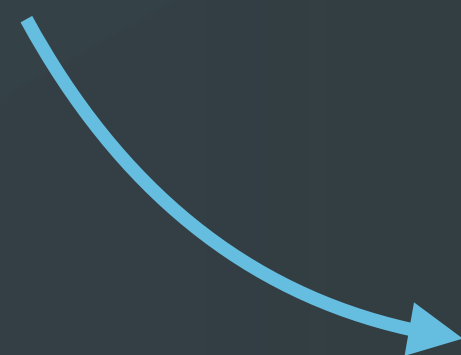
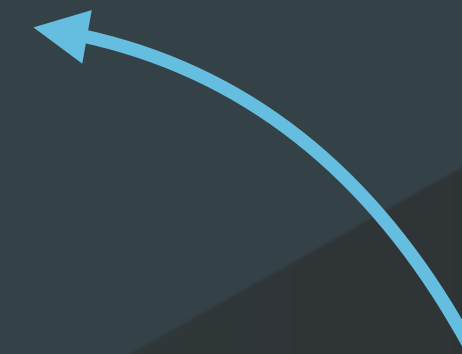
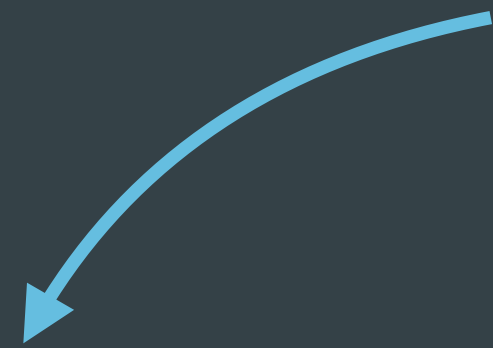
Fix
Diagnosis

“Normal”

Detection



Game Day /
Automated Chaos Experiment





Chaos

Game Days

People, Practices & Process

Automated Experiments

Applications

Automated Experiments

Platform

Automated Experiments

Infrastructure

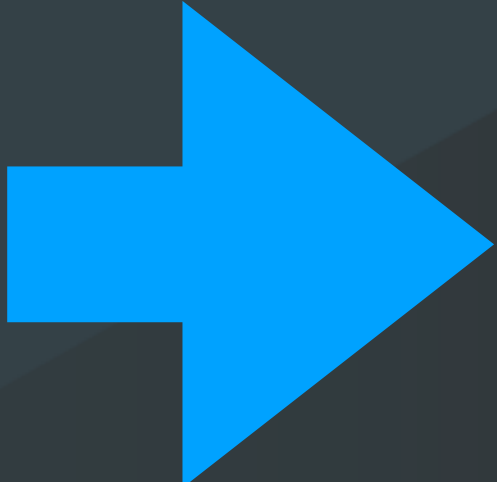
We can *learn after* outages...

but it's even better to
learn from weaknesses *before* an outage.

Being Wrong can be a super
power, if it leads to ***learning***

Establish a Platform for
Pre-mortem, Deliberate Practice
“Being Wrong”

Establish a Platform for
Pre-mortem, Deliberate Practice
“Chaos Engineering”



Automate



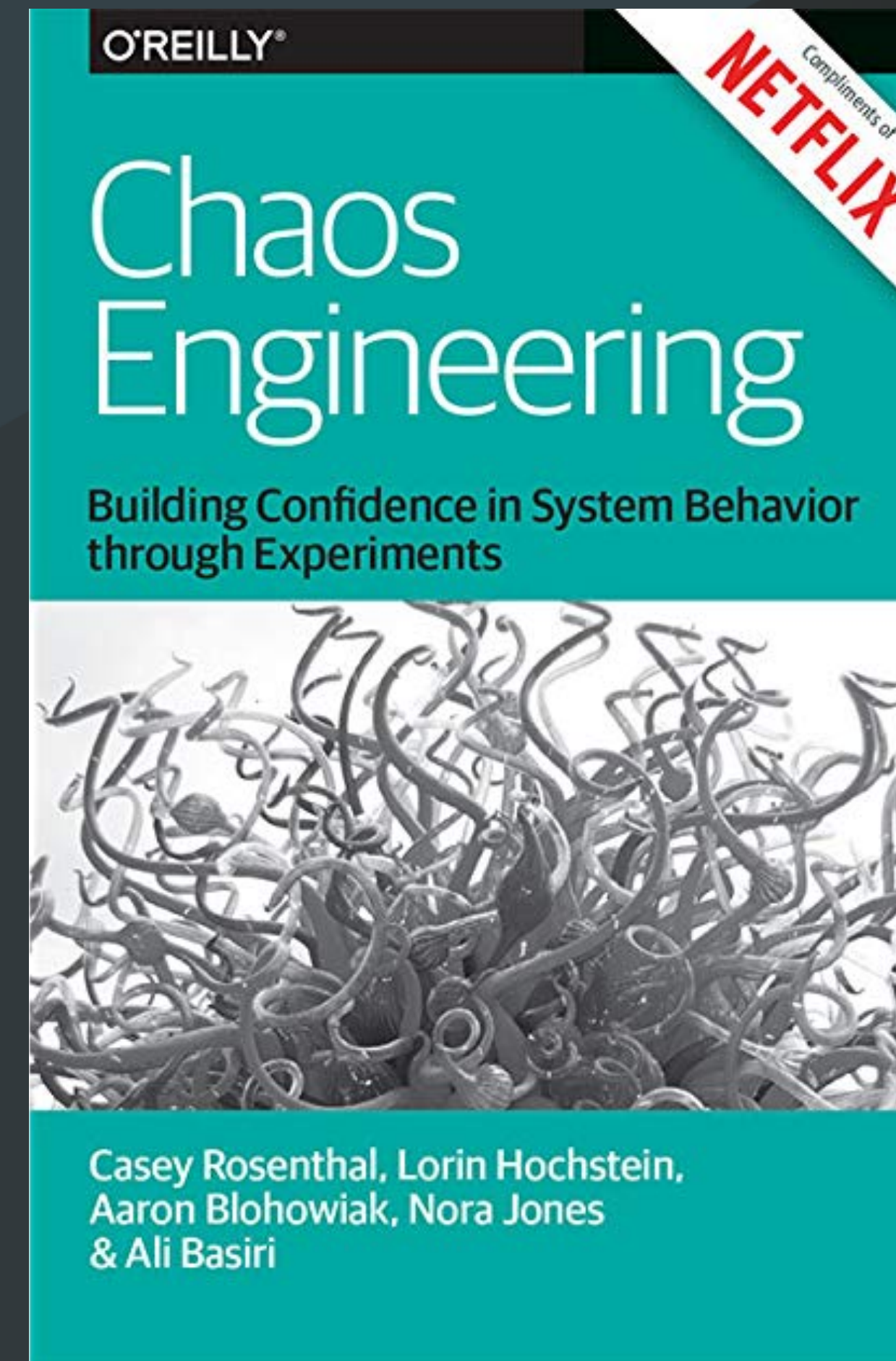
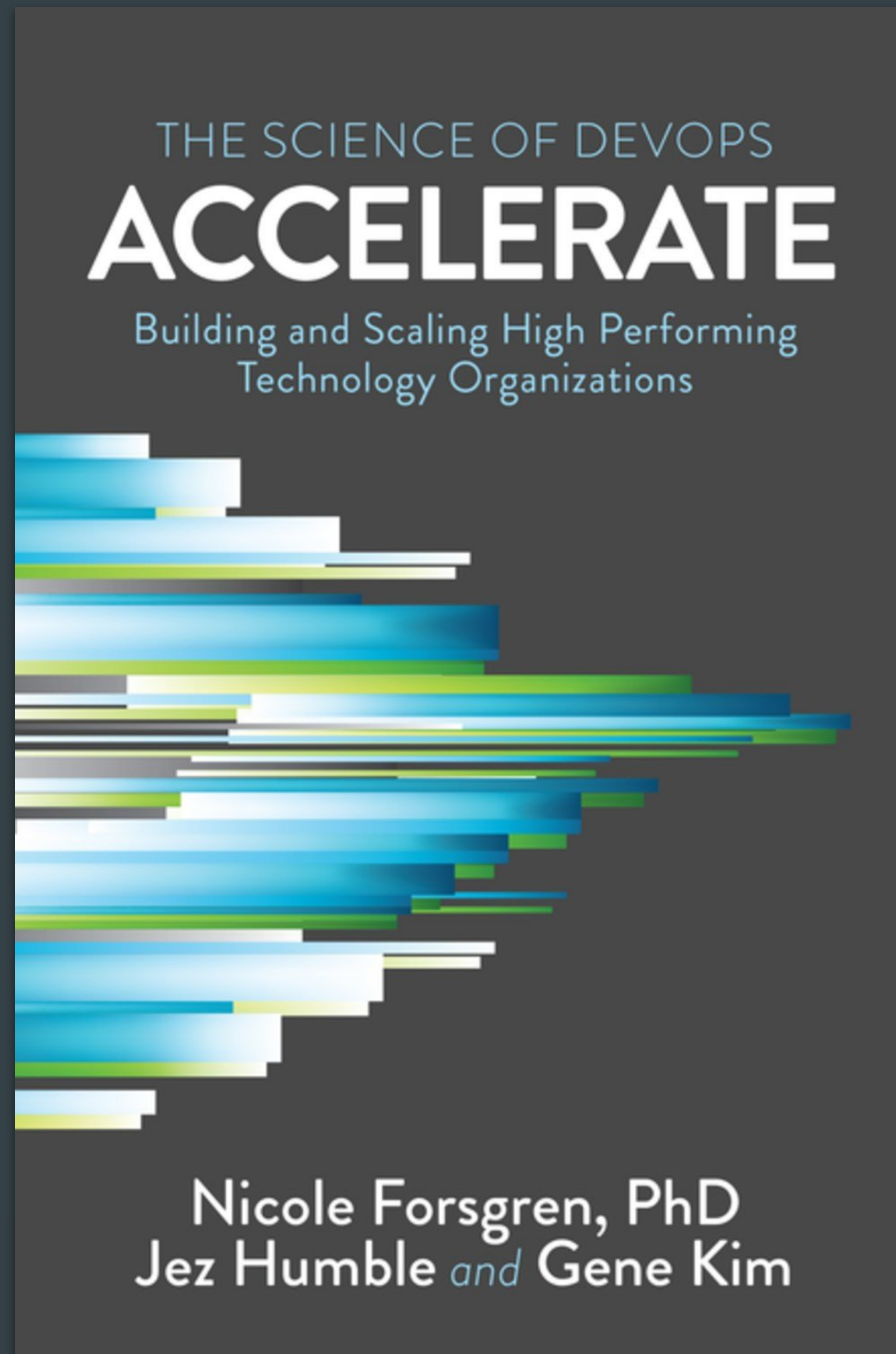
Demo?

Katacoda

<https://katacoda.com/chaostoolkit>

<https://humio.com/chaos-observability>

Reading Recommendations



Start a conversation!



www.chaosiq.io

www.chaostoolkit.org

contact@chaosiq.io

Slack: <https://join.chaostoolkit.org>