DevOps: The Big Picture

Why DevOps is Important Right Now

Richard Seroter
Director of Product Management, Google Cloud

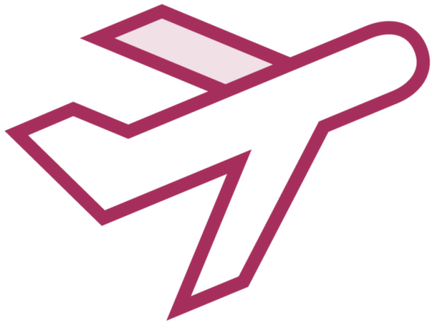
@rseroter www.seroter.com

Overview

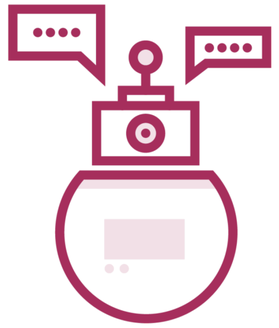
- The ubiquity of software and why “good” matters
- What software development and ops looks like today for many companies
- Overview of Lean and how it applies to software
- Introducing DevOps
- Adjusting to a product mindset
- The data that proves DevOps matters

We Now Expect to Find Software Everywhere

Healthcare



Travel experiences



Customer support

Sporting events

Office work

Shopping

Being “Good at Software” Matters

U.S. consumers changing shopping behavior such as trying new brands or shopping methods

Source: https://www.mckinsey.com/business-functions/marketing-and-sales/our-insights/the-great-consumer-shift-ten-charts-that-show-how-us-shopping-behavior-is-changing

Businesses that think they have less than a year to embrace digital experiences before suffering financially and losing market share

Source: https://www.progress.com/docs/default-source/default-document-library/landing-pages/dach/ebook_digitaltransformation_final.pdf

Let’s Meet Globomantics

Introducing the Company

Midsize enterprise

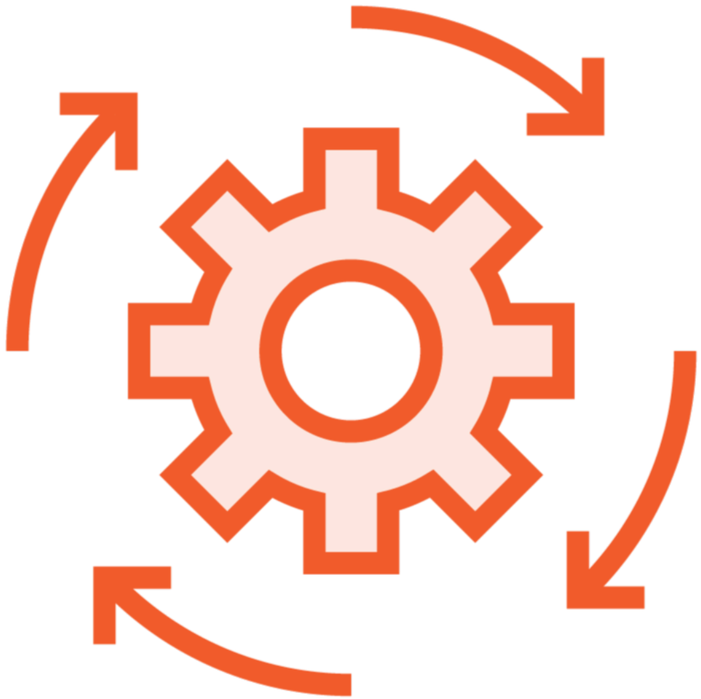
Founded 60 years ago

Sells to businesses, not consumers

Facing serious disruption from competitors

How Globomantics Organizes I.T. Today

Pain Point: Limited Automation



Manual virtual machine builds, even in cloud, causes over-provisioning

Ticket-defined networking prevents rapid changes

Lack of test coverage in code means manual tests

Only parts of environments are stood up with automation, impacting setup time and DR

Pain Point: Slow Delivery

- **Idea-to-production takes months, even years**
- **Work is piled up in front of multiple teams**
- **No self-service platforms for app teams**
- **Multiple change freezes throughout the year**
- **Big-bang releases are the norm, causing downtime**
- **Limited experimentation because of slow feedback loops**

Pain Point: Poor Uptime

Poor change failure rates, with the same mistakes occurring repeatedly

Long mean-time-to-recovery

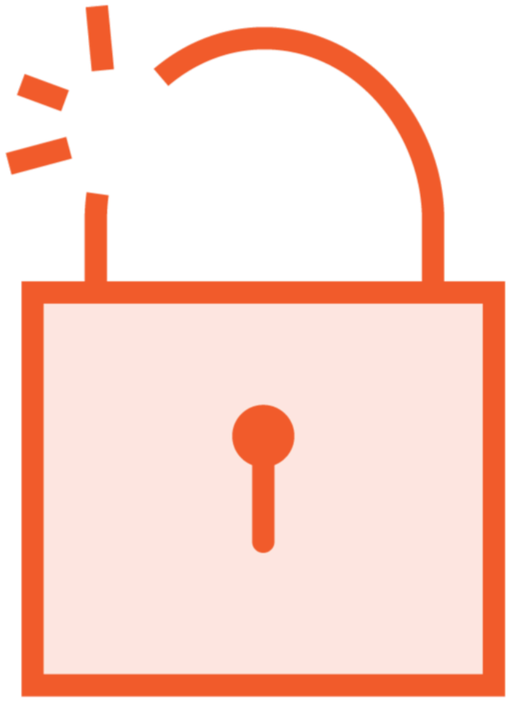
Panic when things go wrong

Lack of complete, centralized telemetry delays resolution time

Over-alerting has led to ignored alarms

Teams focused on preventing failures

Pain Point: Security Gaps



- **Security concerns offloaded to InfoSec**
- **No common approach to credential and secrets management**
- **Inconsistent hardening of compute environments, including container images**
- **Challenges understanding vulnerabilities in app dependencies**
- **Different policies applies to cloud and on-prem environments**
- **Security often sacrificed to attempt speed**

Pain Point: Lacking a Customer Focus

Teams think about their silo, not the total product

Every department has unique measure of success

No shared definition of “done”

Quality issues deprioritized to ship on time

No A/B testing or feature-based usage monitoring

Customer support is handled centrally

The Result

Poor customer satisfaction

Revenue is declining

Best employees are leaving

No innovations in years

LOBs lack faith in I.T.

Us vs. them mentality taking over

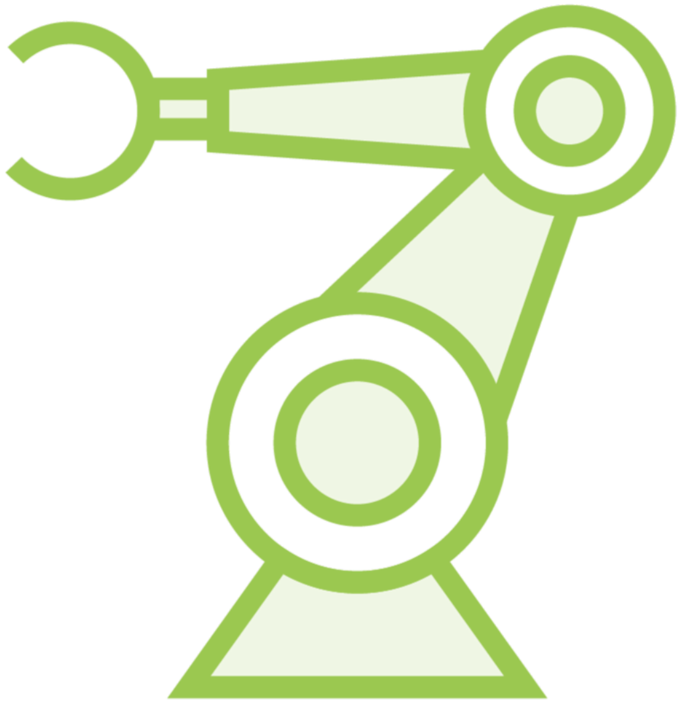
Rethinking Delivery

Looking to Manufacturing for Inspiration

Automotive companies like Toyota reinvented manufacturing

Fresh thinking on flow, continuous improvement, and empowerment

Focus on consistency, sustainable work, and eliminating waste



Identifying Waste

Knowledge waste

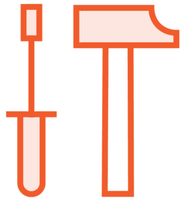
Waiting waste

Over-processing waste

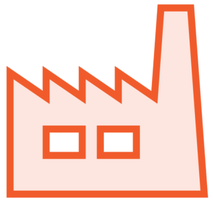
Motion waste

Identifying Waste (cont.)

Transportation waste



Correction waste



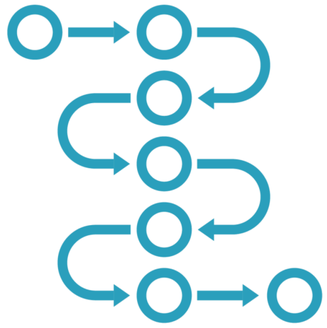
Inventory waste

Over-production waste

What Is Lean All About?

Focus on value

Eliminate waste



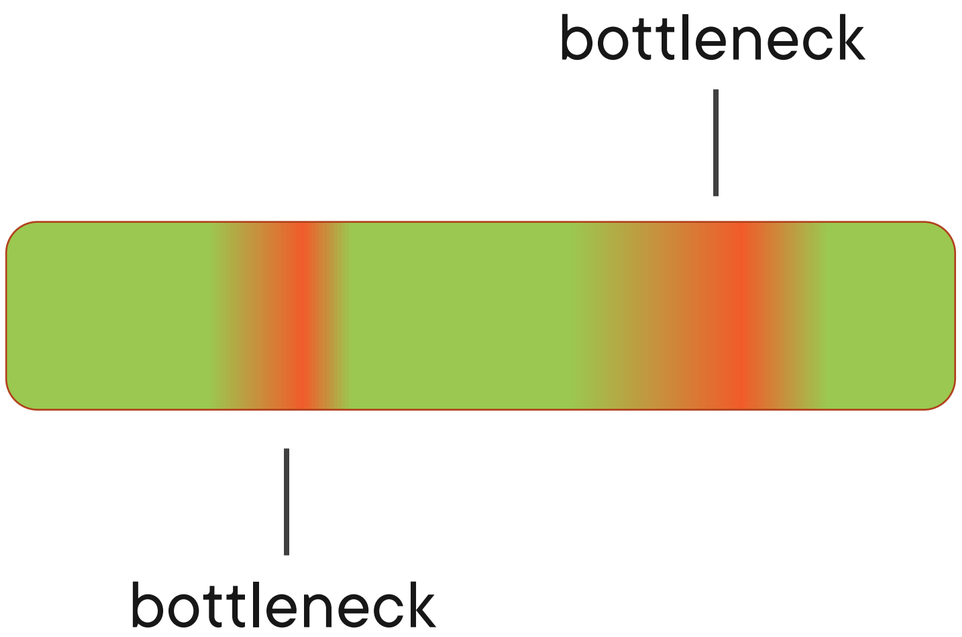
One-piece flow

Continuous learning

No overburdening of people

Apply Theory of Constraints

Exploring the Theory of Constraints

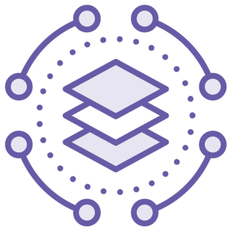


Throughput is limited by the constraint(s)

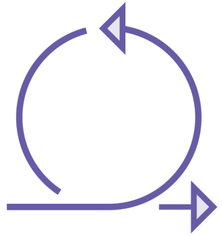
Optimizing before it results in excess inventory

Optimizing after it results in starvation

How Lean Relates to Software Delivery

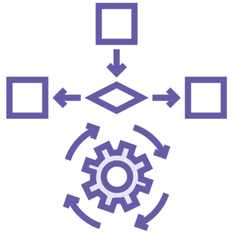


Reorient towards product thinking

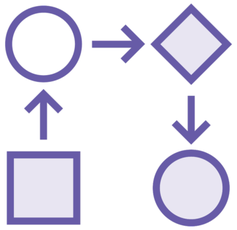


Focus on continuous improvement

Remove wasteful processes



Automate to remove overburdening of staff



Continuously deliver software in small batches

Attack bottlenecks that decrease flow

Introducing DevOps

Defining DevOps

“DevOps represents a change in IT culture, focusing on rapid IT service delivery through the adoption of agile, lean practices in the context of a system-oriented approach.”

Gartner Research, *https://www.gartner.com/en/information-technology/glossary/devops*

Defining DevOps

“DevOps is the combination of cultural philosophies, practices, and tools that increases an organization’s ability to deliver applications and services at high velocity.”

AWS, *https://aws.amazon.com/devops/what-is-devops/*

Defining DevOps

“DevOps is a way to release better software. It is not just technical tools or workflows. DevOps is also a cultural practice. DevOps produces better software, faster, by aligning development, staging, and deployment.”

Puppet, *https://puppet.com/devops/*

Defining DevOps

“DevOps is a set of practices that works to automate and integrate the processes between software development and IT teams, so they can build, test, and release software faster and more reliably.”

Atlassian, *https://www.atlassian.com/devops*

Having a Product Mindset

Project-based mindset	Product-based mindset
Funding by project milestones	Funding based on output
Success based on time, budget targets	Success based on business value
Teams assembled from staff allocated for the term of the project	Teams comprised of multi-disciplinary staff to one product at a time
Durable knowledge lives in documentation	Durable knowledge within the team and select documentation
Has a defined end date, with little attention on post-release lifecycle	Look at multi-year lifecycle and consider ongoing activities

Source: Kersten, Mik, *Project to Product*. IT Revolution Press, 2018

What DevOps is NOT

DevOps is not a team or software methodology

DevOps is not something you buy

DevOps is not an IT-only effort

DevOps is not just automating infrastructure

DevOps is not easy!

The Data Proves This Approach Works

973x

More frequent code deployments

3x

Lower change failure rate

1.6x

More likely to meet or exceed organizational goals when integrating security best practices

Source: 2021 Accelerate State of DevOps Report

Summary

- The ubiquity of software and why “good” matters
- What software development and ops looks like today for many companies
- Overview of Lean and how it applies to software
- Introducing DevOps
- Adjusting to a product mindset
- The data that proves DevOps matters