



Cloud  
Technology  
Partners

# How To Build a DevOps Team in 60 Days

Effective DevOps

# What is DevOps and Why Do Organizations Adopt It?

---

## DevOps drives business agility and performance



### High-performance IT shops:

- Are twice as likely to exceed their profitability, market share, and productivity goals
- Adopt DevOps practices to achieve performance gains
- Deploy 30 times more frequently with 50% fewer failures
- Embrace DevOps culture and its aptitude for continuous learning
- Experience higher employee satisfaction and retention

## What is DevOps?

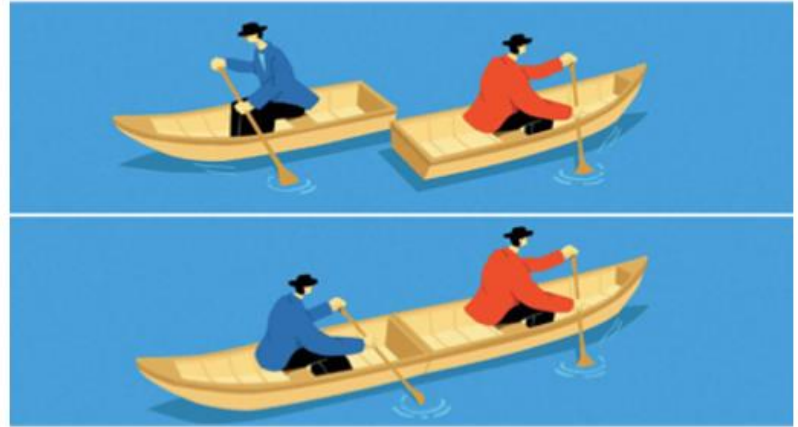
Many believe DevOps is not defined, but something you know when you see it:

- A movement of people who care about developing and operating reliable, secure, high performance systems at scale.
  - *Jez Humble - author of “Continuous Delivery” and “Lean Enterprise”*
- A cultural and professional movement, focused on how we build and operate high-velocity organizations, born from the experiences of its practitioners
  - *Adam Jacob - Founder and CTO at Chef*
- A philosophical movement, and not yet a precise collection of practices, descriptive or prescriptive (e.g., CMM-I, ITIL, Agile, etc.)
  - *Gene Kim - Author “The Phoenix Project” and “The Visible Ops Handbook,” and founder and former CTO of Tripwire*

## A working definition

### Let's start with a simple definition:

**DevOps:** A culture shift or movement that encourages greater collaboration (aka teamwork) to foster building better quality software more quickly with more reliability



## DevOps is a vague term that is broadly misunderstood

### DevOps is NOT:

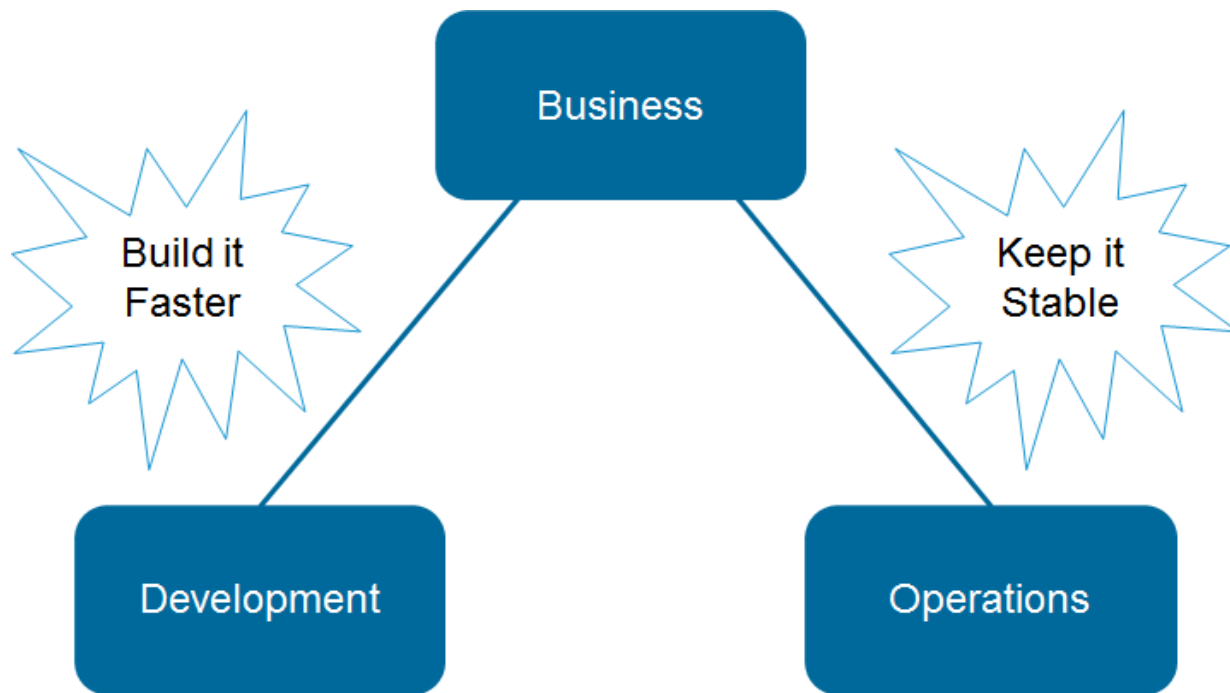
- A role, person, or organization - you can't hire a DevOps
- Something only systems administrators do
- Something only developers do
- Just writing Chef and Puppet scripts
- A set of tools - you cannot purchase DevOps software

## What is driving DevOps adoption?

- Modern architectures are harder to manage and scale using traditional processes
- Automation - the human element remains unpredictable in the operation of fault-tolerant and reliable software
- More and more revenues tied directly to software up-time reliability
- Agile software development and its inevitable application to system administration and operations
- Pressure on CIOs to deliver services faster and cheaper



## Traditional - The competing interests of the cross functional silos





## Traditional - The (not so) great balancing act

### Fast Development

Speed  
APIs  
Agility  
Features



### Stable Operations

Security  
Compliance  
Auditing  
Availability

**When applications fail, the blame game results!**

## A traditional approach does not align everyone with an always-up service model

Shift thinking away from (shrink-wrap) product-centric to service-centric

Old Way		New Way
Software is built and shipped to operations or customers	➔	Services are running and managed
Development of features are done	➔	Services are never done until they are turned off
Each silo owns its own area	➔	All groups collectively own quality, reliability and security
Dev must go through Ops to get work done	➔	Ops enables Dev to get work done

Dev, Ops, and Security teams must work together throughout the systems life-cycle and share responsibility for the services

# DevOps Values and Practices

---

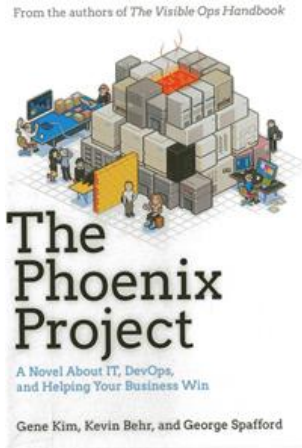
**Summary:** Learn DevOps values and principles - collaboration (relationships), automation, and continuous improvement

**We now see a repetitive set of themes in our study of the DevOps movement:**

- **Collaboration** and building relationships across all functions with all involved in the application development/delivery cycle is critical to solving the many challenges presented by operating a complex system
- **Automation is critical** to manage scaled infrastructure and repeatable processes
- **Continuous improvement** means incremental and iterative enhancements to all phases of the software life cycle

**Objective:** Understand what “the three ways” are as described by Gene Kim, notable author and practitioner of DevOps

## Understand “The Three Ways” as described by Gene Kim, notable author and practitioner of DevOps



“The three ways’ of DevOps is a strategy to improve operations. It provides a framework for managing process, procedures, and practices with a DevOps philosophy.” Gene Kim

Lean  
Agile  
Systems  
Thinking

**Summary:** Learn that the roots of DevOps are in agile - the meaning of the manifesto for agile software development

**“The three ways” of DevOps is a strategy to improve operations:**

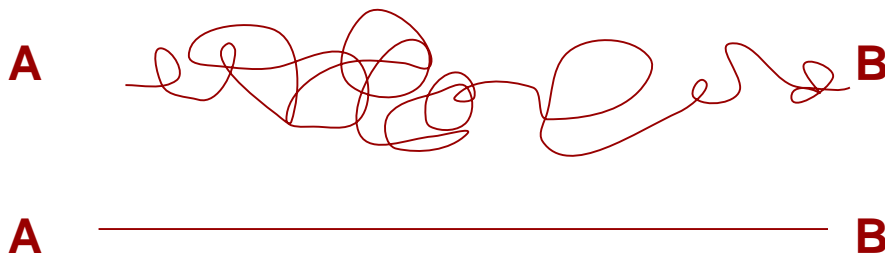
- The first way emphasizes how important it is to know and analyze the performance of the entire value stream, not just its individual silos or work processes
- The second way instructs a pattern where communications in the value stream must flow clearly and in a way that promotes learning
- The third way promotes risk-taking and learning from failure

# The Five Rules of DevOps

---

## Rule 1: Remove waste!

Removing waste is the most fundamental DevOps goal!





## Rule 1: Remove waste!

### Common people bottlenecks:

- Organizational silos
- Dependence on heroic efforts
- Misaligned or inconsistent incentives
- Resistance to change
- Lack of governance

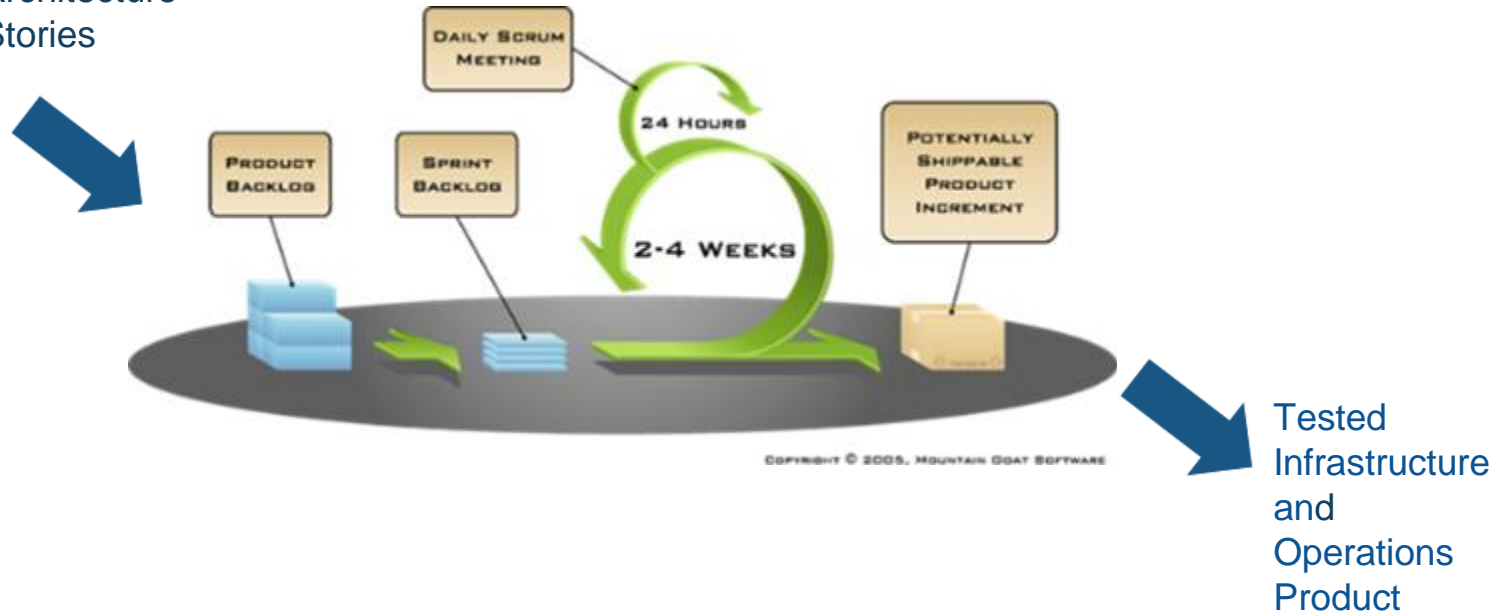
## Rule 2: Treat DevOps like product development

Infrastructure  
Stories

Security  
Stories

Data Mgmt  
Stories

Architecture  
Stories



## Rule 3: Full-stack automation

### The DevOps dream: Automated delivery of everything

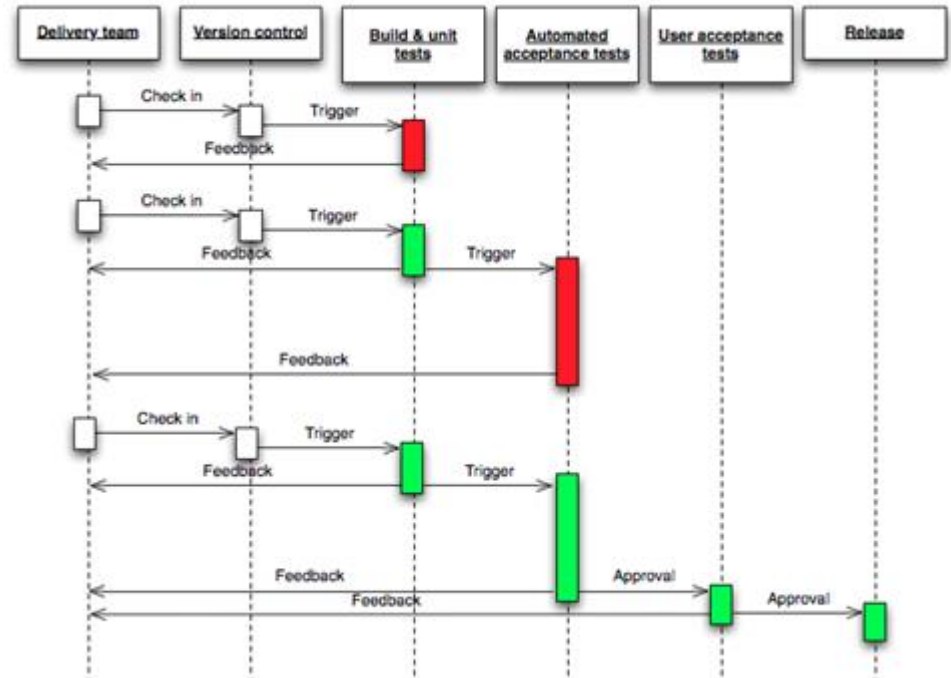
- The cloud makes it easy to get whatever resources you need on-demand
- Teams have a self-service portal for all non-production environments
- Automated provisioning ensures full security and compliance
- Fully automated deployments are simple to roll back



## Rule 4: Continuous delivery and self-service integration

### The four practices of continuous delivery:

- Build binaries (application code) only once
- Use the same repeatable process for deployment for all environments - Test, Integration and Production
- Do basic functionality testing on your deployments
- If anything fails, stop the line immediately and start again



# Chapter 5: The Five Rules of DevOps

## Rule 5: Measure everything for continuous improvement

### Monitoring, logging, and alert dashboards

- Monitor all parts of the value streams
- Monitor all environments and phases in the software development life cycle
- Avoid noise and focus on actionable / meaningful metrics and events



## Rule 5: Measure everything for continuous improvement

### DevOps KPIs/Metrics -- The Four Buckets:

1. Resource utilization
2. Failure rates
3. Throughput
4. Agility



## DevOps Maturity Model illustrates that DevOps is a journey and not a destination!

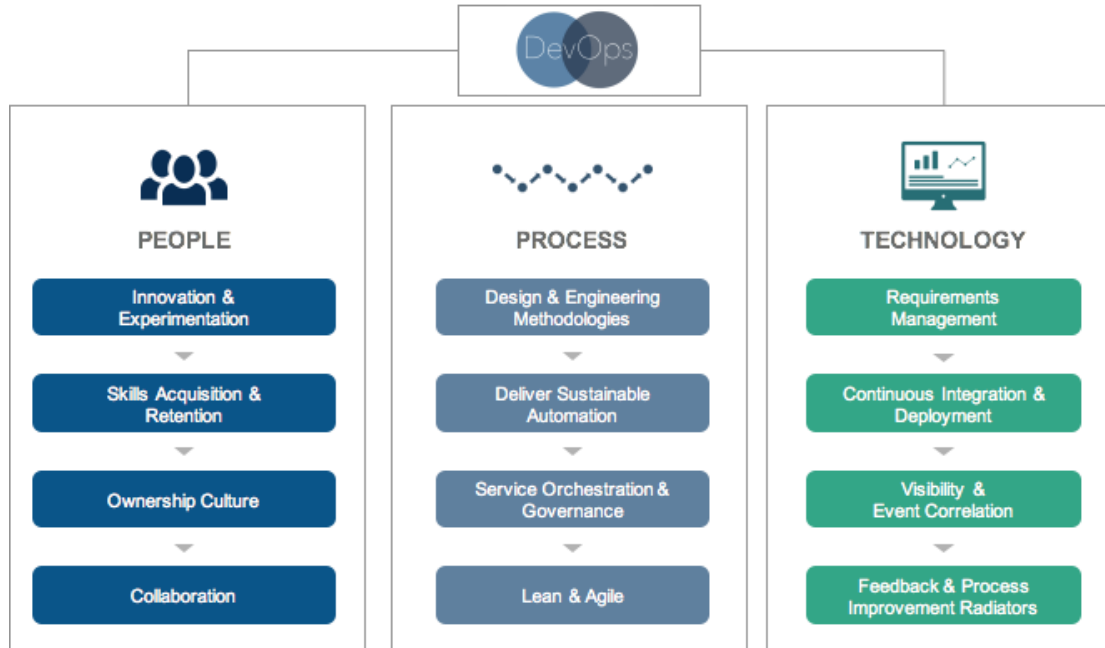
Maturity Level	People	Process	Technology
<b>Level 1 Ad-Hoc</b>	<ul style="list-style-type: none"> <li>• Silo based</li> <li>• Blame, finger pointing</li> <li>• Dependent on experts</li> </ul>	<ul style="list-style-type: none"> <li>• Manual processes</li> <li>• Tribal knowledge is the norm</li> </ul>	<ul style="list-style-type: none"> <li>• Manual builds and deployments</li> <li>• Manual testing</li> </ul>
<b>Level 2 Repeatable</b>	<ul style="list-style-type: none"> <li>• Managed communications</li> <li>• Limited knowledge sharing</li> </ul>	<ul style="list-style-type: none"> <li>• Processes established within silos</li> <li>• No standards</li> </ul>	<ul style="list-style-type: none"> <li>• Automated builds</li> <li>• Automated tests written as part of story development</li> </ul>
<b>Level 3 Defined</b>	<ul style="list-style-type: none"> <li>• Collaboration exists</li> <li>• Shared decision making</li> <li>• Shared Accountability</li> </ul>	<ul style="list-style-type: none"> <li>• Processes are automated across SDLC</li> <li>• Standards across organization</li> </ul>	<ul style="list-style-type: none"> <li>• Automated build &amp; test cycle for every commit</li> <li>• Push button deployments</li> <li>• Automated user &amp; acceptance testing</li> </ul>
<b>Level 4 Measured</b>	<ul style="list-style-type: none"> <li>• Collaboration backed on shared metrics with a focus on removing bottlenecks</li> </ul>	<ul style="list-style-type: none"> <li>• Proactive monitoring</li> <li>• Metrics collected and analyzed against business goals</li> </ul>	<ul style="list-style-type: none"> <li>• Build metrics visible and acted on</li> <li>• Orchestrated deployments with auto rollbacks</li> </ul>
<b>Level 5 Optimized</b>	<ul style="list-style-type: none"> <li>• A culture of continuous improvement permeates through the organization</li> </ul>	<ul style="list-style-type: none"> <li>• Self service automation</li> <li>• Risk &amp; cost optimization</li> <li>• High degree of experimentation</li> </ul>	<ul style="list-style-type: none"> <li>• Zero downtime deployments</li> <li>• Immutable infrastructure</li> </ul>

## DevOps Maturity Model - The journey to Continuous Operations!

Maturity Level	People	Process	Technology
<b>Level 1 Ad-Hoc</b>	<ul style="list-style-type: none"> <li>Silo based</li> <li>Blame, finger pointing</li> <li>Dependent on experts</li> </ul>	<p><b>Chaos Reigns</b></p> <ul style="list-style-type: none"> <li>...ses</li> <li>...ge is the norm</li> </ul>	<ul style="list-style-type: none"> <li>Manual builds and deployments</li> <li>Manual testing</li> </ul>
<b>Level 2 Repeatable</b>	<ul style="list-style-type: none"> <li>Managed comm</li> <li>Limited knowlec</li> </ul>	<p><b>Continuous Integration</b></p> <ul style="list-style-type: none"> <li>No standards</li> </ul>	<ul style="list-style-type: none"> <li>Automated builds</li> <li>Automated tests written as part of story development</li> </ul>
<b>Level 3 Defined</b>	<ul style="list-style-type: none"> <li>Collaboration exists</li> <li>Shared decision r</li> <li>Shared Accountal</li> </ul>	<p><b>Continuous Delivery</b></p> <ul style="list-style-type: none"> <li>Processes are automated</li> <li>organization</li> </ul>	<ul style="list-style-type: none"> <li>Automated build &amp; test cycle for every commit</li> <li>Push button deployments</li> <li>Automated user &amp; acceptance testing</li> </ul>
<b>Level 4 Measured</b>	<ul style="list-style-type: none"> <li>Collaboration backed on shared metrics</li> <li>on removing bottlenecks</li> </ul>	<p><b>Continuous Deployment</b></p> <ul style="list-style-type: none"> <li>Proactive monitoring</li> <li>analyzed against business goals</li> </ul>	<ul style="list-style-type: none"> <li>Build metrics visible and acted on</li> <li>Orchestrated deployments with auto rollbacks</li> </ul>
<b>Level 5 Optimized</b>	<ul style="list-style-type: none"> <li>A culture of continuous improvement pe</li> <li>through the org:</li> </ul>	<p><b>Continuous Operations</b></p> <ul style="list-style-type: none"> <li>Self service automation</li> <li>experimentation</li> </ul>	<ul style="list-style-type: none"> <li>Zero downtime deployments</li> <li>Immutable infrastructure</li> </ul>



# People, process, and technology weave together the DevOps elements





Cloud  
Technology  
Partners

**Boston Headquarters**

263 Summer Street  
Fourth Floor  
Boston MA, 02210

**Contact**

617.674.0874  
info@cloudtp.com  
www.cloudtp.com

**Thank you for your time and interest.**

CLOUD WITH  
**CONFIDENCE**<sup>®</sup>