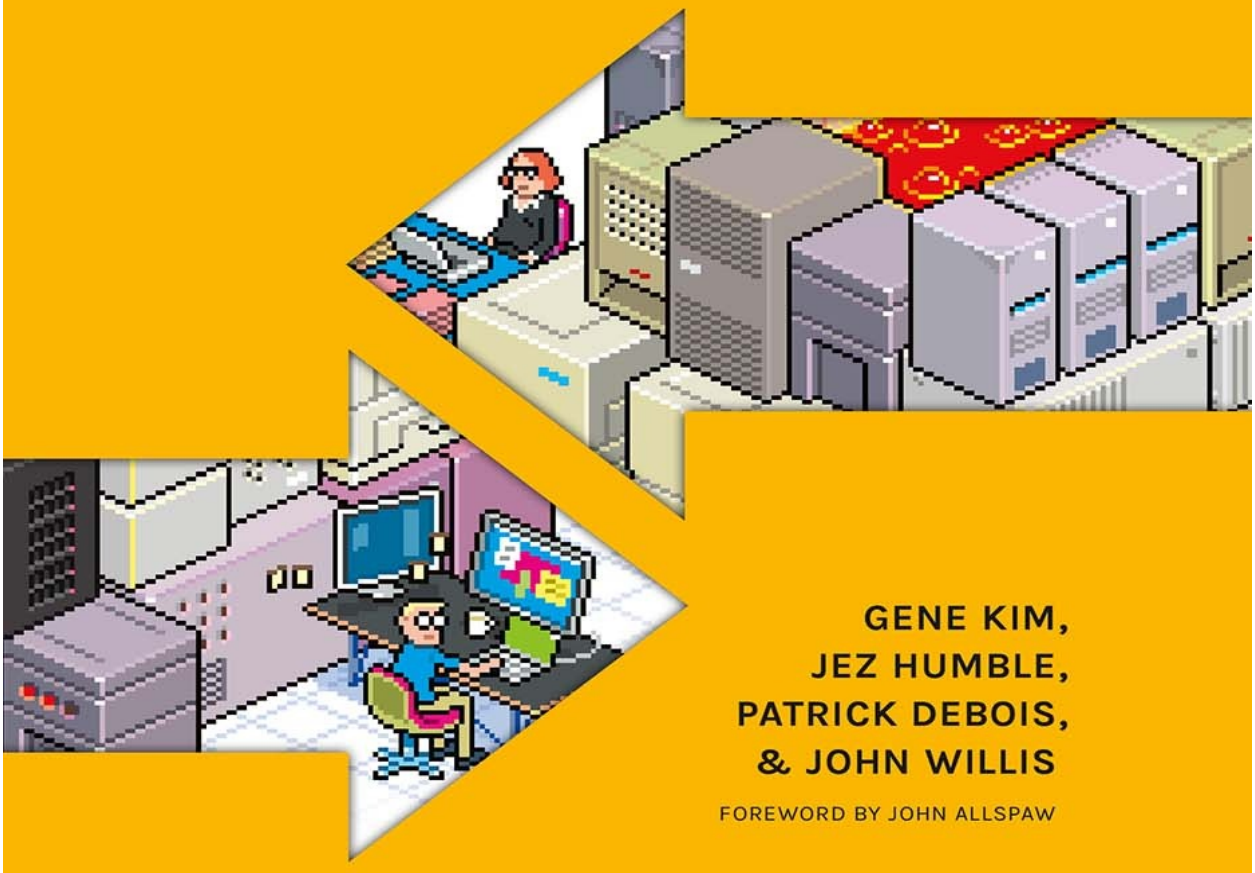
The DevOps Handbook

HOW TO CREATE WORLD-CLASS AGILITY, RELIABILITY, & SECURITY IN TECHNOLOGY ORGANIZATIONS



GENE KIM,
JEZ HUMBLE,
PATRICK DEBOIS,
& JOHN WILLIS

FOREWORD BY JOHN ALLSPAW

TAKE THE DORA DEVOPS X-RAY ASSESSMENT AND SEE WHERE YOU STAND.

configurations in source control and adopt CI/CD delivery patterns for their workflow. Being the old IT Operations person at the time, I think I replied to him with something like, “That idea is going to sink like Led Zeppelin with Ops folk.” (I was clearly wrong.)

Then about a year later in 2009 at another O’Reilly conference, Velocity, I saw Andrew Clay Shafer give a presentation on Agile Infrastructure. In his presentation, Andrew showed this iconic picture of a wall between developers and operations with a metaphorical depiction of work being thrown over the wall. He coined this “the wall of confusion.” The ideas he expressed in that presentation codified what Luke was trying to tell me a year earlier. That was the light bulb for me. Later that year, I was the only American invited to the original DevOpsDays in Ghent. By the time that event was over, this thing we call DevOps was clearly in my blood.

Clearly, the co-authors of this book all came to a similar epiphany, even if they came there from very different directions. But there is now an overwhelming weight of evidence that the problems described above happen almost everywhere, and that the solutions associated with DevOps are nearly universally applicable.

The goal of writing this book is to describe how to replicate the DevOps transformations we’ve been a part of or have observed, as well as dispel many of the myths of why DevOps won’t work in certain situations. Below are some of the most common myths we hear about DevOps.

Myth—*DevOps is Only for Startups:* While DevOps practices have been pioneered by the web-scale, Internet “unicorn” companies such as Google, Amazon, Netflix, and Etsy, each of these organizations has, at some point in their history, risked going out of business because of the problems associated with more traditional “horse” organizations: highly dangerous code releases that

were prone to catastrophic failure, inability to release features fast enough to beat the competition, compliance concerns, an inability to scale, high levels of distrust between Development and Operations, and so forth.

However, each of these organizations was able to transform their architecture, technical practices, and culture to create the amazing outcomes that we associate with DevOps. As Dr. Branden Williams, an information security executive, quipped, “Let there be no more talk of DevOps unicorns or horses but only thoroughbreds and horses heading to the glue factory.”

Myth—*DevOps Replaces Agile:* DevOps principles and practices are compatible with Agile, with many observing that DevOps is a logical continuation of the Agile journey that started in 2001. Agile often serves as an effective enabler of DevOps, because of its focus on small teams continually delivering high quality code to customers.

Many DevOps practices emerge if we continue to manage our work beyond the goal of “potentially shippable code” at the end of each iteration, extending it to having our code always in a deployable state, with developers checking into trunk daily, and that we demonstrate our features in production-like environments.

Myth—*DevOps is incompatible with ITIL:* Many view DevOps as a backlash to ITIL or ITSM (IT Service Management), which was originally published in 1989. ITIL has broadly influenced multiple generations of Ops practitioners, including one of the co-authors, and is an ever-evolving library of practices intended to codify the processes and practices that underpin world-class IT Operations, spanning service strategy, design, and support.

DevOps practices can be made compatible with ITIL process. However, to support the shorter lead times and higher deployment frequencies associated

with DevOps, many areas of the ITIL processes become fully automated, solving many problems associated with the configuration and release management processes (e.g., keeping the configuration management database and definitive software libraries up to date). And because DevOps requires fast detection and recovery when service incidents occur, the ITIL disciplines of service design, incident, and problem management remain as relevant as ever.

Myth—*DevOps is Incompatible with Information Security and Compliance:* The absence of traditional controls (e.g., segregation of duty, change approval processes, manual security reviews at the end of the project) may dismay information security and compliance professionals.

However, that doesn't mean that DevOps organizations don't have effective controls. Instead of security and compliance activities only being performed at the end of the project, controls are integrated into every stage of daily work in the software development life cycle, resulting in better quality, security, and compliance outcomes.

Myth—*DevOps Means Eliminating IT Operations, or "NoOps":* Many misinterpret DevOps as the complete elimination of the IT Operations function. However, this is rarely the case. While the nature of IT Operations work may change, it remains as important as ever. IT Operations collaborates far earlier in the software life cycle with Development, who continues to work with IT Operations long after the code has been deployed into production.

Instead of IT Operations doing manual work that comes from work tickets, it enables developer productivity through APIs and self-serviced platforms that create environments, test and deploy code, monitor and display production telemetry, and so forth. By doing this, IT Operations become more like Development (as do QA and Infosec), engaged in product development, where

the product is the platform that developers use to safely, quickly, and securely test, deploy, and run their IT services in production.

Myth—*DevOps is Just "Infrastructure as Code" or Automation:* While many of the DevOps patterns shown in this book require automation, DevOps also requires cultural norms and an architecture that allows for the shared goals to be achieved throughout the IT value stream. This goes far beyond just automation. As Christopher Little, a technology executive and one of the earliest chroniclers of DevOps, wrote, "DevOps isn't about automation, just as astronomy isn't about telescopes."

Myth—*DevOps is Only for Open Source Software:* Although many DevOps success stories take place in organizations using software such as the LAMP stack (Linux, Apache, MySQL, PHP), achieving DevOps outcomes is independent of the technology being used. Successes have been achieved with applications written in Microsoft.NET, COBOL, and mainframe assembly code, as well as with SAP and even embedded systems (e.g., HP LaserJet firmware).

SPREADING THE AHA! MOMENT

Each of the authors has been inspired by the amazing innovations happening in the DevOps community and the outcomes they are creating: they are creating safe systems of work, and enabling small teams to quickly and independently develop and validate code that can be safely deployed to customers. Given our belief that DevOps is a manifestation of creating dynamic, learning organizations that continually reinforce high-trust cultural norms, it is inevitable that these organizations will continue to innovate and win in the marketplace.

It is our sincere hope that *The DevOps Handbook* will serve as a valuable resource for many people in different ways: a guide for planning and executing

DevOps transformations, a set of case studies to research and learn from, a chronicle of the history of DevOps, a means to create a coalition that spans Product Owners, Architecture, Development, QA, IT Operations, and Information Security to achieve common goals, a way to get the highest levels of leadership support for DevOps initiatives, as well as a moral imperative to change the way we manage technology organizations to enable better effectiveness and efficiency, as well as enabling a happier and more humane work environment, helping everyone become lifelong learners—this not only helps everyone achieve their highest goals as human beings, but also helps their organizations win.