# ACCELERATE

"Business as usual" is no longer enough to remain competitive. Organizations in all industries, from finance and banking to retail, telecommunications, and even government, are turning away from delivering new products and services using big projects with long lead times. Instead, they are using small teams that work in short cycles and measure feedback from users to build products and services that delight their customers and rapidly deliver value to their organizations. These high performers are working incessantly to get better at what they do, letting no obstacles stand in their path, even in the face of high levels of risk and uncertainty about how they may achieve their goals.

To remain competitive and excel in the market, organizations must accelerate:

- delivery of goods and services to delight their customers;
- engagement with the market to detect and understand customer demand;
- anticipation of compliance and regulatory changes that impact their systems; and
- response to potential risks such as security threats or

changes in the economy.

At the heart of this acceleration is software. This is true of organizations in any industry vertical. Banks no longer deliver value by holding gold bars in vaults but by trading faster and more securely, and by discovering new channels and products to engage customers. Retailers win and retain customers by offering them superior selection and service, with service coming in the form of a fast check-out experience, recommended goods at check-out, or a seamless online/offline shopping experience—all of which are enabled by technology. Government organizations cite the ability to harness technology as the key to serving the public more effectively and efficiently while being parsimonious with taxpayer dollars.

Software and technology are key differentiators for organizations to deliver value to customers and stakeholders. We've found it in our own research outlined in this book—and others have found it, too. For example, a recent study by James Bessen of Boston University found that the strategic use of technology explains revenue and productivity gains more than mergers and acquisitions (M&A) and entrepreneurship (2017). Andrew McAfee and Erik Brynjolfsson have also found a link between technology and profitability (2008).

Software is transforming and accelerating organizations of all kinds. The practices and capabilities we talk about in this book have emerged from what is now known as the DevOps movement, and they are transforming industries everywhere. DevOps emerged from a small number of organizations facing a wicked

problem: how to build secure, resilient, rapidly evolving distributed systems at scale. In order to remain competitive, organizations must learn how to solve these problems. We see that large enterprises with long histories and decades-old technologies also gain significant benefits, such as accelerated delivery and lower costs, through adopting the capabilities we outline in this book.

Although many organizations have achieved great success with their technology transformations (notable examples include web-scale tech giants such as Netflix, Amazon, Google, and Facebook, as well as more traditional large organizations including Capital One, Target, and the US Federal Government's Technology Transformation Service and US Digital Service), there is still a lot of work to be done—both in the broader industry and within individual organizations. A recent Forrester (Stroud et al. 2017) report found that 31% of the industry is not using practices and principles that are widely considered to be necessary for accelerating technology transformations, such as continuous integration and continuous delivery, Lean practices, and a collaborative culture (i.e., capabilities championed by the DevOps movement). However, we also know that technology and software transformations are imperative in organizations today. A recent Gartner study found that 47% of CEOs face pressure from their board to digitally transform (Panetta 2017).

Within organizations, technology transformation journeys are at different stages, and reports suggest there is more work to be done than many of us currently believe. Another Forrester report states that DevOps is accelerating technology, but that

organizations often overestimate their progress (Klavens et al. 2017). Furthermore, the report points out that executives are especially prone to overestimating their progress when compared to those who are actually doing the work.

These findings about the disconnect between executive and practitioner estimates of DevOps maturity highlight two considerations that are often missed by leaders. First, if we assume the estimates of DevOps maturity or capabilities from practitioners are more accurate—because they are closer to the work—the potential for value delivery and growth within organizations is much greater than executives currently realize. Second, the disconnect makes clear the need to measure DevOps capabilities accurately and to communicate these measurement results to leaders, who can use them to make decisions and inform strategy about their organization's technology posture.

## FOCUS ON CAPABILITIES, NOT MATURITY

Technology leaders need to deliver software quickly and reliably to win in the market. For many companies, this requires significant changes to the way we deliver software. The key to successful change is measuring and understanding the right things with a focus on capabilities—not on maturity.

While maturity models are very popular in the industry, we cannot stress enough that maturity models are not the appropriate tool to use or mindset to have. Instead, shifting to a capabilities model of measurement is essential for organizations

wanting to accelerate software delivery. This is due to four factors.

First, maturity models focus on helping an organization "arrive" at a mature state and then declare themselves done with their journey, whereas technology transformations should follow a continuous improvement paradigm. Alternatively, capability models focus on helping an organization continually improve and progress, realizing that the technological and business landscape is ever-changing. The most innovative companies and highest-performing organizations are always striving to be better and never consider themselves "mature" or "done" with their improvement or transformation journey—and we see this in our research.

Second, maturity models are quite often a "lock-step" or linear formula, prescribing a similar set of technologies, tooling, or capabilities for every set of teams and organizations to progress through. Maturity models assume that "Level 1" and "Level 2" look the same across all teams and organizations, but those of us who work in technology know this is not the case. In contrast, capability models are multidimensional and dynamic, allowing different parts of the organization to take a customized approach to improvement, and focus on capabilities that will give them the most benefit based on their current context and their short and long-term goals. Teams have their own context, their own systems, their own goals, and their own constraints, and what we should focus on next to accelerate our transformation depends on those things.

Third, capability models focus on key outcomes and how the capabilities, or levers, drive improvement in those outcomes—

that is, they are outcome based. This provides technical leadership with clear direction and strategy on high-level goals (with a focus on capabilities to improve key outcomes). It also enables team leaders and individual contributors to set improvement goals related to the capabilities their team is focusing on for the current time period. Most maturity models simply measure the technical proficiency or tooling install base in an organization without tying it to outcomes. These end up being vanity metrics: while they can be relatively easy to measure, they don't tell us anything about the impact they have on the business.

Fourth, maturity models define a static level of technological, process, and organizational abilities to achieve. They do not take into account the ever-changing nature of the technology and business landscape. Our own research and data have confirmed that the industry is changing: what is good enough and even "high-performing" today is no longer good enough in the next year. In contrast, capability models allow for dynamically changing environments and allow teams and organizations to focus on developing the skills and capabilities needed to remain competitive.

By focusing on a capabilities paradigm, organizations can continuously drive improvement. And by focusing on the *right* capabilities, organizations can drive improvements in their outcomes, allowing them to develop and deliver software with improved speed and stability. In fact, we see that the highest performers do exactly this, continually reaching for gains year over year and never settling for yesterday's accomplishments.

# EVIDENCE-BASED TRANSFORMATIONS FOCUS ON KEY CAPABILITIES

Within both capability and maturity model frameworks, there are disagreements about *which* capabilities to focus on. Product vendors often favor capabilities that align with their product offerings. Consultants favor capabilities that align with their background, their offering, and their homegrown assessment tool. We have seen organizations try to design their own assessment models, choose solutions that align with the skill sets of internal champions, or succumb to analysis paralysis because of the sheer number of areas that need improvement in their organization.

A more guided, evidence-based solution is needed, and the approach discussed in this book describes such a solution.

Our research has yielded insights into what enables both software delivery performance and organizational performance as seen in profitability, productivity, and market share. In fact, our research shows that none of the following often-cited factors predicted performance:

- age and technology used for the application (for example, mainframe "systems of record" vs. greenfield "systems of engagement")
- whether operations teams or development teams performed deployments
- whether a change approval board (CAB) is implemented

The things that *do* make a difference in the success of software

delivery and organizational performance are those that the highest performers and most innovative companies use to get ahead. Our research has identified 24 key capabilities that drive improvement in software delivery performance and, in turn, organizational performance. These capabilities are easy to define, measure, and improve.[1] This book will get you started on defining and measuring these capabilities. We will also point you to some fantastic resources for improving them, so you can accelerate your own technology transformation journey.

## THE VALUE OF ADOPTING DEVOPS

You may be asking yourself: How do we know that these capabilities are drivers of technology and organizational performance, and why can we say it with such confidence?

The findings from our research program show clearly that the value of adopting DevOps is even larger than we had initially thought, and the gap between high and low performers continues to grow.

We discuss how we measure software delivery performance and how our cohort performs in detail in the following chapter. To summarize, in 2017 we found that, when compared to low performers, the high performers have:

- 46 times more frequent code deployments
- 440 times faster lead time from commit to deploy
- 170 times faster mean time to recover from downtime
- 5 times lower change failure rate (1/5 as likely for a change

to fail)

When compared to the 2016 results, the gap between high performers and low performers narrowed for tempo (deployment frequency and change lead time) and widened for stability (mean time to recover and change failure rate). We speculate that this is due to low-performing teams working to increase tempo but not investing enough in building quality into the process. The result is larger deployment failures that take more time to restore service. High performers understand that they don't have to trade speed for stability or vice versa, because by building quality in they get both.

You may be wondering: How do high-performing teams achieve such amazing software delivery performance? They do this by turning the right levers—that is, by improving the right capabilities.

Over our four-year research program we have been able to identify the capabilities that drive performance in software delivery and impact organizational performance, and we have found that they work for all types of organizations. Our research investigated organizations of all sizes, in all industries, using legacy and greenfield technology stacks around the world—so the findings in this book will apply to the teams in your organization too.

[1]These 24 capabilities are listed, along with a pointer to the chapter that discusses them, in Appendix A.