

THE SCIENCE OF LEAN SOFTWARE AND DEVOPS

ACCELERATE

Building and Scaling High Performing
Technology Organizations



Nicole Forsgren, PhD
Jez Humble, *and* Gene Kim

*with forewords by Martin Fowler and Courtney Kissler
and a case study contributed by Steve Bell and Karen Whitley Bell*

Praise for Accelerate

“This is the kind of foresight that CEOs, CFOs, and CIOs desperately need if their company is going to survive in this new software-centric world.

Anyone that doesn’t read this book will be replaced by someone who has.”

— Thomas A. Limoncelli, coauthor of
The Practice of Cloud System Administration

“Here, do this!’ The evidence presented in Accelerate is a triumph of research, tenacity, and insight, proving not just correlation but a causal link between good technical and management behaviors and business performance. It also exposes the myth of ‘maturity models’ and offers a realistic, actionable alternative. As an independent consultant working at the intersection of people, technology, process, and organization design this is manna from heaven!

As chapter 3 concludes: ‘You can act your way to a better culture by implementing these practices in technology organizations’ [emphasis mine]. There is no mystical culture magic, just 24 concrete, specific capabilities that will lead not only to better business results, but more importantly to happier, healthier, more motivated people and an organization people want to work at. I will be giving copies of this book to all my clients.”

— Dan North, independent technology and organization consultant

“Whether they recognize it or not, most organizations today are in the business of software development in one way, shape, or form. And most are being dragged down by slow lead times, buggy output, and complicated features that add expense and frustrate users. It doesn’t need to be this way. Forsgren, Humble, and Kim shine a compelling light on the what, why, and how of DevOps so you, too, can experience what outstanding looks and feels like.”

— Karen Martin, author of
Clarity First and The Outstanding Organization

“Accelerate does a fantastic job of explaining not only what changes organizations should make to improve their software delivery performance, but also the why, enabling people at all levels to truly understand how to level up their organizations.”

—Ryn Daniels, Infrastructure Operations Engineer at Travis CI
and author of *Effective DevOps*

“The ‘art’ of constructing a building is a well-understood engineering practice nowadays. However, in the software world, we have been looking for patterns and practices that can deliver the same predictable and reliable results whilst minimising waste and producing the increasingly high performance our businesses demand.

Accelerate provides research-backed, quantifiable, and real-world principles to create world-class, high-performing IT teams enabling amazing business outcomes.

Backed by the two leading thought leaders (Kim and Humble) in the DevOps community and world-class research from PhD Forsgren, this book is a highly recommended asset!”

—Jonathan Fletcher, Group CTO, Hiscox

“In their book Accelerate, Nicole Forsgren, Jez Humble, and Gene Kim don’t break any new conceptual ground regarding Agile, Lean, and DevOps. Instead, they provide something that might be even more valuable, which is a look inside the methodological rigor of their data collection and analysis approach which led them to their earlier conclusions on the key capabilities that make IT organizations better contributors to the business. This is a book that I will gladly be placing on my bookshelf next to the other great works by the authors.”

—Cameron Haight, VP and CTO, Americas, VMware

“The organizations that thrive in the future will be those that leverage digital technologies to improve their offerings and operations. Accelerate summarizes the best metrics, practices, and principles to use for improving software delivery and digital product performance, based on years of well-documented research. We strongly recommend this book to anyone involved in a digital transformation for solid guidance about what works, what doesn’t work, and what doesn’t matter.”

—Tom Poppendieck and Mary Poppendieck, authors of
the Lean Software Development series of books

“With this work, the authors have made a significant contribution to the understanding and application of DevOps. They show that when properly understood, DevOps is more than just a fad or a new name for an old concept. Their work illustrates how DevOps can improve the state of the art in organizational design, software development culture, and systems architecture. And beyond merely showing, they advance the DevOps community’s qualitative findings with research-based insights that I have heard from no other source.”

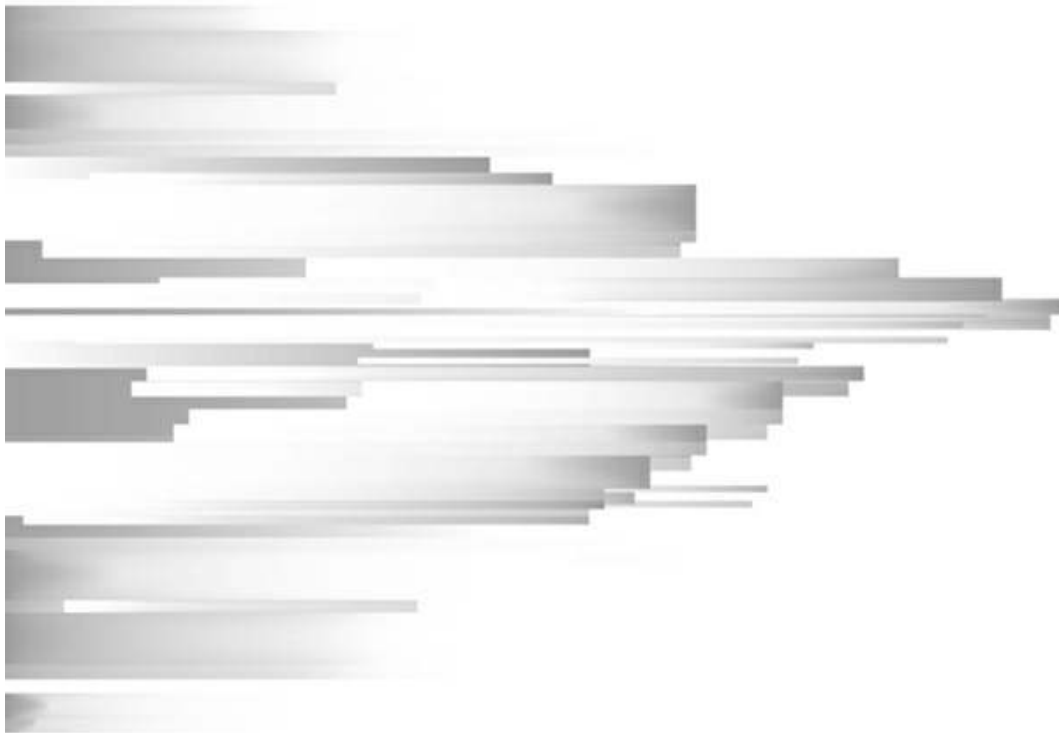
—Baron Schwartz, Founder and CEO of VividCortex
and coauthor of *High Performance MySQL*

ACCELERATE

THE SCIENCE BEHIND DEVOPS

ACCELERATE

Building and Scaling High Performing
Technology Organizations



Nicole Forsgren, PhD
Jez Humble *and* Gene Kim

IT Revolution
Portland, Oregon



25 NW 23rd Pl, Suite 6314
Portland, OR 97210

Copyright © 2018 by Nicole Forsgren, Jez Humble, and Gene Kim.
Chapter 16 Copyright © 2018 by Karen Whitley Bell and Steve Bell,
Lean IT Strategies, LLC.

All rights reserved, for information about permission
to reproduce selections from this book, write to Permissions, IT Revolution Press,
LLC, 25 NW 23rd Pl, Suite 6314, Portland, OR 97210

First Edition
Printed in the United States of America
22 21 20 19 18 1 2 3 4 5 6

Cover and book design by Devon Smith Creative, LLC

Library of Congress Catalog-in-Publication Data is available upon request.

ISBN: 978-1942788331
eBook ISBN: 978-194278355
Kindle ISBN: 978-194278362
Web PDF ISBN: 978-194278379

Publisher's note to readers: Although the authors and publisher have made every effort to ensure that the information in this book is correct, the authors and publisher do not assume and hereby disclaim any liability to any party for any loss, damage, or disruption caused by errors or omissions, whether such errors or omissions result from negligence, accident, or any other cause.

For information about special discounts for bulk purchases or for information on booking authors for an event, please visit our website at www.ITRevolution.com.

ACCELERATE

Contents

Foreword by Martin Fowler

Foreword by Courtney Kissler

Quick Reference: Capabilities to Drive Improvement

Preface

Part I: What We Found

- 1 Accelerate
- 2 Measuring Performance
- 3 Measuring and Changing Culture
- 4 Technical Practices
- 5 Architecture
- 6 Integrating Infosec into the Delivery Lifecycle
- 7 Management Practices for Software
- 8 Product Development
- 9 Making Work Sustainable
- 10 Employee Satisfaction, Identity, and Engagement
- 11 Leaders and Managers

Part II: The Research

- 12 The Science Behind This Book
- 13 Introduction to Psychometrics
- 14 Why Use a Survey
- 15 The Data for the Project

Part III: Transformation

- 16 High-Performance Leadership and Management

Conclusion

Appendix A: Capabilities to Drive Improvement

Appendix B: The Stats

Appendix C: Statistical Methods Used in Our Research

Acknowledgments

Bibliography

Index

About the Authors

Figures

- 2.1 Software Delivery Performance
- 2.2 Year over Year Trends: Tempo
- 2.3 Year over Year Trends: Stability
- 2.4 Impacts of Software Delivery Performance
- 3.1 Likert-Type Questions for Measuring Culture
- 3.2 Westrum Organizational Culture's Outcomes
- 3.3 Westrum Organizational Culture's Drivers
- 4.1 Drivers of Continuous Delivery
- 4.2 Impacts of Continuous Delivery
- 4.3 Continuous Delivery Makes Work More Sustainable
- 4.4 New Work vs. Unplanned Work
- 5.1 Deploys per Developer per Day
- 7.1 Components of Lean Management
- 7.2 Impacts of Lean Management Practices
- 8.1 Components of Lean Product Management
- 8.2 Impacts of Lean Product Management
- 9.1 Impacts of Technical and Lean Practices on Work Life
- 10.1 Impacts of Technical and Lean Practices on Identity
- 10.2 Impacts of Technical and Lean Practices on Job Satisfaction
- 10.3 Gender Demographics in 2017 Study
- 10.4 Underrepresented Minority Demographics in 2017 Study
- 11.1 Impacts of Transformational Leadership on Technical and Lean Capabilities
- 12.1 Spurious Correlation: Per Capita Cheese Consumption and Strangulation by Bedsheets
- 16.1 Leadership Obeya (360-Degree Panorama)
- 16.2 ING's New Agile Organizational Model Has No Fixed Structure—It Constantly Evolves
- 16.3 Stand-up and Catchball Rhythm

- 16.4 High-Performance Team, Management, and Leadership Behaviors and Practices
 - A.1 Overall Research Program
 - B.1 Firmographics: Organization Size, Industry, Number of Servers in 2017

Tables

- 2.1 Design vs. Delivery
- 2.2 Software Delivery Performance for 2016
- 2.3 Software Delivery Performance for 2017
- 3.1 Westrum's Typology of Organizational Culture
- 13.1 Westrum's Typology of Organizational Culture
- B.1 Manual Work Percentages

FOREWORD

By Martin Fowler

A few years ago I read a report that said, “We can now assert with confidence that high IT performance correlates with strong business performance, helping to boost productivity, profitability, and market share.” When I read something like that, my usual response is to toss it with great force into the rubbish bin, because that’s usually a tell for some bogus bullshit masquerading as science. I hesitated this time, however, for this was the “2014 State of DevOps Report.” One of its authors was Jez Humble, a colleague and friend who I knew was equally allergic to this kind of twaddle. (Although I have to confess that another reason for not tossing it was that I was reading it on my iPad.)

So, instead I emailed Jez to find out what lay behind this statement. A few weeks later I was on a call with him and Nicole Forsgren, who patiently walked me through the reasoning. While I’m no expert on the methods they used, she said enough to convince me there was some real analysis going on here, far more than I usually see, even in academic papers. I followed the subsequent State of DevOps reports with interest, but also with growing frustration. The reports gave the results of their work but never contained the explanation that Nicole walked through with me on the phone. This greatly undermined their credibility, as there was little evidence that these reports were based on more than speculation. Finally, those of us that had seen behind the curtains convinced Nicole, Jez, and Gene to reveal their methods by writing this book. For me, it’s been a long wait, but I’m glad I now have something that I can genuinely recommend as a way to look at IT delivery

effectiveness—one that's based on more than a few analysts' scattered experiences.

The picture they paint is compelling. They describe how effective IT delivery organizations take about an hour to get code from “committed to mainline” to “running in production,” a journey lesser organizations take months to do. They, thus, update their software many times a day instead of once every few months, increasing their ability to use software to explore the market, respond to events, and release features faster than their competition. This huge increase in responsiveness does not come at a cost in stability, since these organizations find their updates cause failures at a fraction of the rate of their less-performing peers, and these failures are usually fixed within the hour. Their evidence refutes the bimodal IT notion that you have to choose between speed and stability—instead, speed depends on stability, so good IT practices give you both.

So, as you may expect, I'm delighted that they've put this book into production, and I will be recommending it willy-nilly over the next few years. (I've already been using many bits from its drafts in my talks.) However, I do want to put in a few notes of caution. They do a good job of explaining why their approach to surveys makes them a good basis for their data. However, they are still surveys that capture subjective perceptions, and I wonder how their population sample reflects the general IT world. I'll have more confidence in their results when other teams, using different approaches, are able to confirm their reasoning. The book already has some of this, as the work done by Google on team cultures provides further evidence to support their judgment on how important a Westrum-generative organizational culture is for effective software teams. Such further work would also make me less concerned that their conclusions confirm much of my advocacy—confirmation bias is a strong force (although I mostly notice it in others ;-)). We should also remember that their book focuses on IT delivery, that is, the journey from commit to production, not the entire software development process.

But these quibbles, while present, shouldn't distract us from the main thrust of this book. These surveys, and the careful analysis done on them, provide some of the best justification around for practices that can significantly improve most IT organizations. Anyone running an IT group should take a good hard look at these techniques and work to use them to improve their practice. Anyone working with an IT group, either internally or from an IT delivery company like ours, should look for these practices in place and a steady program of continuous improvement to go with them. Forsgren, Humble, and Kim have laid out a picture of what effective IT looks like in 2017, and IT practitioners should be using this as a map to join the high performers.

*Martin Fowler
Chief Scientist, ThoughtWorks*

FOREWORD

By Courtney Kissler

My journey started in the summer of 2011. I was working at Nordstrom and we had made a strategic decision to focus on digital as the growth engine. Up until that point, our IT organization was optimized for cost; I shared in my DevOps Enterprise Summit 2014 presentation that one of my “aha” moments was the shift to optimizing for speed. I made a lot of mistakes along the way and wish I had access to the information in this book back then. Common traps were stepped in—like trying a top-down mandate to adopt Agile, thinking it was one size fits all, not focusing on measurement (or the right things to measure), leadership behavior not changing, and treating the transformation like a program instead of creating a learning organization (never done).

Throughout the journey, the focus was moving to outcome-based team structures, knowing our cycle time (by understanding our value stream map), limiting the blast radius (starting with one to two teams vs. boiling the ocean), using data to drive actions and decisions, acknowledging that work is work (don’t have a backlog of features and a backlog of technical debt and a backlog of operational work; instead, have a single backlog because NFRs are features and reducing technical debt improves stability of the product). None of this happened overnight, and it took a lot of experimentation and adjusting along the way.

What I know to be true based on my experience is that adopting the guidance in this book *will* make your organization higher performing. It works for all types of software delivery and is methodology agnostic. I have personally experienced it and have multiple examples of applying

these practices within mainframe environments, traditional packaged software application delivery teams, and product teams. It can work across the board. It takes discipline, persistence, transformational leadership, and a focus on people. After all, people are an organization's #1 asset, but so often that is not how organizations operate. Even though the journey will not be easy, I can say that it is definitely worth it, and not only will you see better results, your team will be happier. As an example, when we started measuring eNPS, the teams practicing these techniques had the highest scores throughout our technology organization.

Another thing I learned along the way is how critical it is to have senior leadership support. And support in actions, not words. Senior leaders need to demonstrate their commitment to creating a learning organization. I will share the behaviors I try to model with my teams. I believe passionately in honoring and extracting reality. If I am a senior leader and my team doesn't feel comfortable sharing risks, then I will never truly know reality. And, if I'm not genuinely curious and only show up when there's a failure, then I am failing as a senior leader. It's important to build trust and to demonstrate that failure leads to inquiry (see the Westrum model in this book).

You will encounter skeptics along the way. I heard things like "DevOps is the new Agile," "Lean doesn't apply to software delivery," "Of course this worked for the mobile app team. They are a unicorn." When I encountered the skeptics, I attempted to use external examples to influence the discussion. I leveraged mentors along the way—without them, it would have been challenging to stay focused. Having the information in this book would have been extremely helpful and I strongly encourage you to use it within your organization. I have spent most of my career in retail; in that industry, it has become more and more critical to evolve, and shipping software is now part of the DNA of every organization. Don't ignore the science outlined in this book. It will help you accelerate your transformation to a high-performing technology organization.

Courtney Kissler
VP Digital Platform Engineering, Nike

QUICK REFERENCE:

CAPABILITIES TO DRIVE

IMPROVEMENT

Our research has uncovered 24 key capabilities that drive improvements in software delivery performance. This reference will point you to them in the book. A detailed guide is found in Appendix A. They are presented in no particular order.

The capabilities are classified into five categories:

- Continuous delivery
- Architecture
- Product and process
- Lean management and monitoring
- Cultural

CONTINUOUS DELIVERY CAPABILITIES

1. Version control: Chapter 4
2. Deployment automation: Chapter 4
3. Continuous integration: Chapter 4
4. Trunk-based development: Chapter 4
5. Test automation: Chapter 4
6. Test data management: Chapter 4

7. Shift left on security: Chapter 6
8. Continuous delivery (CD): Chapter 4

ARCHITECTURE CAPABILITIES

9. Loosely coupled architecture: Chapter 5
10. Empowered teams: Chapter 5

PRODUCT AND PROCESS CAPABILITIES

11. Customer feedback: Chapter 8
12. Value stream: Chapter 8
13. Working in small batches: Chapter 8
14. Team experimentation: Chapter 8

LEAN MANAGEMENT AND MONITORING CAPABILITIES

15. Change approval processes: Chapter 7
16. Monitoring: Chapter 7
17. Proactive notification: Chapter 13
18. WIP limits: Chapter 7
19. Visualizing work: Chapter 7

CULTURAL CAPABILITIES

20. Westrum organizational culture: Chapter 3
21. Supporting learning: Chapter 10
22. Collaboration among teams: Chapters 3 and 5
23. Job satisfaction: Chapter 10
24. Transformational leadership: Chapter 11