

The **DevOps** Handbook

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

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

FEATURING NEW FOREWORD AND UPDATED MATERIALS FROM NICOLE FORSGREN, PHD

ORIGINAL FOREWORD BY JOHN ALLSPAW

IT Revolution Portland, Oregon



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

The DevOps Handbook, Second Edition © 2021 by Gene Kim, Matthew "Jez" Humble, Patrick Debois, John Willis, and Nicole Forsgren

First edition © 2016 by Gene Kim, Jez Humble, Patrick Debois, and John Willis Printed in the United States of America

27 26 25 24 23 22 2 3 4 5 6 7 8 9 10

Cover design by Devon Smith Creative Cover illustration by eboy Book design by Devon Smith Creative

ISBN: 9781950508402 eBook ISBN: 9781950508433 Web PDF ISBN: 9781942788867 Audio ISBN: 9781950508440

The author of the 18F case study has dedicated the work to the public domain by waiving all of his or her rights to the work worldwide under copyright law, including all related and neighboring rights, to the extent allowed by law. You can copy, modify, distribute, and perform case study 18F, even for commercial purposes, all without asking permission.

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

THE DEVOPS HANDBOOK, SECOND EDITION

CONTENTS

Х
iii
vii
ix
xi
ix

Part I—The Three Ways

Part I Introduction	3
01 Agile, Continuous Delivery, and the Three Ways	7
NEW Case Study: Approaching Cruising Altitude:	
American Airlines' DevOps Journey (Part 1) (2020)	15
02 The First Way: The Principles of Flow	19
NEW Case Study: Flow and Constraint Management in	
Healthcare (2021)	29
03 The Second Way: The Principles of Feedback	33
NEW Case Study: Pulling the Andon Cord at	
Excella (2018)	39
04 The Third Way: The Principles of Continual Learning and	
Experimentation	45
NEW Case Study: The Story of Bell Labs (1925)	54
Part 1 Conclusion	57

Part II—Where to Start

Part II Introduction	61
05 Selecting Which Value Stream to Start With	63
Case Study: Nordstrom's DevOps Transformation	
(2014–2015)	63
NEW Case Study: Kessel Run: The Brownfield	
Transformation of a Mid-Air Refueling System (2020)	69
NEW Case Study: Scaling DevOps Across the Business:	
American Airlines' DevOps Journey (Part 2) (2020)	74

	NEW Case Study: Saving the Economy From Ruin (With	
	a Hyperscale PaaS) at HMRC (2020)	77
06 1	Understanding the Work in Our Value Stream, Making it	
7	Visible, and Expanding it Across the Organization	81
	Case Study: Nordstrom's Experience with Value Stream	
	Mapping (2015)	81
	Case Study: Operation InVersion at LinkedIn (2011)	91
07 1	How to Design Our Organization and Architecture	
7	with Conway's Law in Mind	97
	Case Study: Conway's Law at Etsy (2015)	98
	Case Study: API Enablement at Target (2015)	112
08]	How to Get Great Outcomes by Integrating Operations	
i	into the Daily Work of Development	115
	Case Study: Big Fish Games (2014)	115
	NEW Case Study: Better Ways of Working at Nationwide	
	Building Society (2020)	124
]	Part II Conclusion	129

Part III—The First Way: The Technical Practices of Flow

Part I	II Introduction	133
09 Creat	e the Foundations of Our Deployment Pipeline	135
Ca	se Study: Enterprise Data Warehouse (2009)	135
NE	W Case Study: How a Hotel Company Ran \$30B of	
]	Revenue in Containers (2020)	143
10 Enabl	e Fast and Reliable Automated Testing	147
Ca	se Study: Google Web Server (2005)	148
11 Enabl	e and Practice Continuous Integration	167
Ca	se Study: HP LaserJet Firmware (2006)	168
Ca	se Study: Continuous Integration of Bazaarvoice	
((2012)	173
12 Automate and Enable Low-Risk Releases		177
Ca	se Study: Daily Deployments at CSG International	
((2013)	181
Ca	se Study: Etsy—Self-Service Developer Deployment:	
1	An Example of Continuous Deployment (2014)	186
Ca	se Study: Dixons Retail—Blue-Green Deployment for	
]	Point-of-Sale System (2008)	193
Ca	se Study: Dark Launch of Facebook Chat (2008)	198

NEW Case Study: Creating a Win-Win for Dev & Ops	
at CSG (2016)	201
13 Architect for Low-Risk Releases	207
Case Study: Evolutionary Architecture at Amazon (2002)	212
Case Study: Strangler Fig Pattern at Blackboard Learn	
(2011)	215
Part III Conclusion	219

Part IV—The Second Way: The Technical Practices of Feedback

223
225
226
237
245
245
251
255
259
259
269
273
273
278
281
281
286
290

Case Study: Pair Programming Replacing Broken Code	
Review Processes at Pivotal Labs (2011)	293
Part IV Conclusion	299

Part V—The Third Way: The Technical Practices of Continual Learning and Experimentation

Part V Introduction	303
19 Enable and Inject Learning into Daily Work	305
Case Study: AWS US-East and Netflix (2011)	305
NEW Case Study: Turning an Outage into a Powerful	
Learning Opportunity at CSG (2020)	318
20 Convert Local Discoveries into Global Improvements	321
Case Study: Standardizing a New Technology Stack	
at Etsy (2010)	332
NEW Case Study: Crowdsourcing Technology Governance	
at Target (2018)	333
21 Reserve Time to Create Organizational Learning	
and Improvement	335
Case Study: Thirty-Day Challenge at Target (2015)	335
Case Study: Internal Technology Conferences at	
Nationwide Insurance, Capital One, and Target (2014)	342
Part V Conclusion	347

Part VI—The Technological Practices of Integrating Information Security, Change Management, and Compliance

Part VI Introduction	351
22 Information Security Is Everyone's Job Every Day	353
Case Study: Static Security Testing at Twitter (2009)	360
Case Study: 18F Automating Compliance for the Federal	
Government with Compliance Masonry (2016)	369
Case Study: Instrumenting the Environment at	
Etsy (2010)	373
NEW Case Study: Shifting Security Left at Fannie	
Mae (2020)	376
23 Protecting the Deployment Pipeline	379

Case Study: Automated Infrastructure Changes as	
Standard Changes at Salesforce.com (2012)	383
Case Study: PCI Compliance and a Cautionary Tale of	
Separating Duties at Etsy (2014)	385
NEW Case Study: Biz and Tech Partnership toward	
Ten "No Fear Releases" Per Day at Capital One (2020)	387
Case Study: Proving Compliance in Regulated	
Environments (2015)	389
Case Study: Relying on Production Telemetry for	
ATM Systems (2013)	392
Part VI Conclusion	395
A Call to Action: Conclusion to <i>The DevOps Handbook</i>	397
Afterword to the Second Edition	401
Appendices	409
Bibliography	423
Notes	441
Index	461
Acknowledgments	481
About the Authors	484

FIGURES & TABLES

Table 0.1.	The Ever Accelerating Trend toward Easter Cheaper	
	Lower Risk Delivery of Software	xxxi
Figure 0.1:	Deployments per Day vs. Number of Developers	xli
Figure 1.1:	Lead Time vs. Process Time of a Deployment Operation	9
Figure 1.2:	A Technology Value Stream with a Deployment Lead Time	
0	of Three Months	10
Figure 1.3:	A Technology Value Stream with a Lead Time of Minutes	11
Figure 1.4:	The Three Ways	13
Figure 1.5:	American Airlines' DevOps Transformation Journey	17
Figure 2.1:	An Example Kanban Board Spanning Requirements, Dev,	
-	Test, Staging, and In Production	20
Figure 2.2:	Simulation of "Envelope Game"	24
Figure 3.1:	Feedback Cycle Times	37
Figure 3.2:	Cycle Time vs. Andon Pulls at Excella	41
Table 4.1:	The Westrum Organizational Typology Model	48
Figure 5.1:	The Technology Adoption Curve	72
Figure 5.2:	American Airlines' Delivery Transformation	75
Table 5.1:	American Airlines' New Vocabulary	77
Figure 6.1:	An Example Value Stream Map	85
Figure 6.2:	Invest 20% of Capacity in those Who Create Positive,	
	User-Invisible Value	90
Figure 7.1:	Functional vs. Market Orientation	103
Table 7.1:	Specialists vs. Generalists vs. "E-shaped" Staff	106
Figure 8.1:	Functional Teams in Silos vs. Long-Lived, Multi-	
	Skilled Teams	126
Figure 10.1:	The Deployment Pipeline	152
Figure 10.2:	The Ideal and Non-Ideal Automated Testing Pyramids	157
Figure 10.3:	Running Automated and Manual Tests in Parallel	158
Figure 12.1:	Number of Developers Deploying per Week at Facebook	178
Figure 12.2:	Daily Deployments at CSG International	183
Figure 12.3:	Elite and High Performers Achieve Faster Deployment	
	Lead Times and MTTR (2019)	186
Figure 12.4:	The Deployinator Console at Etsy	188

Figure 12.5:	Blue-Green Deployment Patterns	191
Figure 12.6:	The Canary Release Pattern	194
Figure 12.7:	How Structure Influences Behavior and Quality	202
Figure 12.8:	From Siloed Approach to Cross-Functional Teams	203
Figure 12.9:	Conventional vs. Cross-Functional Structure	204
Figure 13.1:	Google Cloud Datastore	209
Table 13.1:	Architectural Archetypes	211
Figure 13.2:	Blackboard Learn Code Repository: Before Building Blocks	216
Figure 13.3:	Blackboard Learn Code Repository: After Building Blocks	217
Figure 14.1:	Incident Resolution Time for Elite, High, Medium, and	
	Low Performers (2019)	228
Figure 14.2:	Monitoring Framework	230
Figure 14.3:	One Line of Code to Generate Telemetry using StatsD and	
	Graphite at Etsy	235
Figure 14.4:	User Excitement of New Features in User Forum Posts after	
	Deployments	241
Figure 15.1:	Standard Deviations (σ)Mean (μ) with Gaussian	
	Distribution	247
Figure 15.2:	Downloads per Minute: Over-Alerting when Using	
	"Three Standard Deviation" Rule	250
Figure 15.3:	Downloads per Minute: Histogram of Data Showing Non-	
	Gaussian Distribution	251
Figure 15.4:	Netflix Customer Viewing Demand for Five Days	252
Figure 15.5:	Netflix Scryer Forecasting Customer Traffic and the	
	Resulting AWS Schedule of Computer Resources	252
Figure 15.6:	Autodesk Share Price and Thirty-Day Moving Average	
	Filter	254
Figure 15.7:	Transaction Volume: Under-Alerting Using "Three	
	Standard Deviation" Rule	255
Figure 15.8:	Transaction Volume: Using Kolmogorov-Smirnov Test	
	to Alert on Anomalies	256
Figure 16.1:	Deployment to Etsy.com Causes PHP Run-Time Warnings	
	and Is Quickly Fixed	262
Figure 16.2:	The "Service Handback" at Google	268
Figure 16.3:	The LRR and HRR at Google	271
Figure 18.1:	Comments and Suggestions on a GitHub Pull Request	282
Figure 18.2:	Size of Change vs. Lead Time for Reviews at Google	291
Figure 21.1:	The ASREDS Learning Loop	341

Figure 22.1:	Jenkins Running Automated Security Testing	
Figure 22.2:	Number of Brakeman Security Vulnerabilities Detected	
Figure 22.3:	Time to Remediate vs. Time to Update Dependencies	
	(TTU)	365
Figure 22.4:	Five Behavioral Clusters for Open-Source Projects	366
Figure 22.5:	Developers See SQL Injection Attempts in Graphite at Etsy	374
Figure AF.1:	Average Development Window by Day of Week per User	402
Figure A.1:	The Core, Chronic Conflict Facing Every IT Organization	412
Table A.1:	The Downward Spiral	413
Figure A.2:	Queue Size and Wait Times as Function of Percent	
	Utilization	415
Table A.2:	Two Stories	416
Figure A.3:	The Toyota Andon Cord	417

Cheaper, Lower Risk Delivery of Software				
	1970s-1980s	1990s	2000s-Present	
Era	Mainframes	Client/Server	Commoditization and Cloud	
Representative technology of era	COBOL, DB2 on MVS, etc.	C++, Oracle, Solaris, etc.	Java, MySQL, Red Hat, Ruby on Rails, PHP, etc.	
Cycle time	1–5 years	3–12 months	2–12 weeks	
Cost	\$1M-\$100M	\$100k-\$10M	\$10k-\$1M	
At risk	The whole company	A product line or division	A product feature	
Cost of failure	Bankruptcy, sell the company, massive layoffs	Revenue miss, CIO's job	Negligible	

Table 0.1: The Ever-Accelerating Trend toward Faster,Cheaper, Lower Risk Delivery of Software

Source: Adrian Cockcroft, "Velocity and Volume (or Speed Wins)," presentation at FlowCon, San Francisco, CA, November 2013.



Figure 0.1: Deployments per Day vs. Number of Developers

Only organizations that are deploying at least once per day are shown. Source: Puppet Labs, 2015 State Of DevOps Report.



Operation



Figure 1.2: A Technology Value Stream with a Deployment Lead Time of Three Months Source: Damon Edwards, "DevOps Kaizen," 2015.





Figure 1.4: The Three Ways Source: Gene Kim, "The Three Ways: The Principles Underpinning DevOps," ITRevolution.com (blog), August 22, 2012, http://itrevolution.com /the-three-ways-principles-underpinning-devops/.



Figure 1.5: American Airlines' DevOps Transformation Journey

Source: With permission of Ross Clanton



Figure 2.1: An Example Kanban Board Spanning Requirements, Dev, Test, Staging, and In Production Source: David J. Andersen and Dominica DeGrandis, *Kanban for IT Ops*, training materials for workshop, 2012.



Figure 2.2: Simulation of "Envelope Game"

(Fold, insert, seal, and stamp the envelope.) Source: Stefan Luyten, "Single Piece Flow," Medium.com, August 8, 2014, https:// medium.com/@stefanluyten/single-piece-flow-5d2c2bec845b.



Figure 3.1: Feedback Cycle Times

Source: Hendrickson, Elisabeth. "DOES15—Elisabeth Hendrickson—Its All About Feedback." Posted by DevOps Enterprise Summit, November 5, 2015. YouTube video, 34:47. https://www.youtube.com/watch?v=r2BFTXBundQ.



Figure 3.2: Cycle Time vs. Andon Pulls at Excella

Source: Zach Ayers and Joshua Cohen. "Andon Cords in Development Teams—Driving Continuous Learning," presentation at the DevOps Enterprise Summit Las Vegas, 2019. https://videolibrary.doesvirtual.com/?video=504281981.

Table 4.1: The Westrum OrganizationalTypology Model

How organizations process information.

Pathological Organization	Bureaucratic Organization	Generative Organization
Information is hidden	Information may be ignored	Information is actively sought
Messengers are "shot"	Messengers are tolerated	Messengers are trained
Responsibilities are shirked	Responsibilities are compartmented	Responsibilities are shared
Bridging between teams is discouraged	Bridging between teams allowed but discouraged	Bridging between teams is rewarded
Failure is covered up	Organization is just; merciful	Failure causes injury
New ideas are crushed	New ideas create problems	New ideas are welcomed

Source: Ron Westrum, "A typology of organisation culture," BMJ Quality & Safety 13, no. 2 (2004), doi:10.1136/qshc.2003.009522.









Figure 5.2: American Airlines' Delivery Transformation

Source: Maya Leibman and Ross Clanton, "DevOps: Approaching Cruising Altitude," presentation at DevOps Enterprise Summit-Virtual Las Vegas 2020, videolibrary.doesvirtual.com/?video=467488959.

Before	After
I want to create a pop-up to incentivize people to download the mobile app.	Fragile applications are prone to failure.
What did our competitors do?	What do our customers value?
When will this project be done?	When do we start seeing value?
What went wrong?	What did we learn and how can I help?
I want a completely new website.	What's the first thing we can try to experiment with this idea?

Table 5.1: American Airlines' New Vocabulary

Source: Maya Leibman and Ross Clanton, "DevOps: Approaching Cruising Altitude," presentation at DevOps Enterprise Summit-Virtual Las Vegas 2020, ideolibrary.doesvirtual.com/?video=467488959.



Figure 6.1: An Example of a Value Stream Map Source: Humble, Molesky, and O'Reilly, *Lean Enterprise*, 139.



Figure 6.2: Invest 20% of Capacity in those Who Create Positive, User-Invisible Value

Source: "Machine Learning and Technical Debt with D. Sculley," Software Engineering Daily podcast, November 17, 2015, http://softwareengineeringdaily .com/2015/11/17/machine-learning-and-technical-debt-with-d-sculley/.



Figure 7.1: Functional vs. Market Orientation

Left: Functional orientation: all work flows through centralized IT Operations. Right: Market orientation: all product teams can deploy their loosely coupled components self-service into production. Source: Humble, Molesky, and O'Reilly, *Lean Enterprise*, Kindle edition, 4523 & 4592.

Table 7.1: Specialists vs. Generalists vs. "E-Shaped" Staff(experience, expertise, exploration, and execution)

"I-shaped" (Specialists)	"T-shaped" (Generalists)	"E-shaped"
Deep expertise in one area	Deep expertise in one area	Deep expertise in a few areas
Very few skills or experience in other areas	Broad skills across many areas	Experience across many areas Proven execution skills Always innovating
Creates bottlenecks quickly	Can step up to remove bottlenecks	Almost limitless potential
Insensitive to downstream waste and impact	Sensitive to downstream waste and impact	
Prevents planning flexibility or absorption of variability	Helps make planning flexible and absorbs variability	

Source: Scott Prugh, "Continuous Delivery," ScaledAgileFramework.com, February 14, 2013, http://scaledagileframework.com/continuous-delivery/.



Figure 8.1: Functional Teams in Silos vs. Long-Lived, Multiskilled Teams

Source: Chapman, Janet, and Patrick Eltridge. "On A Mission: Nationwide Building Society," presentation at DevOps Enterprise Summit-Virtual London 2020. https://videolibrary.doesvirtual.com/?video=432109857.



Figure 10.1: The Deployment Pipeline

Source: Humble and Farley, Continuous Delivery, 3.

Ideal vs. Non-Ideal Testing Pyramids



Figure 10.2: The Ideal and Non-Ideal Automated Testing Pyramids Source: Martin Fowler, "TestPyramid," MartinFowler.com.



Parallel

Source: Humble and Farley, Continuous Delivery, Kindle edition, location 3868.



Figure 12.1: Number of Developers Deploying per Week at Facebook Source: Chuck Rossi, "Ship early and ship twice as often."



Figure 12.2: Daily Deployments at CSG International

Daily deployments and increasing release frequency resulted in a decrease in the number of production incidents and MTTR.

Source: "DOES15 - Scott Prugh & Erica Morrison—Conway & Taylor Meet the Strangler (v2.0)," YouTube video, 29:39, posted by DevOps Enterprise Summit, November 5, 2015, https://www.youtube.com/watch?v=tKdIHCL0DUg.



Figure 12.3: Elite and High Performers Achieve Faster Deployment Lead Times and MTTR (2019)¹ Source: Forsgren, et. al., Accelerate: State of DevOps Report, 2019.

¹ We have replaced this figure with an updated metric for the second edition. This new figure more accurately portrays experiences over the last five years.
Deploy to QA (Trunk)	Log		
Message:	Add log message:		
Push to QA \rightarrow	• [web] 2010-05-18 22:20:50 PRODUCTION sandrews Production deploy: old 25134, new; 25145 diff • [web] 2010-05-18 22:18:00 PRINCESS sandrews Princess Deploy: old: 25144, new:		
Princess is in the other castle	25145 diff • [web] 2010-05-18 22:17:22 QA sandrews kyles bug fix old: , new: 25145 diff • [web] 2010-05-18 22:12:03 QA sandrews pushing again banned user cache busting old: ,new: 25144 diff		
Save the Princess	 [web] 2010-05-18 22:08:39 [PHINCES5] sandrews [Princess Deploy: dd:25144, new: [web] 2010-05-18 22:05:35 [QA] sandrews [IIIIIIIIIIIIIII dd:25134, new: 25144 diff [web] 2010-05-18 20:56:50 [PRODUCTION cmunns Production deploy: eld 25134, new; 25144 diff [web] 2010-05-18 20:49:02 [PRODUCTION cmunns Production deploy: eld 25134, 		
Deployite Production	new; 25134 diff • [web] 2010-05-18 20:44:43 PRODUCTION cmunns Production deploy: old 25030, new; 25134 diff • [web] 2010-05-18 20:41:17 PRINCESS ahashim Princess Deploy: old: 25030, new: 9514		
	 (web) 2010-05-18 20:40:38 QA ahashim Penut butter jelly time!!!!!: old: 25030, new: 25134 (web) 2010-05-17 15:23:26 PRODUCTION sandrews Production deploy: old 24951, new; 25030 diff (web) 2010-05-17 15:08:05 PRINCESS sandrews Production deploy: old 24951, new; 25030 diff 		
	25030 diff		

Figure 12.4: The Deployinator Console at Etsy

Source: Erik Kastner, "Quantum of Deployment," CodeasCraft.com, May 20, 2010, https://codeascraft.com/2010/05/20/quantum-of-deployment/.



Figure 12.5: Blue-Green Deployment Patterns Source: Humble and North, *Continuous Delivery*, 261.



Figure 12.6: The Canary Release Pattern Source: Humble and Farley, *Continuous Delivery*, 263.



Problem management and multi-tiered support (L1 \rightarrow L2 \rightarrow L3) won't save you.





Figure 12.8: From Siloed Approach to Cross-Functional Teams Image courtesy of Scott Prugh.



Figure 12.9: Conventional vs. Cross-Functional Structure Image courtesy of Scott Prugh.



Figure 13.1: Google Cloud Datastore Source: Shoup, "From the Monolith to Micro-services."

	Pros	Pros Cons	
Monolithic v1 (all functionality in one application)	 Simple at first Low interprocess latencies Single code base; one deployment unit Resource efficient at small scales 	 Coordination overhead increases as team grows Poor enforcement of modularity Poor scaling All-or-nothing deploy (downtime, failures) Long build times 	
Monolithic v2 (sets of monolithic tiers: "front end presentation," "application server," "database layer")	 Simple at first Join queries are easy Single schema, deployment Resource efficient at small scales 	 Tendency for increased coupling over time Poor scaling and redundancy (all-ornothing, vertical only) Difficult to tune properly All-or-nothing schema management 	
Microservice (modular, independent, graph relationship vs. tiers, isolated persistence)	 Each unit is simple Independent scaling and performance Independent testing and deployment Can optimally tune performance (caching, replication, etc.) 	 Many cooperating units Many small repos Requires more sophisticated tooling and dependency management Network latencies 	

Table 13.1: Architectural Archetypes

Source: Shoup, "From the Monolith to Micro-services."



Figure 13.2: Blackboard Learn Code Repository: Before Building Blocks

Source: "DOES14—David Ashman—Blackboard Learn—Keep Your Head in the Clouds," YouTube video, 30:43, posted by DevOps Enterprise Summit 2014, October 28, 2014, https://www.youtube.com/watch?v=SSmixnMpsI4.



Figure 13.3: Blackboard Learn Code Repository: After Building Blocks

Source: "DOES14—David Ashman—Blackboard Learn—Keep Your Head in the Clouds," YouTube video, 30:43, posted by DevOps Enterprise Summit 2014, October 28, 2014, https://www.youtube.com/watch?v=SSmixnMpsI4.



Time to Restore Service





Figure 14.2: Monitoring Framework Source: Turnbull, *The Art of Monitoring*, Kindle edition, chap. 2.



Figure 14.3: One Line of Code to Generate Telemetry using StatsD and Graphite at Etsy Source: Ian Malpass, "Measure Anything, Measure Everything."



Figure 14.4: User Excitement of New Features in User Forum Posts after Deployments Source: Mike Brittain, "Tracking Every Release," CodeasCraft.com, December 8, 2010, https://codeascraft.com/2010/12/08/track-every-release/.





Source: Wikipedia, "Normal Distribution," https://en.wikipedia.org /wiki/Normal_distribution.



Figure 15.2: Downloads per Minute: Over-Alerting when Using "Three Standard Deviations" Rule

Source: Dr. Toufic Boubez, "Simple math for anomaly detection."



Figure 15.3: Downloads per Minute: Histogram of Data Showing Non-Gaussian Distribution Source: Dr. Toufic Boubez, "Simple math for anomaly detection."







(Source: Jacobson, Yuan, Joshi, "Scryer: Netflix's Predictive Auto Scaling Engine.")





Figure 15.7: Transaction Volume: Under-Alerting Using "Three Standard Deviations" Rule (Source: Dr. Toufic Boubez, "Simple math for anomaly detection.")



Figure 15.8: Transaction Volume: Using Kolmogorov-Smirnov Test to Alert on Anomalies Source: Dr. Toufic Boubez, "Simple math for anomaly detection."



Figure 16.1: Deployment to Etsy.com Causes PHP Run-Time Warnings and Is Quickly Fixed Source: Mike Brittain, "Tracking Every Release."



Figure 16.2: The "Service Handback" at Google

Source: "SRE@Google: Thousands of DevOps Since 2004," YouTube video, 45:57, posted by USENIX, January 12, 2012, https://www.youtube.com/watch?v=iIuTnhdTzK0.



Figure 16.3: The LRR and HRR at Google

Source: "SRE@Google: Thousands of DevOps Since 2004," YouTube video, 45:57, posted by USENIX, January 12, 2012, https://www.youtube.com /watch?v=iIuTnhdTzK0.



Figure 18.1: Comments and Suggestions on a GitHub Pull Request

Source: Scott Chacon, "GitHub Flow," ScottChacon.com, August 31, 2011, http:// scottchacon.com/2011/08/31/github-flow.html.



Figure 18.2: Size of Change vs. Lead Time for Reviews at Google Source: Ashish Kumar, "Development at the Speed and Scale of Google," presentation at QCon, San Francisco, CA, 2010, qconsf.com/sf2010/dl /qcon-sanfran-2010/slides/AshishKumar_DevelopingProductsattheSpeedand ScaleofGoogle.pdf.



Figure 21.1: The ASREDS Learning Loop Source: Smart et al., *Sooner Safer Happier: Antipatterns and Patterns for Business Agility* (Portland, OR: IT Revolution, 2020).

Jenkins							
S	W	Name	Last Success	Last Failure	Last Duration		
	۲	Static analysis scan	7 days 1 hr - #2	N/A	6.3 sec		
	-	Check known vulnerabilities in dependencies	N/A	7 days 1 hr - #2	1.6 sec		
	۲	Download and unit test	7 days 1 hr - #2	N/A	32 sec		
	۲	Scan with OWASP ZAP	7 days 1 hr - #2	N/A	4 min 43 sec		
	۲	Start	7 days 1 hr - #2	N/A	5 min 46 sec		
	۲	Virus scanning	7 days 1 hr - #2	N/A	4.7 sec		

Figure 22.1: Jenkins Running Automated Security Testing

Source: James Wicket and Gareth Rushgrove, "Battle-tested code without the battle," Velocity 2014 conference presentation, posted to Speakerdeck.com, June 24, 2014, https://speakerdeck.com/garethr/battle-tested-code-without-the-battle.



Figure 22.2: Number of Brakeman Security Vulnerabilities Detected





Source: Sonatype, 2019 Software Supply Chain Report.



Figure 22.4: Five Behavioral Clusters for Open-Source Projects Source: Sonatype, 2019 Software Supply Chain Report.



Figure 22.5: Developers See SQL Injection Attempts in Graphite at Etsy

Source: "DevOpsSec: Applying DevOps Principles to Security, DevOpsDays Austin 2012," SlideShare.net, posted by Nick Galbreath, April 12, 2012, http://www .slideshare.net/nickgsuperstar/devopssec-apply-devops-principles-to-security.



per User

Source: Forsgren et. al., 2020 State of the Octoverse.

APPENDICES

Appendix 1: The Convergence of DevOps

We believe that DevOps is benefiting from an incredible convergence of management movements, which are all mutually reinforcing and can help create a powerful coalition to transform how organizations develop and deliver IT products and services.

John Willis named this "the convergence of DevOps." The various elements of this convergence are described below in approximate chronological order. (Note that these descriptions are not intended to be exhaustive, but merely enough to show the progression of thinking and the rather improbable connections that led to DevOps.)

The Lean Movement

The Lean Movement started in the 1980s as an attempt to codify the Toyota Production System with the popularization of techniques such as value stream mapping, kanban boards, and total productive maintenance.¹

Two major tenets of Lean were the deeply held belief that lead time (i.e., the time required to convert raw materials into finished goods) was the best predictor of quality, customer satisfaction, and employee happiness; and that one of the best predictors of short lead times was small batch sizes, with the theoretical ideal being "single piece flow" (i.e., "1x1" flow: inventory of 1, batch size of 1).

Lean principles focus on creating value for the customer—thinking systematically, creating constancy of purpose, embracing scientific thinking, creating flow and pull (versus push), assuring quality at the source, leading with humility, and respecting every individual.

The Agile Movement

Started in 2001, the Agile Manifesto was created by seventeen of the leading thinkers in software development, with the goal of turning lightweight methods such as DP and DSDM into a wider movement that could take on heavyweight software development processes such as waterfall development and methodologies such as the rational unified process.

A key principle was to "deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale."² Two other principles focus on the need for small, self-motivated teams, working in a high-trust management model and an emphasis on small batch sizes. Agile is also associated with a set of tools and practices such as Scrum, Standups, and so on.

The Velocity Conference Movement

Started in 2007, the Velocity Conference was created by Steve Souders, John Allspaw, and Jesse Robbins to provide a home for the IT Operations and Web Performance tribe. At the Velocity 2009 conference, John Allspaw and Paul Hammond gave the seminal "10 Deploys per Day: Dev and Ops Cooperation at Flickr," presentation.

The Agile Infrastructure Movement

At the 2008 Agile Toronto conference, Patrick Debois and Andrew Shafer held a "birds of a feather" session on applying Agile principles to infrastructure as opposed to application code. They rapidly gained a following of like-minded thinkers, including John Willis. Later, Debois was so excited by Allspaw and Hammond's "10 Deploys per Day: Dev and Ops Cooperation at Flickr" presentation that he created the first DevOpsDays in Ghent, Belgium, in 2009, coining the word "DevOps."

The Continuous Delivery Movement

Building upon the Development discipline of continuous build, test, and integration, Jez Humble and David Farley extended the concept of continuous delivery, which included a "deployment pipeline" to ensure that code and infrastructure are always in a deployable state and that all code checked into truck is deployed
into production.³ This idea was first presented at Agile 2006 and was also independently developed by Tim Fitz in a blog post titled "Continuous Deployment."⁴

The Toyota Kata Movement

In 2009, Mike Rother wrote *Toyota Kata: Managing People for Improvement, Adaptiveness and Superior Results*, which described learnings over his twenty-year journey to understand and codify the causal mechanisms of the Toyota Production System. *Toyota Kata* describes the "unseen managerial routines and thinking that lie behind Toyota's success with continuous improvement and adaptation . . . and how other companies develop similar routines and thinking in their organizations."⁵

His conclusion was that the Lean community missed the most important practice of all, which he described as the Improvement Kata. He explains that every organization has work routines, and the critical factor in Toyota was making improvement work habitual and building it into the daily work of everyone in the organization. The Toyota Kata institutes an iterative, incremental, scientific approach to problem-solving in the pursuit of a shared organizational true north.⁶

The Lean Startup Movement

In 2011, Eric Ries wrote *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*, codifying the lessons he learned at IMVU, a Silicon Valley startup, which built upon the work of Steve Blank in *The Four Steps to the Epiphany* as well as continuous deployment techniques. Eric Ries also codified related practices and terms including minimum viable product, the build-measure-learn cycle, and many continuous deployment technical patterns.⁷

The Lean UX Movement

In 2013, Jeff Gothelf wrote *Lean UX: Applying Lean Principles to Improve User Experience*, which codified how to improve the "fuzzy front end" and explained how product owners can frame business hypotheses, experiment, and gain confidence in those business hypotheses before investing time and resources in the resulting features. By adding Lean UX, we now have the tools to fully optimize the flow between business hypotheses, feature development, testing, deployment, and service delivery to the customer.

The Rugged Computing Movement

In 2011, Joshua Corman, David Rice, and Jeff Williams examined the apparent futility of securing applications and environments late in the life cycle. In response, they created a philosophy called "Rugged Computing," which attempts to frame the non-functional requirements of stability, scalability, availability, survivability, sustainability, security, supportability, manageability, and defensibility.

Because of the potential for high release rates, DevOps can put incredible pressure on QA and Infosec, because when deploy rates go from monthly or quarterly to hundreds or thousands daily, no longer are two-week turnaround times from Infosec or QA tenable. The Rugged Computing movement posited that the current approach to fighting the vulnerable industrial complex being employed by most information security programs is hopeless.

Appendix 2: Theory of Constraints and Core, Chronic Conflicts

The Theory of Constraints body of knowledge extensively discusses the use of creating core conflict clouds (often referred to as "C3"). Figure A.1 shows the conflict cloud for IT:



Figure A.1: The Core, Chronic Conflict Facing Every IT Organization

During the 1980s, there was a very well-known core, chronic conflict in manufacturing. Every plant manager had two valid business goals: protect sales

and reduce costs. The problem was that in order to protect sales, sales management was incentivized to increase inventory to ensure that it was always possible to fulfill customer demand.

On the other hand, in order to reduce cost, production management was incentivized to decrease inventory to ensure that money was not tied up in work-in-progress that wasn't immediately shippable to the customer in the form of fulfilled sales.

They were able to break the conflict by adopting Lean principles, such as reducing batch sizes, reducing work in process, and shortening and amplifying feedback loops. This resulted in dramatic increases in plant productivity, product quality, and customer satisfaction.

The principles behind DevOps work patterns are the same as those that transformed manufacturing, allowing us to optimize the IT value stream, converting business needs into capabilities and services that provide value for our customers.

Appendix 3: Tabular Form of Downward Spiral

The tabular form of the downward spiral depicted in *The Phoenix Project* is shown in Table A.1.

IT Operations sees	Development sees
Fragile applications are prone to failure	Fragile applications are prone to failure
Long time required to figure out which bit got flipped	More urgent, date-driven projects put into the queue
Detective control is a salesperson	Even more fragile code (less secure) put into production
Too much time required to restore service	More releases have increasingly turbulent installs
Too much firefighting and unplanned work	Release cycles lengthen to amortize cost of deployments
Urgent security rework and remediation	Failing bigger deployments more difficult to diagnose
Planned project work cannot be completed	Most senior and constrained IT ops resources have less time to fix underlying process problems

Table A.1: The Downward Spiral

Frustrated customers leave	Ever increasing backlog of work that could help the business win
Market share goes down	Ever increasing amount of tension between IT Ops, Development, Design
Business misses Wall Street commitments	_
Business makes even larger promises to Wall Street	_

Appendix 4: The Dangers of Handoffs and Queues

The problem with high amounts of queue time is exacerbated when there are many handoffs, because that is where queues are created. Figure A.2 shows wait time as a function of how busy a resource at a work center is. The asymptotic curve shows why a "simple thirty-minute change" often takes weeks to complete—specific engineers and work centers often become problematic bottlenecks when they operate at high utilization. As a work center approaches 100% utilization, any work required from it will languish in queues and won't be worked on without someone expediting/escalating.

In Figure A.2, the x-axis is the percent busy for a given resource at a work center and the y-axis is the approximate wait time (or, more precisely stated, the queue length). What the shape of the line shows is that as resource utilization goes past 80%, wait time goes through the roof.

In *The Phoenix Project*, Bill and his team realized the devastating consequences of this property on lead times for the commitments they were making to the project management office:⁸

I tell them about what Erik told me at MRP-8, about how wait times depend upon resource utilization. "The wait time is the 'percentage of time busy' divided by the 'percentage of time idle.' In other words, if a resource is fifty percent busy, then it's fifty percent idle. The wait time is fifty percent divided by fifty percent, so one unit of time. Let's call it one hour.

So, on average, our task would wait in the queue for one hour before it gets worked.

"On the other hand, if a resource is ninety percent busy, the wait time is 'ninety percent divided by ten percent or nine hours'. In other words,





Figure A.2: Queue Size and Wait Times as a Function of Percent Utilization

Source: Kim, Behr, and Spafford, The Phoenix Project, ePub edition, 557.

I conclude, "So . . . For the Phoenix task, assuming we have seven handoffs, and that each of those resources is busy ninety percent of the time, the tasks would spend in queue a total of nine hours times the seven steps . . ."

"What? Sixty-three hours, just in queue time?" Wes says, incredulously. "That's impossible!"

Patty says with a smirk, "Oh, of course. Because it's only thirty seconds of typing, right?"

Bill and team realize that their "simple thirty-minute task" actually requires seven handoffs (e.g., server team, networking team, database team, virtualization team, and, of course, Brent, the "rockstar" engineer).

Assuming that all work centers were 90% busy, Figure A.2 shows us that the average wait time at each work center is nine hours—and because the work

had to go through seven work centers, the total wait time is seven times that: sixty-three hours.

In other words, the total % of *value added time* (sometimes known as process time) was only 0.16% of the total lead time (thirty minutes divided by sixty-three hours). That means that for 99.8% of our total lead time, the work was simply sitting in queue, waiting to be worked on.

Appendix 5: Myths of Industrial Safety

Decades of research into complex systems shows that countermeasures are based on several myths. In "Some Myths about Industrial Safety," by Denis Besnard and Erik Hollnagel, they are summarized as such:

- Myth 1: "Human error is the largest single cause of accidents and incidents."9
- **Myth 2:** "Systems will be safe if people comply with the procedures they have been given."¹⁰
- **Myth 3:** "Safety can be improved by barriers and protection; more layers of protection results in higher safety."¹¹
- **Myth 4:** "Accident analysis can identify the root cause (the 'truth') of why the accident happened."¹²
- **Myth 5:** "Accident investigation is the logical and rational identification of causes based on facts."¹³
- Myth 6: "Safety always has the highest priority and will never be compromised."¹⁴

The differences between what is myth and what is true are shown in Table A.2.

Myth	Reality
Human error is seen as the cause of failure.	Human error is seen as the effect of systemic vulnerabilities deeper inside the organization.
Saying what people should have done is a satisfying way to describe failure.	Saying what people should have done doesn't explain why it made sense for them to do what they did.

Table A.2: Two Stories

Telling people to be more careful will make the problem go away.

Only by constantly seeking out their vulnerabilities can organizations enhance safety.

Appendix 6: The Toyota Andon Cord

Many ask, how can any work be completed if the Andon cord is being pulled over five thousand times per day? To be precise, not every Andon cord pull results in stopping the entire assembly line. Rather, when the Andon cord is pulled, the team leader overseeing the specified work center has fifty seconds to resolve the problem. If the problem has not been resolved by the time the fifty seconds is up, the partially assembled vehicle will cross a physically drawn line on the floor, and the assembly line will be stopped.¹⁵



Figure A.3: The Toyota Andon Cord

Appendix 7: COTS Software

Currently, in order to get complex COTS (commercial off-the-shelf) software (e.g., SAP, IBM WebSphere, Oracle WebLogic) into version control, we may have to eliminate the use of graphical point-and-click vendor installer tools. To do that, we need to discover what the vendor installer is doing, and we may need to do an install on a clean server image, diff the file system, and put those added files into version control. Files that don't vary by environment are put into one place ("base install"), while environment-specific files are put into their own directory ("test" or "production"). By doing this, software install operations become merely a version control operation, enabling better visibility, repeatability, and speed.

We may also have to transform any application configuration settings so that they are in version control. For instance, we may transform application configurations that are stored in a database into XML files and vice versa.

Appendix 8: Post-Mortem Meetings (Retrospective)

A sample agenda of the post-mortem meeting is shown below:¹⁶

- An initial statement will be made by the meeting leader or facilitator to reinforce that this meeting is a blameless post-mortem and that we will not focus on past events or speculate on "would haves" or "could haves." Facilitators might read the "Retrospective Prime Directive" from the website Retrospective.com.
- Furthermore, the facilitator will remind everyone that any countermeasures must be assigned to someone, and if the corrective action does not warrant being a top priority when the meeting is over, then it is not a corrective action. (This is to prevent the meeting from generating a list of good ideas that are never implemented.)
- Those at the meeting will reach an agreement on the complete timeline of the incident, including when and who detected the issue, how it was discovered (e.g., automated monitoring, manual detection, customer notified us), when service was satisfactorily restored, and so forth. We will also integrate into the timeline all external communications during the incident.
- When we use the word "timeline," it may evoke the image of a linear set of steps of how we gained an understanding of the problem and eventually

fixed it. In reality, especially in complex systems, there will likely be many events that contributed to the accident, and many troubleshooting paths and actions will have been taken in an effort to fix it. In this activity, we seek to chronicle all of these events and the perspectives of the actors and establish hypotheses concerning cause and effect where possible.

- The team will create a list of all the factors that contributed to the incident, both human and technical. They may then sort them into categories, such as "design decision," "remediation," "discovering there was a problem," and so forth. The team will use techniques such as brainstorming and the "infinite hows" to drill down on contributing factors they deem particularly important to discover deeper levels of contributing factors. All perspectives should be included and respected—nobody should be permitted to argue with or deny the reality of a contributing factor somebody else has identified. It's important for the post-mortem facilitator to ensure that sufficient time is spent on this activity and that the team doesn't try and engage in convergent behavior such as trying to identify one or more "root causes."
- Those at the meeting will reach an agreement on the list of corrective actions that will be made top priorities after the meeting. Assembling this list will require brainstorming and choosing the best potential actions to either prevent the issue from occurring or enable faster detection or recovery. Other ways to improve the systems may also be included.
- Our goal is to identify the smallest number of incremental steps to achieve the desired outcomes, as opposed to "Big Bang" changes, which not only take longer to implement, but delay the improvements we need.
- We will also generate a separate list of lower-priority ideas and assign an owner. If similar problems occur in the future, these ideas may serve as the foundation for crafting future countermeasures.
- Those at the meeting will reach an agreement on the incident metrics and their organizational impact. For example, we may choose to measure our incidents by the following metrics:
 - Event severity: How severe was this issue? This directly relates to the impact on the service and our customers.
 - Total downtime: How long were customers unable to use the service to any degree?

- Time to detect: How long did it take for us or our systems to know there was a problem?
- Time to resolve: How long after we knew there was a problem did it take for us to restore service?

Bethany Macri from Etsy observed, "Blamelessness in a post-mortem does not mean that no one takes responsibility. It means that we want to find out what the circumstances were that allowed the person making the change or who introduced the problem to do this. What was the larger environment? . . . The idea is that by removing blame, you remove fear, and by removing fear, you get honesty."¹⁷

Appendix 9: The Simian Army

After the 2011 AWS US-East outage, Netflix had numerous discussions about engineering their systems to automatically deal with failure. These discussions have evolved into a service called "Chaos Monkey."¹⁸

Since then, Chaos Monkey has evolved into a whole family of tools, known internally as the "Netflix Simian Army," to simulate increasingly catastrophic levels of failures:¹⁹

- Chaos Gorilla: simulates the failure of an entire AWS availability zone.
- **Chaos Kong:** simulates failure of entire AWS regions, such as North America or Europe.

Other member of the Simian Army now include:

- Latency Monkey: induces artificial delays or downtime in their RESTful client-server communication layer to simulate service degradation and ensure that dependent services respond appropriately.
- **Conformity Monkey**: finds and shuts down AWS instances that don't adhere to best-practices (e.g., when instances don't belong to an auto-scaling group or when there is no escalation engineer email address listed in the service catalog).
- **Doctor Monkey:** taps into health checks that run on each instance and finds unhealthy instances and proactively shuts them down if owners don't fix the root cause in time.

- Janitor Monkey: ensures that their cloud environment is running free of clutter and waste; searches for unused resources and disposes of them.
- Security Monkey: an extension of Conformity Monkey; finds and terminates instances with security violations or vulnerabilities, such as improperly configured AWS security groups.

Appendix 10: Transparent Uptime

Lenny Rachitsky wrote about the benefits of what he called "transparent uptime:"²⁰

Your support costs go down as your users are able to self-identify system wide problems without calling or emailing your support department. Users will no longer have to guess whether their issues are local or global and can more quickly get to the root of the problem before complaining to you.

You are better able to communicate with your users during downtime events, taking advantage of the broadcast nature of the internet versus the one-to-one nature of email and the phone. You spend less time communicating the same thing over and over and more time resolving the issue.

You create a single and obvious place for your users to come to when they are experiencing downtime. You save your users' time currently spent searching forums, Twitter, or your blog.

Trust is the cornerstone of any successful SaaS adoption. Your customers are betting their business and their livelihoods on your service or platform. Both current and prospective customers require confidence in your service. Both need to know they won't be left in the dark, alone and uninformed, when you run into trouble. Real time insight into unexpected events is the best way to build this trust. Keeping them in the dark and alone is no longer an option.

It's only a matter of time before every serious SaaS provider will be offering a public health dashboard. Your users will demand it.

BIBLIOGRAPHY

- "A Conversation with Werner Vogels." ACM Queue 4, no. 4 (2006): 14–22. https://queue.acm.org/ detail.cfm?id=1142065.
- Adler, Paul. "Time-and-Motion Regained." Harvard Business Review, January-February 1993. https://hbr.org/1993/01/time-and-motion-regained.
- Agile Alliance. "Information Radiators." Glossary. Accessed May 31, 2016. https://www.agilealliance .org/glossary/information-radiators/.
- ALICE. "Pair Programming." Wiki page. Updated April 4, 2014. http://euler.math.uga .edu/wiki/index.php?title=Pair_programming.
- Allspaw, John. "Blameless PostMortems and a Just Culture." *Code as Craft* (blog), Etsy, May 22, 2012. http://codeascraft.com/2012/05/22/blameless-postmortems/.
- Allspaw, John. "Convincing Management that Cooperation and Collaboration Was Worth It." Kitchen Soap (blog), January 5, 2012. http://www.kitchensoap.com/2012/01/05/ convincing-management-that-cooperation-and-collaboration-was-worth-it/.
- Allspaw, John. "Counterfactual Thinking, Rules, and the Knight Capital Accident." Kitchen Soap (blog), October 29, 2013. http://www.kitchensoap.com/2013/10/29/counterfactuals-knight -capital/.
- Allspaw, John interviewed by Jenn Webb. "Post-Mortems, Sans Finger-Pointing." The O'Reilly Radar Podcast. Podcast audio, 30:34. August 21, 2014. http://radar.oreilly.com/2014/08/post mortems-sans-finger-pointing-the-oreilly-radar-podcast.html
- Amazon Web Services, "Summary of the Amazon DynamoDB Service Disruption and Related Impacts in the US-East Region." Amazon Web Services. Accessed May 28, 2016. https://aws .amazon.com/message/5467D2/.
- Anderson, David J. Kanban: Successful Evolutionary Change for Your Technology Business. Sequim, WA: Blue Hole Press, 2010.
- Anderson, David J., and Dragos Dumitriu. From Worst to Best in 9 Months: Implementing a Drum-Buffer-Rope Solution in Microsoft's IT Department. Microsoft Corporation, 2005.
- Antani, Snehal. "IBM Innovate DevOps Keynote." Posted by IBM DevOps, June 12, 2014. YouTube video, 47:57. https://www.youtube.com/watch?v=s0M1P05-6Io.
- Arbuckle, Justin. "What Is ArchOps: Chef Executive Roundtable." 2013.

- Ashman, David. "DOES14—David Ashman—Blackboard Learn—Keep Your Head in the Clouds." Posted by DevOps Enterprise Summit 2014, October 28, 2014. YouTube video, 30:43. https:// www.youtube.com/watch?v=SSmixnMpsI4.
- Associated Press. "Number of Active Users at Facebook over the Years," Yahoo! News. May 1, 2013. https://www.yahoo.com/news/number-active-users-facebook-over-230449748.html?ref=gs.
- Atwood, Jeff. "Pair Programming vs. Code Reviews." *Coding Horror* (blog), November 18, 2013. http://blog.codinghorror.com/pair-programming-vs-code-reviews/.
- Atwood, Jeff. "Software Branching and Parallel Universes." *Coding Horror* (blog), October 2, 2007. http://blog.codinghorror.com/software-branching-and-parallel-universes/.
- Axelos. ITIL Service Transition. Belfast, Ireland: The Stationary Office, 2011.
- Ayers, Zach, and Joshua Cohen. "Andon Cords in Development Teams—Driving Continuous Learning." Presentation at the DevOps Enterprise Summit, Las Vegas, 2019. https://videolibrary. doesvirtual.com/?video=504281981.
- Azzarello, Domenico, Frédéric Debruyne, and Ludovica Mottura. "The Chemisty of Enthusiasm: How Engaged Employees Create Loyal Customers," Bain & Company, May 4, 2012. https:// www.bain.com/insights/the-chemistry-of-enthusiasm.
- Bahri, Sami. "Few Patients-in-Process and Less Safety Scheduling; Incoming Supplies Are Secondary." The W. Edwards Deming Institute Blog, August 22, 2013. https://blog.deming.org/2013/08/ fewer-patients-in-process-and-less-safety-scheduling-incoming-supplies-are-secondary/.
- Barr, Jeff. "EC2 Maintenance Update." *AWS News Blog.* Amazon Web Services, September 25, 2014. https://aws.amazon.com/blogs/aws/ec2-maintenance-update/.
- Basu, Biswanath, Rakesh Goyal, and Jennifer Hansen. "Biz & Tech Partnership Towards 10 'No Fear Releases' Per Day," presenation at DevOps Enterprise Summit, Las Vegas, 2020. https://video library.doesvirtual.com/?video=468711236.
- Bazaarvoice, Inc. Announces Its Financial Results for the Fourth Fiscal Quarter and Fiscal Year Ended April 30, 2012." Bazaar Voice, June 6, 2012. http://investors.bazaarvoice.com/release detail.cfm?ReleaseID=680964.
- Beck, Kent. "Slow Deployment Causes Meetings." Facebook, November 19, 2015. https://www.face book.com/notes/kent-beck/slow-deployment-causes-meetings/1055427371156793?_rdr=p.
- Beck, Kent, Mike Beedle, Arie van Bennekum, Alastair Cockburn, Ward Cunnigham, Martin Fowler, James Grenning, et al. "Twelve Principles of Agile Software." Agile Manifesto, 2001. http:// agilemanifesto.org/principles.html.
- Besnard, Denis and Erik Hollnagel. "Some Myths about Industrial Safety." Paris: Centre De Recherche Sur Les Risques Et Les Crises Mines Working Paper Series 2012. ParisTech, Paris, France, December 2012. http://gswong.com/?wpfb_dl=31.
- Betz, Charles. Architecture and Patterns for IT Service Management, Resource Planning, and Governance: Making Shoes for the Cobbler's Children. Witham, MA: Morgan Kaufmann, 2011.
- Beyond Lean."The 7 Wastes (Seven forms of Muda)." The 7 Wastes Explained. Accessed July 28, 2016. http://www.beyondlean.com/7-wastes.html.
- Big Fish Games. "Big Fish Celebrates 11th Consecutive Year of Record Growth." Pressroom. January 28, 2014. http://pressroom.bigfishgames.com/2014-01-28-Big-Fish-Celebrates-11th -Consecutive-Year-of-Record-Growth.

- Bland, Mike. "DOES15—Mike Bland—Pain Is Over, If You Want It." Posted by Gene Kim to slideshare.net, November 18, 2015. Slideshow. http://www.slideshare.net/ITRevolution/does15 -mike-bland-pain-is-over-if-you-want-it-55236521.
- Bland, Mike. "Fixits, or I Am the Walrus," Mike Bland (blog). Mike Bland, October 4, 2011. https:// mike-bland.com/2011/10/04/fixits.html.
- Bosworth, Andrew. "Building and Testing at Facebook." Facebook, August 8, 2012. https:// www.facebook.com/notes/facebook-engineering/building-and-testing-at-facebook/ 10151004157328920.
- Boubez, Toufic. "Simple Math for Anomaly Detection Toufic Boubez—Metafor Software—Monitorama PDX 2014-05-05," Posted by tboubez to slideshare.net, May 6, 2014. Slideshow. http:// www.slideshare.net/tboubez/simple-math-for-anomaly-detection-toufic-boubez-metafor -software-monitorama-pdx-20140505.
- Brooks, Jr., Frederick P. *The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition.* Upper Saddle River, NJ: Addison-Wesley, 1995.
- Buchanan, Leigh. "The Wisdom of Peter Drucker from A to Z." Inc., November 19, 2009. http://www .inc.com/articles/2009/11/drucker.html.
- Buhr, Sarah. "Etsy Closes Up 86 Percent on First Day of Trading." Tech Crunch, April 16, 2015. http://techcrunch.com/2015/04/16/etsy-stock-surges-86-percent-at-close-of-firstday-of-trading-to-30-per-share/.
- Burrows, Mike. "The Chubby Lock Service for Loosely-Coupled Distributed Systems." Paper presented at OSDI 2006: Seventh Symposium on Operating System Design and Implementation, November 2006. http://static.googleusercontent.com/media/research.google.com/en// archive/chubby-osdi06.pdf.
- Cagan, Marty. Inspired: How to Create Products Customers Love. Saratoga, CA: SVPG Press, 2008.
- Campbell-Pretty, Em. "DOES14—Em Campbell-Pretty—How a Business Exec Led Agile, Lead, CI/ CD." Posted by DevOps Enterprise Summit, April 20, 2014. YouTube video, 29:47. https:// www.youtube.com/watch?v=-4pIMMTbtwE.
- Canahuati, Pedro. "Growing from the Few to the Many: Scaling the Operations Organization at Facebook." Filmed December 16, 2013 for QCon. Video, 39:39. http://www.infoq.com/ presentations/scaling-operations-facebook.
- Chacon, Scott. "GitHub Flow." Scott Chacon (blog), August 31, 2011. http://scottchacon .com/2011/08/31/github-flow.html.
- Chakrabarti, Arup. "Common Ops Mistakes." Filmed presentation at Heavy Bit Industries, June 3, 2014. Video, 36:12. http://www.heavybit.com/library/video/common-ops-mistakes/.
- Chan, Jason. "OWASP AppSecUSA 2012: Real World Cloud Application Security." Posted by Christiaan008, December 10, 2012. Youtube video, 37:45. https://www.youtube.com/ watch?v=daNA0jXDvYk.
- Chandola, Varun, Arindam Banerjee, and Vipin Kumar. "Anomaly Detection: A Survey." ACM Computing Surveys 41, no. 3 (July 2009): 15. http://doi.acm.org/10.1145/1541880.1541882.
- Chapman, Janet, and Patrick Eltridge. "On A Mission: Nationwide Building Society," presentation at DevOps Enterprise Summit, London, 2020. https://videolibrary.doesvirtual.com/?video= 432109857.
- Chuvakin, Anton. "LogLogic/Chuvakin Log Checklist," republished with permission, 2008, http://juliusdavies.ca/logging/llclc.html.

- Clanton, Ross, and Michael Ducy interviewed by Courtney Kissler and Jason Josephy. "Continuous Improvement at Nordstrom." The Goat Farm, season 1, episode 17. Podcast audio, 53:18. June 25, 2015. http://goatcan.do/2015/06/25/the-goat-farm-episode-7-continuousimprovement-at-nordstrom/.
- Clanton, Ross and Heather Mickman. "DOES14—Ross Clanton and Heather Mickman—DevOps at Target." Posted by DevOps Enterprise Summit 2014, October 29, 2014. YouTube video, 29:20. https://www.youtube.com/watch?v=exrjV9V9vhY.
- Claudius, Jonathan. "Attacking Cloud Services with Source Code." Posted by Jonathan Claudius to speakerdeck.com, April 16, 2013. Slideshow.https://speakerdeck.com/claudijd/ attacking-cloud-services-with-source-code.
- Clemm, Josh. "LinkedIn Started Back in 2003—Scaling LinkedIn—A Brief History." Posted by Josh Clemm to slideshare.net, November 9, 2015. Slideshow. http://www.slideshare.net/joshclemm/ how-linkedin-scaled-a-brief-history/3-LinkedIn_started_back_in_2003.
- Cockcroft, Adrian, Cory Hicks, and Greg Orzell. "Lessons Netflix Learned from the AWS Outage." *The Netflix Tech Blog*, April 29, 2011. http://techblog.netflix.com/2011/04/lessons-netflix -learned-from-aws-outage.html.
- Cockcroft, Adrian, interviewed by Michael Ducy and Ross Clanton. "Adrian Cockcroft of Battery Ventures." *The Goat Farm* season 1, episode 8. Podcast audio, July 31, 2015. http://goatcan .do/2015/07/31/adrian-cockcroft-of-battery-ventures-the-goat-farm-episode-8/.
- Cockcroft, Adrian. "Monitorama—Please, No More Minutes, Milliseconds, Monoliths or Monitoring Tools." Posted by Adrian Cockroft to slideshare.net, May 5, 2014. Slideshow. http://www .slideshare.net/adriancockcroft/monitorama-please-no-more.
- Collins, Justin, Alex Smolen, and Neil Matatall. "Putting to your Robots to Work V1.1." Posted by Neil Matatall to slideshare.net, April 24, 2012. Slideshow. http://www .slideshare.net/xplodersuv/sf-2013-robots/.
- Conrad, Ben, and Matt Hyatt. "Saving the Economy from Ruin (with a Hyperscale Paas)," presentation at the 2021 DevOps Enterprise Summit-Europe Virtual. https://videolibrary.doesvirtual .com/?video=550704128.
- Conway, Melvin E. "How Do Committees Invent?" Mel Conway. Originally published in *Datamation* magazine, April 1968. http://www.melconway.com/research/committees.html.
- Cook, Scott. "Leadership in an Agile Age: An Interview with Scott Cook." By Cassie Divine. Intuit, April 20, 2011. https://web.archive.org/web/20160205050418/http://network.intuit .com/2011/04/20/leadership-in-the-agile-age/.
- Corman, Josh and John Willis. "Immutable Awesomeness—Josh Corman and John Willis at DevOps Enterprise Summit 2015." Posted by Sonatype, October 21, 2015. YouTube video, 34:25. https://www.youtube.com/watch?v=-S8-lrm3iV4.
- Cornago, Fernando, Vikalp Yadav, and Andreia Otto. "From 6-Eye Principle to Release at Scale - adidas Digital Tech 2021," presentation at DevOps Enterprise Summit-Eurpoe, 2021. https:// videolibrary.doesvirtual.com/?video=524020857.
- Cox, Jason. "Disney DevOps: To Infinity and Beyond." Presentated at DevOps Enterprise Summit, San Francisco, 2014.

- Cundiff, Dan, Levi Geinert, Lucas Rettif. "Crowdsourcing Technology Governance," presentation at DevOps Enterprise Summit, San Francisco, 2018. https://videolibrary.doesvirtual .com/?video=524020857.
- Cunningham, Ward. "Ward Explains Debt Metaphor," c2. Last updated January 22, 2011. http://c2.com/cgi/wiki?WardExplainsDebtMetaphor.
- Daniels, Katherine. "Devopsdays Minneapolis 2015—Katherine Daniels—DevOps: The Missing Pieces." Posted by DevOps Minneapolis, July 13, 2015. YouTube video, 33:26. https://www .youtube.com/watch?v=LNJkVw93yTU.
- Davis, Jennifer and Katherine Daniels. Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale. Sebastopol, CA: O'Reilly Media, 2016.
- "Decreasing False Positives in Automated Testing." Posted by Sauce Labs to slideshare.net, March 24, 2015. Slideshow. http://www.slideshare.net/saucelabs/decreasing-false-positives -in-automated-testing.
- DeGrandis, Dominica. "DOES15—Dominica DeGrandis—The Shape of Uncertainty." Posted by DevOps Enterprise Summit, November 5, 2015. Youtube video, 22:54. https://www.youtube .com/watch?v=Gp05i0d34gg.
- Dekker, Sidney. "DevOpsDays Brisbane 2014—Sidney Decker—System Failure, Human Error: Who's to Blame?" Posted by info@devopsdays.org, 2014. Vimeo video, 1:07:38. https://vimeo .com/102167635.
- Dekker, Sidney. Just Culture: Balancing Safety and Accountability. Lund University, Sweden: Ashgate Publishing, 2007.
- Dekker, Sidney. The Field Guide to Understanding Human Error. Lund University, Sweden: Ashgate Publishing, 2006.
- DeLuccia, James, Jeff Gallimore, Gene Kim, and Byron Miller. DevOps Audit Defense Toolkit. Portland, OR: IT Revolution, 2015. http://itrevolution.com/devops-and-auditorsthe-devops-audit-defense-toolkit.
- "DevOps Culture: How to Transform," Cloud.Google.com, accessed August 26, 2021. https://cloud .google.com/architecture/devops/devops-culture-transform.
- Dickerson, Chad. "Optimizing for Developer Happiness." *Code As Craft* (blog), Etsy, June 6, 2011. https://codeascraft.com/2011/06/06/optimizing-for-developer-happiness/.
- Dignan, Larry. "Little Things Add Up." Baseline, October 19, 2005. http://www.baselinemag .com/c/a/Projects-Management/Profiles-Lessons-From-the-Leaders-in-the-iBaselinei500/3.
- Douglas, Jake. "Deploying at Github." The GitHub Blog. GitHub, August 29, 2012. https://github .com/blog/1241-deploying-at-github.
- Dweck, Carol. "Carol Dweck Revisits the 'Growth Mindset." Education Week, September 22, 2015. http://www.edweek.org/ew/articles/2015/09/23-/caroldweck-revisits-the-growth-mindset .html.
- Edmondson, Amy C. "Strategies for Learning from Failure." *Harvard Business Review*, April 2011. https://hbr.org/2011/04/strategies-for-learning-from-failure.
- Edwards, Damon. "DevOps Kaizen: Find and Fix What Is Really Behind Your Problems." Posted by dev2ops to slideshare.net. Slideshow., May 4, 2015. http://www.slideshare.net/dev2ops/ dev-ops-kaizen-damon-edwards.

- Exner, Ken. "Transforming Software Development." Posted by Amazon Web Services, April 10, 2015. YouTube video, 40:57. https://www.youtube.com/watch?v=YCrhemssYuI.
- Figureau, Brice. "The 10 Commandments of Logging." Masterzen's Blog, January 13, 2013. http:// www.masterzen.fr/2013/01/13/the-10-commandments-of-logging/.
- Fitz, Timothy. "Continuous Deployment at IMVU: Doing the Impossible Fifty Times a Day." *Timothy Fitz* (blog), February 10, 2009. http://timothyfitz.com/2009/02/10/ continuous-deployment-at-imvu-doing-the-impossible-fifty-times-a-day/.
- Forsgren, Nicole, Bas Alberts, Kevin Backhouse, and Grey Baker. The 2020 State of the Octoverse. GitHub, 2020. https://octoverse.github.com/static/github-octoverse-2020-security-report .pdf.
- Forsgren, Nicole, Jez Humble, and Gene Kim. Accelerate: State of DevOps 2018. DORA and Google Cloud, 2018. https://lp.google-mkto.com/rs/248-TPC-286/images/DORA-State%20of%20 DevOps.pdf.
- Forsgren, Nicole, Jez Humble, Nigel Kersten, and Gene Kim. "2014 State Of DevOps Findings! Velocity Conference." Posted by Gene Kim to slideshare.net, June 30, 2014. Slideshow. http://www .slideshare.net/realgenekim/2014-state-of-devops-findings-velocity-conference.
- Forsgren, Nicole, Dustin Smith, Jez Humble, and Jessie Frazelle. Accelerate State of DevOps 2019. DORA and Google Cloud, 2019. https://services.google.com/fh/files/misc/state-of -devops-2019.pdf.
- Forsgren, Nicole, Margaret-Anne Storey, Chandra Maddila, Thomas Zimmermann, Brain Houck, and Jenna Butler. "The SPACE of Developer Productivity." ACM Queue 19, no. 1 (2021): 1–29. https://queue.acm.org/detail.cfm?id=3454124.
- Fowler, Chad. "Trash Your Servers and Burn Your Code: Immutable Infrastructure and Disposable Components." *Chad Fowler* (blog), June 23, 2013. http://chadfowler.com/2013/06/23/ immutable-deployments.html.
- Fowler, Martin. "Continuous Integration." Martin Fowler (blog), May 1, 2006. http://www.martinfowler.com/articles/continuousIntegration.html.
- Fowler, Martin. "Eradicating Non-Determinism in Tests." Martin Fowler (blog), April 14, 2011. http://martinfowler.com/articles/nonDeterminism.html.
- Fowler, Martin. "StranglerFigApplication." *Martin Fowler* (blog), June 29, 2004. http://www.martinfowler.com/bliki/StranglerApplication.html.
- Fowler, Martin. "TestDrivenDevelopment." Martin Fowler (blog), March 5, 2005. http://martin fowler.com/bliki/TestDrivenDevelopment.html.
- Fowler, Martin. "TestPyramid." Martin Fowler (blog), May 1, 2012. http://martinfowler.com/bliki/ TestPyramid.html.
- Freakonomics. "Fighting Poverty With Actual Evidence: Full Transcript." Freakonomics blog, November 27, 2013. http://freakonomics.com/2013/11/27/ fighting-poverty-with-actual-evidence-full-transcript/.
- Furtado, Adam, and Lauren Knausenberger. "The Air Force's Digital Journey in 12 Parsecs or Less." Presentation at DevOps Enterprise Summit, London, 2020. https://videolibrary.doesvirtual .com/?video=467489046.
- Gaekwad, Karthik. "Agile 2013 Talk: How DevOps Change Everything." Posted by Karthik Gaekwad to slideshare.net, August 7, 2013. Slideshow. http://www.slideshare.net/karthequian/howdevopschangeseverything agile2013karthikgaekwad/.

- Galbreath, Nick. "Continuous Deployment—The New #1 Security Feature, from BSildesLA 2012." Posted by Nick Galbreath to slideshare.net, August 16, 2012. Slideshow. http://www.slide share.net/nickgsuperstar/continuous-deployment-the-new-1-security-feature.
- Galbreath, Nick. "DevOpsSec: Applying DevOps Principles to Security, DevOpsDays Austin 2012." Posted by Nick Galbreath to slideshare.net, April 12, 2012. Slideshow. http://www.slideshare .net/nickgsuperstar/devopssec-apply-devops-principles-to-security.
- Galbreath, Nick. "Fraud Engineering, from Merchant Risk Council Annual Meeting 2012." Posted by Nick Galbreath to slideshare.net, May 3, 2012. Slideshow. http://www.slideshare.net/ nickgsuperstar/fraud-engineering.
- Gallagher, Sean. "When 'Clever' Goes Wrong: How Etsy Overcame Poor Architectural Choices." Arstechnica, October 3, 2011. http://arstechnica.com/business/2011/10/ when-clever-goes-wrong-how-etsy-overcame-poor-architectural-choices/.
- Gardner, Tom. "Barnes & Noble, Blockbuster, Borders: The Killer B's Are Dying." The Motley Fool, July 21, 2010. http://www.fool.com/investing/general/2010/07/21/barnes-noble-blockbuster -borders-the-killer-bs-are.aspx.
- Geer, Dan and Joshua Corman. "Almost Too Big to Fail." ;*login: The Usenix Magazine* 39, no. 4 (August 2014): 66–68. https://www.usenix.org/system/files/login/articles/15_geer_0.pdf.
- Gertner, Jon. The Idea Factory: Bell Labs and the Great Age of American Innovation. New York: Penguin Books, 2012.
- GitHub. "Etsy's Feature Flagging API Used for Operational Rampups and A/B testing." Etsy/feature. Last updated January 4, 2017. https://github.com/etsy/feature.
- GitHub."Library for Configuration Management API." Netflix/archaius. Last updated December 4, 2019. https://github.com/Netflix/archaius.
- Golden, Bernard. "What Gartner's Bimodal IT Model Means to Enterprise CIOs." CIO Magazine, January 27, 2015. http://www.cio.com/article/2875803/cio-role/what-gartner-s-bimodal-it -model-means-to-enterprise-cios.html.
- Goldratt, Eliyahu M. Beyond the Goal: Eliyahu Goldratt Speaks on the Theory of Constraints (Your Coach in a Box). Prince Frederick, MD: Gildan Media, 2005.
- Google App Engine Team. "Post-Mortem for February 24, 2010 Outage." Google App Engine website, March 4, 2010. https://groups.google.com/forum/#!topic/google-appengine/p2QKJ0OSLc8.
- Govindarajan, Vijay, and Chris Trimble. *The Other Side of Innovation: Solving the Execution Challenge*. Boston, MA: Harvard Business Review, 2010, Kindle.
- Gruver, Gary. "DOES14—Gary Gruver—Macy's—Transforming Traditional Enterprise Software Development Processes." Posted by DevOps Enterprise Summit 2014, October 29, 2014. You-Tube video, 27:24. https://www.youtube.com/watch?v=-HSSGiYXA7U.
- Gruver, Gary, and Tommy Mouser. *Leading the Transformation: Applying Agile and DevOps Principles at Scale*. Portland, OR: IT Revolution Press, 2015.
- Gupta, Prachi. "Visualizing LinkedIn's Site Performance." LinkedIn Engineering blog, June 13, 2011. https://engineering.linkedin.com/25/visualizing-linkedins-site-performance.
- Hammant, Paul. "Introducing Branch by Abstraction." *Paul Hammant's Blog*, April 26, 2007. http://paulhammant.com/blog/branch_by_abstraction.html.
- Hastings, Reed. "Netflix Culture: Freedom and Responsibility." Posted by Reed Hastings to slideshare .net, August 1, 2009. Slideshow. http://www.slideshare.net/reed2001/culture-1798664.

- Hendrickson, Elisabeth. "DOES15—Elisabeth Hendrickson—Its All About Feedback." Posted by DevOps Enterprise Summit, November 5, 2015. YouTube video, 34:47. https://www.youtube. com/watch?v=r2BFTXBundQ.
- Hendrickson, Elisabeth. "On the Care and Feeding of Feedback Cycles." Posted by Elisabeth Hendrickson to slideshare.net, November 1, 2013. Slidshow. http://www.slideshare.net/ehendrickson/ care-and-feeding-of-feedback-cycles.
- Hodge, Victoria, and Jim Austin. "A Survey of Outlier Detection Methodologies." Artificial Intelligence Review 22, no. 2 (October 2004): 85–126. http://www.geo.upm.es/postgrado/Carlos Lopez/papers/Hodge+Austin_OutlierDetection_AIRE381.pdf.
- Holmes, Dwayne. "How A Hotel Company Ran \$30B of Revenue in Containers," presentation at DevOps Enterprise Summit, Las Vegas, 2020. https://videolibrary.doesvirtual .com/?video=524020857.
- "How I Structured Engineering Teams at LinkedIn and AdMob for Success." First Round Review, 2015. http://firstround.com/review/how-i-structured-engineering-teams-at-linkedin-and-admob-for -success/.
- Hrenko, Michael. "DOES15—Michael Hrenko—DevOps Insured By Blue Shield of California." Posted by DevOps Enterprise Summit, November 5, 2015. YouTube video, 42:24. https://www .youtube.com/watch?v=NlgrOT24UDw.
- Huang, Gregory T. "Blackboard CEO Jay Bhatt on the Global Future of Edtech." *Xconomy*, June 2, 2014. http://www.xconomy.com/boston/2014/06/02/ blackboard-ceo-jay-bhatt-on-the-global-future-of-edtech/.
- Humble, Jez. "What is Continuous Delivery?" Continuous Delivery (website), accessed May 28, 2016, https://continuousdelivery.com/.
- Humble, Jez, and David Farley. Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation. Upper Saddle River, NJ: Addison-Wesley, 2011.
- Humble, Jez, Joanne Molesky, and Barry O'Reilly. Lean Enterprise: How High Performance Organizations Innovate at Scale. Sebastopol, CA: O'Reilly Media, 2015.
- "IDC Forecasts Worldwide IT Spending to Grow 6% in 2012, Despite Economic Uncertainty." Business Wire, September 10, 2012. http://www.businesswire.com/news/home/20120910005280/ en/IDC-Forecasts-Worldwide-Spending-Grow-6-2012.
- Immelt, Jeff. "GE CEO Jeff Immelt: Let's Finally End the Debate over Whether We Are in a Tech Bubble." Business Insider, December 9, 2015. http://www.businessinsider.com/ ceo-of-ge-lets-finally-end-the-debate-over-whether-we-are-in-a-tech-bubble-2015-12.
- Intuit, Inc. "2012 Annual Report: Form 10-K." July 31, 2012. http://s1.q4cdn.com/018592547/ files/doc_financials/2012/INTU_2012_7_31_10K_r230_at_09_13_12_FINAL_and_ Camera_Ready.pdf.
- Jacobson, Daniel, Danny Yuan, and Neeraj Joshi. "Scryer: Netflix's Predictive Auto Scaling Engine." *The Netflix Tech Blog*, November 5, 2013. http://techblog.netflix.com/2013/11/scryernetflixs-predictive-auto-scaling.html.
- Jenkins, Jon. "Velocity 2011: Jon Jenkins, 'Velocity Culture." Posted by O'Reilly, June 20, 2011. YouTube video, 15:13. https://www.youtube.com/watch?v=dxk8b9rSKOo.
- "Jeremy Long: The (Application) Patching Manifesto," YouTube video, 41:17, posted by Loco-MocoSec: Hawaii Product Security Conference, May 17, 2018. https://www.youtube.com/ watch?v=qVVZrTRJ290.

- JGFLL. Review of *The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win*, by Gene Kim, Kevin Behr, and George Spafford. Amazon review, March 4, 2013. http://www.amazon.com/review/R1KSSPTEGLWJ23.
- Johnson, Kimberly H., Tim Judge, Christopher Porter, and Ramon Richards. "How Fannie Mae Uses Agility to Support Homeowners and Renters," presentation at DevOps Enterprise Summit, Las Vegas, 2020. https://videolibrary.doesvirtual.com/?video=467488997.
- Jones, Angie. "3 Ways to Get Test Automation Done Within Your Sprints." TechBeacon. Accessed February 15, 2021. https://techbeacon.com/app-dev-testing/3-ways-get-testautomation-done-within-your-sprints.
- Kash, Wyatt. "New Details Released on Proposed 2016 IT Spending." FedScoop, February 4, 2015. http://fedscoop.com/what-top-agencies-would-spend-on-it-projects-in-2016.
- Kastner, Erik. "Quantum of Deployment." Code as Craft (blog). Etsy, May 20, 2010. https:// codeascraft.com/2010/05/20/quantum-of-deployment/.
- Kersten, Mik. "Project to Product: From Stories to Scenius," Tasktop blog, November 21, 2018, https://www.tasktop.com/blog/project-product-stories-scenius/.
- Kersten, Mik. Project to Product: How to Survive and Thrive in the Age of Digital Disruption with the Flow Framework. Portland, Oregon: IT Revolution Press, 2018.
- Kersten, Nigel, IT Revolution, and PwC. 2015 State of DevOps Report. Portland, OR: Puppet Labs, 2015. https://puppet.com/resources/white-paper/2015-state-of-devops-report?_ga=1.66126 58.168869.1464412647&link=blog.
- Kim, Gene. "The Amazing DevOps Transformation of the HP LaserJet Firmware Team (Gary Gruver)." IT Revolution blog, February 13, 2014. http://itrevolution.com/ the-amazing-devops-transformation-of-the-hp-laserjet-firmware-team-gary-gruver/.
- Kim, Gene. "Organizational Learning and Competitiveness: Revisiting the 'Allspaw/Hammond 10 Deploys Per Day at Flickr' Story." IT Revolution blog, December 13, 2014. http://itrevolution .com/organizational-learning-and-competitiveness-a-different-view-of-the-allspaw hammond-10-deploys-per-day-at-flickr-story/.
- Kim, Gene. "State of DevOps: 2020 and Beyond," IT Revolution blog, March 1, 2021. https://itrevolution.com/state-of-devops-2020-and-beyond/.
- Kim, Gene. "The Three Ways: The Principles Underpinning DevOps." IT Revolution blog, August 22, 2012. http://itrevolution.com/the-three-ways-principles-underpinning-devops/.
- Kim, Gene, Kevin Behr, and George Spafford. The Visible Ops Handbook: Implementing ITIL in 4 Practical and Auditable Steps. Eugene, OR: IT Process Institute, 2004.
- Kim, Gene, Gary Gruver, Randy Shoup, and Andrew Phillips. "Exploring the Uncharted Territory of Microservices." Xebia Labs. Webinar, February 20, 2015. https://xebialabs.com/community/ webinars/exploring-the-uncharted-territory-of-microservices/.
- Kissler, Courtney. "DOES14—Courtney Kissler—Nordstrom—Transforming to a Culture of Continuous Improvement." Posted by DevOps Enterprise Summit 2014, October 29, 2014. YouTube video, 29:59. https://www.youtube.com/watch?v=0ZAcsrZBSlo.
- Kohavi, Ron, Thomas Crook, and Roger Longbotham. "Online Experimentation at Microsoft." Paper presented at the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 2009. http://www.exp-platform.com/documents/exp_ dmcasestudies.pdf.

- Krishnan, Kripa. "Kripa Krishnan: 'Learning Continuously From Failures' at Google." Posted by Flowcon, November 11, 2014. YouTube video, 21:35. https://www.youtube.com/ watch?v=KqqS3wgQum0.
- Krishnan, Kripa. "Weathering the Unexpected." Communications of the ACM 55, no. 11 (November 2012):48–52.http://cacm.acm.org/magazines/2012/11/156583-weathering-the-unexpected/ abstract.
- Kumar, Ashish. "Development at the Speed and Scale of Google." PowerPoint presented at QCon, San Francisco, CA, 2010. https://qconsf.com/sf2010/dl/qcon-sanfran-2010/slides/Ashish Kumar_DevelopingProductsattheSpeedandScaleofGoogle.pdf.
- Leibman, Maya, and Ross Clanton. "DevOps: Approaching Cruising Altitude." Presentation at DevOps Enterprise Summit, Las Vegas, 2020, https://videolibrary.doesvirtual .com/?video=550704282.
- Letuchy, Eugene. "Facebook Chat." Facebook, May 3, 2008. http://www.facebook.com/note. php?note_id=14218138919&id=944554719.
- Lightbody, Patrick. "Velocity 2011: Patrick Lightbody, 'From Inception to Acquisition." Posted by O'Reilly, June 17, 2011. YouTube video, 15:28. https://www.youtube.com/ watch?v=ShmPod8JecQ.
- Limoncelli, Tom. "Python Is Better than Perl6." Everything SysAdmin blog, January 10, 2011. http://everythingsysadmin.com/2011/01/python-is-better-than-perl6.html.
- Limoncelli, Tom. "SRE@Google: Thousands Of DevOps Since 2004." USENIX Association Talk, NYC. Posted by USENIX, January 12, 2012.Youtube video, 45:57. http://www.youtube.com/ watch?v=iIuTnhdTzK0.
- Limoncelli, Tom. "Stop Monitoring Whether or Not Your Service Is Up!" Everything SysAdmin blog, November 27, 2013. http://everythingsysadmin.com/2013/11/stop-monitoring-ifservice-is-up.html.
- Limoncelli, Tom. "Yes, You Can Really Work from HEAD." Everything SysAdmin blog, March 15, 2014. http://everythingsysadmin.com/2014/03/yes-you-really-can-work-from-head.html.
- Lindsay, Jeff. "Consul Service Discovery with Docker." Progrium blog, August 20, 2014. http:// progrium.com/blog/2014/08/20/consul-service-discovery-with-docker.
- Loura, Ralph, Olivier Jacques, and Rafael Garcia. "DOES15—Ralph Loura, Olivier Jacques, & Rafael Garcia—Breaking Traditional IT Paradigms to ..." Posted by DevOps Enterprise Summit, November 16, 2015. YouTube video, 31:07. https://www.youtube.com/watch?v= q9nNqqie_sM.
- Lublinsky, Boris. "Versioning in SOA." *The Architecture Journal*, April 2007. https://msdn.microsoft .com/en-us/library/bb491124.aspx.
- Lund University. "Just Culture: Balancing Safety and Accountability." Human Factors & System Safety website, November 6, 2015. http://www.humanfactors.lth.se/sidney-dekker/books/ just-culture/.
- Luyten, Stefan. "Single Piece Flow: Why Mass Production Isn't the Most Efficient Way of Doing 'Stuff." Medium (blog), August 8, 2014. https://medium.com/@stefanluyten/ single-piece-flow-5d2c2bec845b#.9o7sn74ns.
- Macri, Bethany. "Morgue: Helping Better Understand Events by Building a Post Mortem Tool— Bethany Macri." Posted by info@devopsdays.org, October 18, 2013. Vimeo video, 33:34. http://vimeo.com/77206751.

- Malpass, Ian. "DevOpsDays Minneapolis 2014—Ian Malpass, Fallible Humans." Posted by DevOps Minneapolis, July 20, 2014. YouTube video, 35:48. https://www.youtube.com/ watch?v=5NY-SrQFrBU.
- Malpass, Ian. "Measure Anything, Measure Everything." Code as Craft (blog). Etsy, February 15, 2011. http://codeascraft.com/2011/02/15/measure-anything-measure-everything/.
- Mangot, Dave, and Karthik Rajan. "Agile.2013.effecting.a.devops.transformation.at.salesforce." Posted by Dave Mangot to slideshare.net, August 12, 2013. Slideshow. http://www.slideshare .net/dmangot/agile2013effectingadev-opstransformationatsalesforce.
- Marsh, Dianne. "Dianne Marsh: 'Introducing Change while Preserving Engineering Velocity." Posted by Flowcon, November 11, 2014. YouTube video, 17:37. https://www.youtube.com/ watch?v=eW3ZxY67fnc.
- Martin, Karen, and Mike Osterling. Value Stream Mapping: How to Visualize Work and Align Leadership for Organizational Transformation. New York: McGraw Hill, 2013.
- Maskell, Brian. "What Does This Guy Do? Role of Value Stream Manager." Maskell (blog), July 3, 2015. http://blog.maskell.com/?p=2106http://www.lean.org/common/display/?o=221.
- Masli, Adi., Vernon J. Richardson, Marcia Widenmier Watson, and Robert W. Zmud. "Senior Executives' IT Management Responsibilities: Serious IT-Related Deficiencies and CEO/CFO Turnover." MIS Quarterly 40, no. 3 (2016): 687–708. https://doi.org/10.25300/misq/2016/40.3.08.
- Massachusetts Institute of Technology. "Creating High Velocity Organizations." Course Descriptions. Accessed May 30, 2016. http://executive.mit.edu/openenrollment/program/ organizational-development-high-velocity-organizations.
- Mathew, Reena, and Dave Mangot. "DOES14—Reena Mathew and Dave Mangot—Salesforce." Posted by ITRevolution to slideshare.net, October 29, 2014. Slideshow. http://www.slideshare. net/ITRevolution/does14-reena-matthew-and-dave-mangot-salesforce.
- Mauro, Tony. "Adopting Microservices at Netflix: Lessons for Architectural Design." NGINX (blog), February 19, 2015. https://www.nginx.com/blog/microservices-at-netflix -architectural-best-practices/.
- McDonnell, Patrick. "Continuously Deploying Culture: Scaling Culture at Etsy—Velocity Europe 2012." Posted by Patrick McDonnell to slideshare.net, October 4, 2012. Slideshow. http://www .slideshare.net/mcdonnps/continuously-deploying-culture-scaling-culture-at-etsy-14588485.
- McKinley, Dan. "Why MongoDB Never Worked Out at Etsy." *Dan McKinley* (blog), December 26, 2012. http://mcfunley.com/why-mongodb-never-worked-out-at-etsy.
- Mell, Peter, and Timothy Grance. The NIST Definition of Cloud Computing: Recommendations of the National Institute of Standards and Technology. Washington, DC>: National Institute of Standards and Technology, 2011.
- Messeri, Eran. "What Goes Wrong When Thousands of Engineers Share the Same Continuous Build?" Presented at the GOTO Conference, Aarhus, Denmark, October 2, 2013.
- Metz, Cade. "Google Is 2 Billion Lines of Code—and It's All in One Place." Wired, September 16, 2015. http://www.wired.com/2015/09/google-2-billion-lines-codeand-one-place/.
- Metz, Cade. "How Three Guys Rebuilt the Foundation of Facebook." *Wired*, June 10, 2013. http://www.wired.com/wiredenterprise/2013/06/facebook-hhvm-saga/all/.
- Mickman, Heather, and Ross Clanton. "DOES15—Heather Mickman & Ross Clanton—(Re)building an Engineering Culture: DevOps at Target." Posted by DevOps Enterprise Summit, November 5, 2015. YouTube video, 33:39. https://www.youtube.com/watch?v=7s-VbB1fG5o.

- Milstein, Dan. "Post-Mortems at HubSpot: What I Learned from 250 Whys." Hub-Spot (blog), June 1, 2011. http://product.hubspot.com/blog/bid/64771/ Post-Mortems-at-HubSpot-What-I-Learned-From-250-Whys.
- Morgan, Timothy Prickett. "A Rare Peek Into The Massive Scale of AWS." Enterprise AI, November 14, 2014. http://www.enterprisetech.com/2014/11/14/rare-peek-massive-scale.
- Morrison, Erica. "How We Turned Our Company's Worst Outage into a Powerful Learning Opportunity." Presentation at DevOps Enterprise Summit, London, 2020. https://videolibrary.does virtual.com/?video=431872263.
- Moore, Geoffrey A., and Regis McKenna. Crossing the Chasm: Marketing and Selling High-Tech Products to Mainstream Customers. New York: HarperCollins, 2009.
- Mueller, Ernest. "2012—A Release Odyssey." Posted by Ernest Mueller to slideshare.net, March 12, 2014. Slideshow. http://www.slideshare.net/mxyzplk/2012-a-release-odyssey.
- Mueller, Ernest. "Business Model Driven Cloud Adoption: What NI Is Doing in the Cloud." posted by Ernest Mueller to slideshare.net, June 28, 2011. Slideshow. http://www.slideshare.net/ mxyzplk/business-model-driven-cloud-adoption-what-ni-is-doing-in-the-cloud.
- Mueller, Ernest. "DOES15—Ernest Mueller—DevOps Transformations at National Instruments and . . . " Posted by DevOps Enterprise Summit, November 5, 2015. YouTube video, 34:14. https://www.youtube.com/watch?v=6Ry40h1UAyE.
- Mulkey, Jody. "DOES15—Jody Mulkey—DevOps in the Enterprise: A Transformation Journey." Posted by DevOps Enterprise Summit, November 5, 2015. YouTube video, 28:22. https://www .youtube.com/watch?v=USYrDaPEFtM.
- Nagappan, Nachiappan, E. Michael Maximilien, Thirumalesh Bhat, and Laurie Williams. "Realizing Quality Improvement through Test Driven Development: Results and Experiences of Four Industrial Teams." *Empire Software Engineering* 13 (2008): 289–302. http://research.microsoft .com/en-us/groups/ese/nagappan_tdd.pdf.
- Naraine, Ryan. "Twilio, HashiCorp Among Codecov Supply Chain Hack Victims," SecurityWeek, May 10, 2021. https://www.securityweek.com/twilio-hashicorp-among-codecov-supplychain-hack-victims.
- Nationwide. 2014 Annual Report. 2014 https://www.nationwide.com/about-us/nationwideannual-report-2014.jsp.
- Nielsen, Jonas Klit. "8 Years with LinkedIn—Looking at the Growth." Mind Jumpers blog, May 10, 2011. http://www.mindjumpers.com/blog/2011/05/linkedin-growth-infographic/.
- Netflix. Letter to Shareholders, January 19, 2016. http://files.shareholder.com/downloads/NFLX /2432188684x0x870685/C6213FF9-5498-4084-A0FF-74363CEE35A1/Q4_15_Letter_to_ Shareholders_-_COMBINED.pdf.
- Newland, Jesse. "ChatOps at GitHub." Posted on speakerdeck.com, February 7, 2013. Slideshow . https://speakerdeck.com/jnewland/chatops-at-github.
- North, Dan. "Ops and Operability." Posted to speakerdeck.com, February 25, 2016. Slideshow . https://speakerdeck.com/tastapod/ops-and-operability.
- "NUMMI." *This American Life* episode 403, March 26, 2010. Radio. http://www.thisamericanlife.org/ radio-archives/episode/403/transcript.
- Nygard, Michael T. *Release It!: Design and Deploy Production-Ready Software*. Raleigh, NC: Pragmatic Bookshelf, 2007, Kindle.

- O'Donnell, Glenn. "DOES14—Glenn O'Donnell—Forrester—Modern Services Demand a DevOps Culture Beyond Apps." Posted by DevOps Enterprise Summit 2014, November 5, 2014. You-Tube video, 12:20. https://www.youtube.com/watch?v=pvPWKuO4_48.
- O'Reilly, Barry. "How to Implement Hypothesis-Driven Development." Barry O'Reilly blog, October 21, 2013. http://barryoreilly.com/explore/blog/how-to-implement-hypothesisdriven-development/.
- Osterweil, Leon. "Software Processes Are Software Too." Paper presented at International Conference on Software Engineering, Monterey, CA, 1987. http://www.cs.unibo.it/cianca/www pages/ids/letture/Osterweil.pdf.
- OWASP. "OWASP Cheat Sheet Series." Updated March 2, 2016. https://www.owasp.org/index.php/ OWASP_Cheat_Sheet_Series.
- Özil, Giray. "Ask a programmer to review 10 lines of code." Twitter, February 27, 2013. https:// twitter.com/girayozil/status/306836785739210752.
- Pal, Tapabrata. "DOES15—Tapabrata Pal—Banking on Innovation & DevOps." Posted by DevOps Enterprise Summit, January 4, 2016. YouTube video, 32:57. https://www.youtube.com/ watch?v=bbWFCKGhxOs.
- Parikh, Karan. "From a Monolith to Microservices + REST: The Evolution of LinkedIn's Architecture." Posted by Karan Parikh to slideshare.net, November 6, 2014. Slideshow. http://www .slideshare.net/parikhk/restli-and-deco.
- "Paul O'Neill." Forbes, October 11, 2001. http://www.forbes.com/2001/10/16/poneill.html.
- Paul, Ryan. "Exclusive: A Behind-the-Scenes Look at Facebook Release Engineering." Ars Technica, April 5, 2012. http://arstechnica.com/business/2012/04/exclusive-a-behind-thescenes-look-at-facebook-release-engineering/1/.
- PCI Security Standards Council. "Glossary." Glossary of terms (website). Accessed May 30, 2016. https://www.pcisecuritystandards.org/pci_security/glossary.
- PCI Security Standards Council. Payment Card Industry (PCI) Data Security Stands: Requirements and Security Assessment Procedures, Version 3.1. PCI Security Standards Council, 2015, Section 6.3.2. https://webcache.googleusercontent.com/search?q=cache:hpRe2COzzdA-J:https://www.cisecuritystandards.org/documents/PCI_DSS_v3-1_SAQ_D_Merchant_rev1-1 .docx+&cd=2&hl=en&ct=clnk&gl=us.
- Pepitone, Julianne. "Amazon EC2 Outage Downs Reddit, Quora." CNN Money, April 22, 2011. http:// money.cnn.com/2011/04/21/technology/amazon _server_outage/index.htm.
- Perrow, Charles. Normal Accidents: Living with High-Risk Technologies. Princeton, NJ: Princeton University Press, 1999.
- Plastic SCM. "Version Control History." History of version control. Accessed May 31, 2016. https:// www.plasticscm.com/version-control-history.
- Pomeranz, Hal. "Queue Inversion Week." *Righteous IT*, February 12, 2009. https://righteousit .wordpress.com/2009/02/12/queue-inversion-week/.
- Poppendieck, Mary, and Tom Poppendieck. *Implementing Lean Software: From Concept to Cash*. Upper Saddle River, NJ: Addison-Wesley, 2007.
- Potvin, Rachel, and Josh Levenber. "Why Google Stores Billions of Lines of Code in a Single Repository." Communications of the ACM 59, no.7 (July 2016): 78–87. https://cacm.acm.org/ magazines/2016/7/204032-why-google-stores-billions-of-lines-of-code-in-a-single-repository/fulltext.

- "Post Event Retrospective—Part 1." *Rally Blogs*, accessed May 31, 2016. https://www.rallydev.com/ blog/engineering/post-event-retrospective-part-i.
- Protalinski, Emil. "Facebook Passes 1.55B Monthly Active Users and 1.01B Daily Active Users." Venture Beat, November 4, 2015. http://venturebeat.com/2015/11/04/ facebook-passes-1-55b-monthly-active-users-and-1-01-billion-daily-active-users/.
- Prugh, Scott. "Continuous Delivery." Scaled Agile Framework. Updated February 14, 2013, http:// www.scaledagileframework.com/continuous-delivery/.
- Prugh, Scott. "DOES14: Scott Prugh, CSG—DevOps and Lean in Legacy Environments." Posted by DevOps Enterprise Summit to slideshare.net, November 14, 2014. Slideshow. http://www .slideshare.net/DevOps EnterpriseSummit/scott-prugh.
- Prugh, Scott, and Erica Morrison. "DOES15—Scott Prugh & Erica Morrison—Conway & Taylor Meet the Strangler (v2.0)." Posted by DevOps Enterprise Summit, November 5, 2015. YouTube video, 29:39. https://www.youtube.com/watch?v=tKdIHCL0DUg.
- Prugh, Scott, and Erica Morrison. "When Ops Swallows Dev," presentation at DevOps Enteprise Summit 2016. https://videolibrary.doesvirtual.com/?video=524430639.
- Puppet Labs and IT Revolution Press. 2013 State of DevOps Report. Portland, OR: Puppet Labs, 2013. http://www.exin-library.com/Player/eKnowledge/2013-state-of-devops-report.pdf.
- Ratchitsky, Lenny. "7 Keys to a Successful Public Health Dashboard." Transparent Uptime, December 1, 2008. http://www.transparentuptime.com/2008/11/rules-for-successful-public-health. html.
- Raymond, Eric S. "Conway's Law." Eric Raymond. Accessed May 31, 2016. http://catb.org/~esr/ jargon/.
- Rembetsy, Michael, and Patrick McDonnell. "Continuously Deploying Culture: Scaling Culture at Etsy." Posted by Patrick McDonnel.bl to slideshare.net, October 4, 2012. Slideshow. http://www .slideshare.net/mcdonnps/continuously-deploying-culture-scaling-culture-at-etsy-14588485.
- Ries, Eric. The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. New York: Random House, 2011. Audiobook.
- Ries, Eric. "Work in Small Batches." *Startup Lessons Learned* (blog), February 20, 2009. http://www .startuplessonslearned.com/2009/02/work-in-small-batches.html.
- Robbins, Jesse. "GameDay: Creating Resiliency Through Destruction—LISA11." Posted by Jesse Robbins to slideshare.net, December 7, 2011. Slideshow. http://www.slideshare.net/jesserobbins/ ameday-creating-resiliency-through-destruction.
- Robbins, Jesse. "Hacking Culture at VelocityConf." posted by Jesse Robbins to slideshare.net, June 28, 2012. Slideshow. http://www.slideshare.net/jesserobbins/hacking-culture-at-velocityconf.
- Robbins, Jesse, Kripa Krishnan, John Allspaw, and Tom Limoncelli. "Resilience Engineering: Learning to Embrace Failure." ACM Queue 10, no. 9 (September 13, 2012). https://queue.acm.org/ detail.cfm?id=2371297.
- Roberto, Michael, Richard M. J. Bohmer, and Amy C. Edmondson. "Facing Ambiguous Threats." Harvard Business Review, November 2006. https://hbr.org/2006/11/facing-ambiguous-threats.
- Rossi, Chuck. "Release Engineering and Push Karma: Chuck Rossi." Facebook, April 5, 2012. https://www.facebook.com/notes/facebook-engineering/release-engineering-and-pushkarma-chuck-rossi/10150660826788920.

- Rossi, Chuck. "Ship early and ship twice as often." Facebook, August 3, 2012. https://www.facebook .com/notes/facebook-engineering/ship-early-and-ship-twice-as-often/10150985860363920.
- Rother, Mike. Toyota Kata: Managing People for Improvement, Adaptiveness and Superior Results. New York: McGraw Hill, 2010. Kindle.
- Rubinstein, Joshua S., David E. Meyer, and Jeffrey E. Evans. "Executive Control of Cognitive Processes in Task Switching." *Journal of Experimental Psychology: Human Perception and Performance* 27, no. 4 (2001): 763–797. http://www.umich.edu/~bcalab/documents/RubinsteinMeyer Evans2001.pdf.
- Senge, Peter M. The Fifth Discipline: The Art & Practice of the Learning Organization. New York: Doubleday, 2006.
- Sharwood, Simon. "Are Your Servers PETS or CATTLE?" The Register, March 18 2013. http://www .theregister.com/2013/03/18/servers_pets_or_cattle _cern/.
- Shingo, Shigeo. A Study of the Toyota Production System: From an Industrial Engineering Viewpoint. London: Productivity Press, 1989.
- Shinn, Bill. "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations." PostedbyGeneKimtoslideshare.net,November20,2015.Slideshow.http://www.slideshare.net/ ITRevolution/does15-bill-shinn-prove-it-the-last-mile-for-devops-in-regulated-organizations.
- Shook, John. "Five Missing Pieces in Your Standardized Work (Part 3 of 3)." Lean Enterprise Institute, October 27, 2009. http://www.lean.org/shook/DisplayObject.cfm?o=1321.
- Shoup, Randy. "Exploring the Uncharted Territory of Microservices." Posted by XebiaLabs, Inc., February 20, 2015. YouTube video, 56:50. https://www.youtube.com/watch?v=MRa21icSIQk.
- Shoup, Randy. "The Virtuous Cycle of Velocity: What I Learned About Going Fast at eBay and Google by Randy Shoup." Posted by Flowcon, December 26, 2013. YouTube video, 30:05. https://www .youtube.com/watch?v =EwLBoRyXTOI.
- Skinner, Chris. "Banks Have Bigger Development Shops than Microsoft." Chris Skinner's Blog, September 9, 2011. http://thefinanser.com/2011/09/banks-have-bigger-development -shops-than-microsoft.html/.
- Smart, Jonathan, Zsolt Berend, Myles Ogilvie, and Simon Rohrer. Sooner Safer Happier: Antipatterns and Patterns for Business Agility. Portland, OR: IT Revolution, 2020.
- Snyder, Ross. "Scaling Etsy: What Went Wrong, What Went Right." Posted by Ross Snyder to slideshare.net, October 5, 2011. Slideshow. http://www.slideshare.net/beamrider9/ scaling-etsy-what-went-wrong-what-went-right.
- Snyder, Ross. "Surge 2011—Scaling Etsy: What Went Wrong, What Went Right." Posted by OmniTi-Surge Conference, December 23, 2011. YouTube video, 37:17. https://www.youtube.com/ watch?v=eenrfm50mXw.
- Sonatype. 2015 State of the Software Supply Chain Report: Hidden Speed Bumps on the Way to "Continuous." Fulton, MD: Sonatype, Inc., 2015. http://cdn2.hubspot.net/hubfs/1958393/White_ Papers/2015_State_of_the_Software_Supply_Chain_Report-pdf?t=1466775053631.
- Sonatype. 2019 Stae of the Software Supply Chain Report. 2019. https://www.sonatype.com/ resources/white-paper-state-of-software-supply-chain-report-2019.
- Sonatype. 2020 State of the Software Supply Chain Report. 2020. https://www.sonatype.com/ resources/white-paper-state-of-the-software-supply-chain-2020.
- Sowell, Thomas. Basic Economics, Fifth Edition. New York: Basic Books, 2014.

Sowell, Thomas. Knowledge and Decisions. New York: Basic Books, 1980.

- Spear, Steven J. The High-Velocity Edge: How Market Leaders Leverage Operational Excellence to Beat the Competition. New York: McGraw Hill Education, 2009.
- Srivastava, Shivam, Kartik Trehan, Dilip Wagle, and Jane Wang. "Developer Velocity: How Software Excellence Fuels Business Performance." *McKinsey*, April 20, 2020. https://www .mckinsey.com/industries/technology-media-and-telecommunications/our-insights/ developer-velocity-how-software-excellence-fuels-business-performance.
- Staats, Bradley, and David M. Upton. "Lean Knowledge Work." Harvard Business Review, October 2011. https://hbr.org/2011/10/lean-knowledge-work.
- Strear, Chris. "Leadership Lessons Learned from Improving Flow in Hospital Settings using Theory of Constraints." Presentation at DevOps Enterprise Summit, Europe, 2021. https:// videolibrary.doesvirtual.com/?video=550704199.
- Stehr, Nico, and Reiner Grundmann. Knowledge: Critical Concepts, vol. 3. London: Routledge, 2005.
- Sterling, Bruce. "Scenius, or Communal Genius." Wired, June 16, 2008. https://www.wired .com/2008/06/scenius-or-comm/#:~:text=His%20actual%20definition%20is%3A%20 %22Scenius,scenius%2C%20you%20act%20like%20genius.
- Stillman, Jessica. "Hack Days: Not Just for Facebookers." *Inc.*, February 3, 2012. http://www.inc .com/jessica-stillman/hack-days-not-just-for-facebookers.html.
- Sussman, Noahand and Laura Beth Denker. "Divide and Conquer." Code as Craft (blog). Etsy, April 20, 2011. https://codeascraft.com/2011/04/20/divide-and-concur/.
- Sussna, Jeff. "From Design Thinking to DevOps and Back Again: Unifying Design & Operations." Posted by William Evans, June 5, 2015. Vimeo video, 21:19. https://vimeo.com/129939230.
- Takeuchi, Hirotaka, and Ikujiro Nonaka. "New Product Development Game." Harvard Business Review (January 1986): 137-146.
- Taleb, Nicholas. Antifragile: Things That Gain from Disorder (Incerto). New York: Random House, 2012.
- Target. "All About Target." A Bullseye View. Accessed June 9, 2016. https://corporate.target.com.
- Temple-Raston, Dina. "A 'Worst Nightmare' Cyberattack: The Untold Story of the SolarWinds Hack," NPR, April 16, 2021. https://www.npr.org/2021/04/16/985439655/a-worst-nightmarecyberattack-the-untold-story-of-the-solarwinds-hack.
- Thomas, John and Ashish Kumar. "Welcome to the Google Engineering Tools Blog." Google Engineering Tools blog, posted May 3, 2011. http://google-engtools.blogspot.com/2011/05/ welcome-to-google-engineering-tools.html.
- Townsend, Mark L. Review of *The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win,* by Gene Kim, Kevin Behr, and George Spafford. Amazon review, March 2, 2013. http:// uedata.amazon.com/gp/customer-reviews/R1097DFODM12VD/ref=cm_cr_getr_d_rvw _ttl?ie=UTF8&ASIN=B00VATFAMI.
- Treynor, Ben. "Keys to SRE." Presented at Usenix SREcon14, Santa Clara, CA, May 30, 2014. https:// www.usenix.org/conference/srecon14/technical-sessions/presentation/keys-sre.
- Tucci, Linda. "Four Pillars of PayPal's 'BigBang' Agile Transformation." *TechTarget*, August 2014. http://searchcio.techtarget.com/feature/Four-pillars-of-PayPals-big-bang-Agile-transformation.
- Turnbull, James. The Art of Monitoring. Seattle, WA: Amazon Digital Services, 2016. Kindle.

- Twitter Engineering. "Hack Week @ Twitter." Twitter blog, January 25, 2012. https://blog.twitter .com/2012/hack-week-twitter.
- Van Den Elzen, Scott. Review of The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win, by Gene Kim, Kevin Behr, and George Spafford. Amazon review, March 13, 2013. http://uedata.amazon.com/gp/customer-reviews/R2K95XEH5OL3Q5/ref=cm_cr_getr_d_ rvw_ttl?ie=UTF8&ASIN=B00VATFAMI.
- Van Leeuwen, Evelijn and Kris Buytaert. "DOES15—Evelijn Van Leeuwen and Kris Buytaert— Turning Around the Containership." Posted by DevOps Enterprise Summit, December 21, 2015. YouTube video, 30:28. https://www.youtube.com/watch?v=0GId4AMKvPc.
- Van Kemande, Ron. "Nothing Beats Engineering Talent: The Agile Transformation at ING." Presented at the DevOps Enterprise Summit, London, UK, June 30–July 1, 2016.
- Vance, Ashlee. "Inside Operation InVersion, the Code Freeze that Saved LinkedIn." Bloomberg, April 11, 2013. http://www.bloomberg.com/news/articles/2013-04-10/ inside-operation-inversion-the-code-freeze-that-saved-linkedin.
- Vance, Ashlee. "LinkedIn: A Story About Silicon Valley's Possibly Unhealthy Need for Speed." Bloomberg, April 30, 2013. http://www.bloomberg.com/articles/2013-04-29/linkedin-astory-about-silicon-valleys-possibly-unhealthy-need-for-speed.
- Vault. "Nordstrom, Inc." Company Profile. Accessed March 30, 2021. http://www.vault.com/ company-profiles/retail/nordstrom,-inc/company-overview.aspx.
- Velasquez, Nicole Forsgren, Gene Kim, Nigel Kersten, and Jez Humble. 2014 State of DevOps Report. Portland, OR: Puppet Labs, IT Revolution Press, and ThoughtWorks, 2014. https://services. google.com/fh/files/misc/state-of-devops-2014.pdf.
- Verizon Wireless. 2014 Data Breach Investigations Report. Verizon Enterprise Solutions, 2014. https://dti.delaware.gov/pdfs/rp_Verizon-DBIR-2014_en_xg.pdf.
- Verizon Wireless. 2021 Data Breach Investigations Report. Verizon, 2021. https://enterprise.verizon .com/resources/reports/2021-data-breach-investigations-report.pdf.
- "VPC Best Configuration Practices." Flux7 blog, January 23, 2014. http://blog .flux7.com/blogs/aws/vpc-best-configuration-practices.
- Walsh, Mark. "Ad Firms Right Media, AdInterax Sell to Yahoo." *MediaPost*, October 18, 2006. http:// www.mediapost.com/publications/article/49779/ad-firms-right-media-adinterax-sell-toyahoo.html.
- Wang, Kendrick. "Etsy's Culture Of Continuous Experimentation and A/B Testing Spurs Mobile Innovation." Apptimize blog, January 30, 2014. http://apptimize.com/blog/2014/01/ etsy-continuous-innovation-ab-testing/.
- "Weekly Top 10: Your DevOps Flavor." Electric Cloud, April 1, 2016. http://electric-cloud.com/ blog/2016/04/weekly-top-10-devops-flavor/.
- West, David. "Water scrum-fall is-reality_of_agile_for_most." Posted by harsoft to slideshare.net, April 22, 2013. Slideshow. http://www.slideshare.net/harsoft/ water-scrumfall-isrealityofagileformost.
- Westrum, Ron. "A Typology of Organisation Culture." *BMJ Quality & Safety* 13, no. 2 (2004): ii22–ii27. doi:10.1136/qshc.2003.009522.
- Westrum, Ron. "The Study of Information Flow: A Personal Journey." Proceedings of Safety Science 67 (August 2014): 58–63. https://www.researchgate.net/publication/ 261186680_The_study_of_information_flow_A_personal_journey.

- "What Happens to Companies That Get Hacked? FTC Cases." Posted by SuicidalSnowman to Giant Bomb forum, July 2012. http://www.giantbomb.com/forums/off-topic-31/what-happens-to-companies-that-get-hacked-ftc-case-540466/.
- "When will Google permit languages other than Python, C++, Java and Go to be used for internal projects?" Quora forum. Accessed May 29, 2016. https://www.quora.com/When-will-Google-permit-languages-other-than-Python-C-Java-and-Go-to-be-used-for-internalprojects/answer/Neil-Kandalgaonkar.
- "Which programming languages does Google use internally?" Quora forum. Accessed May 29, 2016. https://www.quora.com/Which-programming-languages-does-Google-use-internally.
- Wickett, James. "Attacking Pipelines—Security Meets Continuous Delivery." Posted by James Wickett to slideshare.net, June 11, 2014. Slideshow. http://www.slideshare.net/wickett/ attacking-pipelinessecurity-meets-continuous-delivery.
- Wiggins, Adams. "The Twelve-Factor App." 12Factor, January 30, 2012. http://12factor.net/.
- Wikipedia. "Direct Marketing." Wikipedia. Updated May 28, 2016. https://en.wikipedia.org/wiki/ Direct_marketing.
- Wikipedia. "Imposter Syndrome." Wikipedia. Updated November 17, 2020. https://en.wikipedia. org/wiki/Impostor_syndrome#:~:text=Impostor%20syndrome%20(also%20known%20 as,exposed%20as%20a%20%22fraud%22.
- Wikipedia. "Kaizen." Wikipedia. Updated May 12, 2016. https://en.wikipedia.org/wiki/Kaizen.
- Wikipedia. "Kolmogorov-Smirnov Test." Wikipedia. Updated May 19, 2016. http://en.wikipedia .org/wiki/Kolmogorov-Smirnov_test.
- Wikipedia. "Telemetry." Wikipedia. Updated May 5, 2016. https://en.wikipedia.org/wiki/Telemetry.
- Willis, John. "Docker and the Three Ways of DevOps Part 1: The First Way—Systems Thinking." Docker blog, May 26, 2015. https://blog.docker.com/2015/05/docker-three-ways-devops/.
- Winslow, Michael, Tamara Ledbetter, Adam Zimman, John Esser, Tim Judge, Carmen DeArdo. Change in a Successful Organization: Avoid Complacency by Making a Case for Continuous Improvemen. Portland, OR: IT Revolution, 2020. https://myresources.itrevolution.com/id006657108/ Change-in-a-Successful-Organization.
- Wiseman, Ben. 2021 Work Trend Index: Annual Report: The Next Great Disruption Is Hybrid Work— Are We Ready? Microsoft, March 22, 2021. https://ms-worklab.azureedge.net/files/reports/ hybridWork/pdf/2021_Microsoft_WTI_Report_March.pdf
- Womack, Jim. Gemba Walks. Cambridge, MA: Lean Enterprise Institute, 2011). Kindle.
- Wong, Bruce, and Christos Kalantzis. "A State of Xen—Chaos Monkey & Cassandra." The Netflix Tech Blog, October 2, 2014. http://techblog.netflix.com/2014/10/a-state-of-xen-chaosmonkey-cassandra.html.
- Wong, Eric. "Eric the Intern: The Origin of InGraphs." LinkedIn Engineering blog, June 30, 2011. http://engineering.linkedin.com/32/eric-intern-origin-ingraphs.
- Womack, James P., and Daniel T. Jones. Lean Thinking: Banish Waste and Create Wealth in Your Corporation. New York: Free Press, 2010.
- Zhao, Haiping. "HipHop for PHP: Move Fast." Facebook, February 2, 2010. https://www.facebook. com/notes/facebook-engineering/hiphop-for-php-move-fast/280583813919.
- Zia, Mossadeq, Gabriel Ramírez, and Noah Kunin. "Compliance Masonry: Building a Risk Management Platform, Brick by Brick." 18F blog, April 15, 2016. https://18f.gsa.gov/2016/04/15/ compliance-masonry-buildling-a-risk-management-platform/.

NOTES

Note from the Publisher

1. Kim, "State of DevOps: 2020 and Beyond."

Preface

1. Branden Williams, personal corresponence with the authors, 2015.

2. Christopher Little, personal correspondence with Gene Kim, 2010.

Introduction

- 1. Goldratt, Beyond the Goal.
- 2. Immelt, "Let's Finally End the Debate."
- 3. "Weekly Top 10: Your DevOps Flavor," Electric Cloud.
- 4. Goldratt, Beyond the Goal.
- 5. Spear, The High-Velocity Edge, Chapter 3.
- 6. Christopher Little, personal correspondence with Gene Kim, 2010.
- 7. Skinner, "Banks Have Bigger Development Shops than Microsoft."
- 8. Stehr and Grundmann, Knowledge, 139.
- 9. Masli et al., "Senior Executive's IT Management Responsibilities."
- 10. "IDC Forecasts Worldwide IT Spending to Grow 6%," Business Wire.
- 11. Kersten, IT Revolution, and PwC, 2015 State of DevOps Report.
- 12. Azzarello, Debruyne, and Mottura, "The Chemistry of Enthusiasm."
- 13. Brooks, *The Mythical Man-Month*.
- 14. Kim et al., "Exploring the Uncharted Territory of Microservices."
- 15. Kersten, IT Revolution, and PwC, 2015 State of DevOps Report.
- 16. Jenkins, "Velocity Culture"; Exner, "Transforming Software Development."
- 17. Goldratt, Beyond the Goal.
- JGFLL, review of The Phoenix Project; Townsend, review of The Phoenix Project; Van Den Elzen, review of The Phoenix Project.

Part I Introduction

- 1. Beck, et al., "Twelve Principles of Agile Software."
- 2. Rother, Toyota Kata, Part III.

Chapter 1

- 1. Martin and Osterling, Value Stream Mapping, Chapter 1.
- 2. Martin and Osterling, Value Stream Mapping, Chapter 3.
- 3. Martin and Osterling, Value Stream Mapping, Chapter 3.
- 4. Kersten, Project to Product.
- 5. Forsgren, Humble, and Kim, Accelerate 2018.
- 6. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 7. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 8. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 9. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 10. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 11. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 12. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 13. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 14. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."

Chapter 2

- 1. Rubinstein, Meyer, and Evans, "Executive Control of Cognitive Processes in Task Switching."
- 2. DeGrandis, "DOES15—Dominica DeGrandis—The Shape of Uncertainty."
- 3. Bahri, "Few Patients-In-Process and Less Safety Scheduling."
- Meeting between David J. Andersen and team at Motorola with Daniel S. Vacanti, February 24, 2004; story retold at USC CSSE Research Review with Barry Boehm in March 2004.
- 5. Womack and Jones, Lean Thinking. Chapter 1.
- 6. Ries, "Work in Small Batches."
- 7. Goldratt, Beyond the Goal.
- 8. Goldratt, The Goal, "Five Focusing Steps."
- 9. Shingo, A Study of the Toyota Production System.
- 10. Poppendieck and Poppendieck, Implementing Lean Software, 74.
- 11. Poppendieck and Poppendieck, Implementing Lean Software, Chapter 4.
- 12. Edwards, "DevOps Kaizen."
- 13. Strear, "Leadership Lessons Learned From Improving Flow."
- 14. Strear, "Leadership Lessons Learned From Improving Flow."
- 15. Strear, "Leadership Lessons Learned From Improving Flow."

Chapter 3

1. Perrow, Normal Accidents.

- 2. Dekker, The Field Guide to Understanding Human Error.
- 3. Spear, The High-Velocity Edge, Chapter 8.
- 4. Spear, The High-Velocity Edge, Chapter 8.
- 5. Senge, The Fifth Discipline, Chapter 5.
- 6. "NUMMI," This American Life.
- 7. Hendrickson, "DOES15—Elisabeth Hendrickson—Its All About Feedback."
- 8. Hendrickson, "DOES15-Elisabeth Hendrickson-Its All About Feedback."
- 9. Spear, The High-Velocity Edge, Chapter 1.
- 10. Spear, The High-Velocity Edge, Chapter 4.
- 11. Ayers and Cohen, "Andon Cords in Development Teams."
- 12. Ayers and Cohen, "Andon Cords in Development Teams."
- 13. Ayers and Cohen, "Andon Cords in Development Teams."
- 14. Jeff Gallimore, personal correspondence with the authors, 2021.
- 15. Sowell, Knowledge and Decisions, 222.
- 16. Sowell, Basic Economics.
- 17. Gary Gruver, personal correspondence with Gene Kim, 2014.

- 1. Adler, "Time-and-Motion Regained."
- 2. Dekker, The Field Guide to Understanding Human Error, Chapter 1.
- 3. Dekker, "Just Culture: Balancing Safety and Accountability."
- 4. Westrum, "The Study of Information Flow."
- 5. Westrum, "A Typology of Organisation Culture."
- 6. Velasquez et al., 2014 State of DevOps Report.
- 7. Macri, "Morgue."
- 8. Spear, The High-Velocity Edge, Chapter 1.
- 9. Senge, The Fifth Discipline, Chapter 1.
- 10. Rother, Toyota Kata, 12.
- 11. Mike Orzen, personal correspondence with Gene Kim, 2012.
- 12. "Paul O'Neill," Forbes.
- 13. Spear, The High-Velocity Edge, Chapter 4.
- 14. Spear, The High-Velocity Edge, Chapter 4.
- 15. Spear, The High-Velocity Edge, Chapter 4.
- 16. Spear, The High-Velocity Edge, Chapter 4.
- 17. Taleb, Antifragile.
- 18. Womack, Gemba Walks, Kindle location 4113.
- 19. Rother, Toyota Kata, Part IV.
- 20. Rother, Toyota Kata, Conclusion.
- 21. Winslow et al., Change in a Successful Organization.
- 22. Gertner, The Idea Factory.
- 23. Kersten, Project to Product; Kersten, "Project to Product: From Stories to Scenius."
- 24. Brian Eno, as quoted in Sterling, "Scenius, or Communal Genius."
- 25. Gertner, The Idea Factory.
- 26. Gertner, The Idea Factory.

- 1. Rembetsy and McDonnell, "Continuously Deploying Culture."
- 2. "Nordstrom, Inc.," Vault (website).
- 3. Kissler, "DOES14—Courtney Kissler—Nordstrom."
- 4. Gardner, "Barnes & Noble, Blockbuster, Borders."
- 5. Kissler, "DOES14—Courtney Kissler—Nordstrom."
- Kissler, "DOES14—Courtney Kissler—Nordstrom" [Alterations to quote made by Courtney Kissler via personal correspondence with Gene Kim, 2016.]
- Kissler, "DOES14—Courtney Kissler—Nordstrom" [Alterations to quote made by Courtney Kissler via personal correspondence with Gene Kim, 2016.]
- Kissler, "DOES14—Courtney Kissler—Nordstrom" [Alterations to quote made by Courtney Kissler via personal correspondence with Gene Kim, 2016.]
- Kissler, "DOES14—Courtney Kissler—Nordstrom" [Alterations to quote made by Courtney Kissler via personal correspondence with Gene Kim, 2016.]
- Kissler, "DOES14—Courtney Kissler—Nordstrom" [Alterations to quote made by Courtney Kissler via personal correspondence with Gene Kim, 2016.]
- 11. Mueller, "Business Model Driven Cloud Adoption."
- 12. Unpublished calculation by Gene Kim after the 2014 DevOps Enterprise Summit.
- 13. Kersten, IT Revolution, and PwC, 2015 State of DevOps Report.
- 14. Prugh, "DOES14: Scott Prugh, CSG."
- 15. Rembetsy and McDonnell, "Continuously Deploying Culture."
- 16. Golden, "What Gartner's Bimodal IT Model Means to Enterprise CIOs."
- 17. Furtado and Knausenberger, "The Air Force's Digital Journey in 12 Parsecs or Less."
- 18. Furtado and Knausenberger, "The Air Force's Digital Journey in 12 Parsecs or Less."
- 19. Furtado and Knausenberger, "The Air Force's Digital Journey in 12 Parsecs or Less."
- 20. Furtado and Knausenberger, "The Air Force's Digital Journey in 12 Parsecs or Less."
- 21. Furtado and Knausenberger, "The Air Force's Digital Journey in 12 Parsecs or Less."
- 22. Furtado and Knausenberger, "The Air Force's Digital Journey in 12 Parsecs or Less."
- 23. Furtado and Knausenberger, "The Air Force's Digital Journey in 12 Parsecs or Less."
- 24. Golden, "What Gartner's Bimodal IT Model Means to Enterprise CIOs."
- 25. Golden, "What Gartner's Bimodal IT Model Means to Enterprise CIOs."
- 26. Kersten, IT Revolution, and PwC, 2015 State of DevOps Report.
- 27. Scott Prugh, personal correspondence with Gene Kim, 2014.
- 28. Moore and McKenna, Crossing the Chasm, 11.
- 29. Tucci, "Four Pillars of PayPal's 'Big Bang' Agile Transformation."
- 30. Fernandez and Spear, "Creating High Velocity Organizations."
- 31. Van Kemande, "Nothing Beats Engineering Talent."
- 32. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 33. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 34. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 35. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 36. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 37. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 38. Leibman and Clanton, "DevOps: Approaching Cruising Altitude."
- 39. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."

- 40. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."
- 41. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."
- 42. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."
- 43. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."
- 44. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."
- 45. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."
- 46. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."
- 47. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."
- 48. Conrad and Hyatt, "Saving the Economy from Ruin (with a Hyperscale PaaS)."
- 49. Buchanan, "The Wisdom of Peter Drucker from A to Z."

- 1. Kissler, "DOES14—Courtney Kissler—Nordstrom."
- 2. Clanton and Ducy, interview of Courtney Kissler and Jason Josephy, "Continuous Improvement at Nordstrom."
- Clanton and Ducy, interview of Courtney Kissler and Jason Josephy, "Continuous Improvement at Nordstrom."
- Clanton and Ducy, interview of Courtney Kissler and Jason Josephy, "Continuous Improvement at Nordstrom."
- 5. Maskell, "What Does This Guy Do? Role of Value Stream Manager."
- 6. Edwards, "DevOps Kaizen."
- 7. Govindarajan and Trimble, The Other Side of Innovation.
- 8. Govindarajan and Trimble, The Other Side of Innovation, Part I.
- 9. Cagan, Inspired, 12
- 10. Cagan, Inspired, 12.
- 11. Vance, "LinkedIn."
- 12. Clemm, "LinkedIn Started Back in 2003."
- 13. Nielsen, "8 Years with LinkedIn."
- 14. Clemm, "LinkedIn Started Back in 2003."
- 15. Parikh, "From a Monolith to Microservices + REST."
- 16. Clemm, "LinkedIn Started back in 2003."
- 17. Vance, "LinkedIn."
- 18. "How I Structured Engineering Teams at LinkedIn and AdMob for Success," First Round Review.
- 19. Vance, "Inside Operation InVersion."
- 20. Vance, "LinkedIn."
- 21. Clemm, "LinkedIn Started Back in 2003."
- 22. "How I Structured Engineering Teams," First Round Review.
- 23. Christopher Little, personal correspondence with Gene Kim, 2011.
- 24. Ryan Martens, personal correspondence with Gene Kim, 2013.

- 1. Conway, "How Do Committees Invent?"
- 2. Conway, "How Do Committees Invent?"

- 3. Raymond, "Conway's Law."
- 4. Buhr, "Etsy Closes Up 86 Percent on First Day of Trading."
- 5. Snyder, "Scaling Etsy."
- 6. Snyder, "Scaling Etsy."
- 7. Gallagher, "When 'Clever' Goes Wrong."
- 8. Snyder, "Scaling Etsy."
- 9. Snyder, "Scaling Etsy."
- 10. Forsgren et al., Accelerate State of DevOps 2019.
- 11. Snyder, "Scaling Etsy."
- 12. Snyder, "Surge 2011."
- 13. Snyder, "Surge 2011."
- 14. Snyder, "Surge 2011."
- 15. McDonnell, "Continuously Deploying Culture."
- 16. Fernandez and Spear, "Creating High Velocity Organizations."
- 17. Adrian Cockcroft, personal correspondence with Gene Kim, 2014.
- 18. "A Conversation with Werner Vogels."
- Forsgren, Humble, and Kim, Accelerate State of DevOps 2018; Forsgren et al., Accelerate State of DevOps 2019..
- 20. Spear, The High-Velocity Edge, Chapter 8.
- 21. Rother, Toyota Kata, 250.
- 22. Mulkey, "DOES15-Jody Mulkey."
- 23. Mulkey, "DOES15—Jody Mulkey."
- 24. Canahuati, "Growing from the Few to the Many."
- 25. Spear, The High-Velocity Edge, Chapter 1.
- 26. Prugh, "Continuous Delivery."
- 27. Prugh, "Continuous Delivery."
- 28. Prugh, "Continuous Delivery."
- 29. Dweck, "Carol Dweck Revisits the 'Growth Mindset."
- 30. Cox, "Disney DevOps."
- 31. John Lauderbach, personal conversation with Gene Kim, 2001.
- 32. Mauro, "Adopting Microservices at Netflix"; Wiggins, "The Twelve-Factor App."
- 33. Shoup, "Exploring the Uncharted Territory of Microservices."
- 34. Humble, O'Reilly, and Molesky, Lean Enterprise, Part III.
- 35. Hastings, "Netflix Culture."
- 36. Dignan, "Little Things Add Up."
- 37. Mickman and Clanton, "DOES15—Heather Mickman & Ross Clanton."
- 38. Mickman and Clanton, "DOES15—Heather Mickman & Ross Clanton."
- 39. Mickman and Clanton, "DOES15—Heather Mickman & Ross Clanton."
- 40. Mickman and Clanton, "DOES15-Heather Mickman & Ross Clanton."
- 41. Mickman and Clanton, "DOES15-Heather Mickman & Ross Clanton."
- 42. Mickman and Clanton, "DOES15-Heather Mickman & Ross Clanton."
- 43. Mickman and Clanton, "DOES15-Heather Mickman & Ross Clanton."
- 44. Mickman and Clanton, "DOES15-Heather Mickman & Ross Clanton."

- 1. "Big Fish Celebrates 11th Consecutive Year of Record Growth," Big Fish Games (website).
- 2. Paul Farrall, personal correspondence with Gene Kim, January 2015.
- 3. Paul Farrall, personal correspondence with Gene Kim, 2014.
- 4. Paul Farrall, personal correspondence with Gene Kim, 2014.
- 5. Ernest Mueller, personal correspondence with Gene Kim, 2014.
- 6. Edwards, "DevOps Kaizen."
- 7. Marsh, "Dianne Marsh 'Introducing Change while Preserving Engineering Velocity."
- 8. Cox, "Disney DevOps."
- 9. Daniels, "Devopsdays Minneapolis 2015—Katherine Daniels—DevOps: The Missing Pieces."
- 10. Ernest Mueller, personal correspondence with Gene Kim, 2015.
- 11. Takeuchi and Nonaka, "New Product Development Game."
- 12. Chapman and Eltridge. "On A Mission: Nationwide Building Society."
- 13. Chapman and Eltridge. "On A Mission: Nationwide Building Society."
- 14. Chapman and Eltridge. "On A Mission: Nationwide Building Society."
- 15. Chapman and Eltridge. "On A Mission: Nationwide Building Society."
- 16. Chapman and Eltridge. "On A Mission: Nationwide Building Society."
- 17. Chapman and Eltridge. "On A Mission: Nationwide Building Society."
- 18. Chapman and Eltridge. "On A Mission: Nationwide Building Society."

- 1. Campbell-Pretty, "DOES14—Em Campbell-Pretty—How a Business Exec Led Agile, Lead, CI/CD."
- 2. Campbell-Pretty, "DOES14—Em Campbell-Pretty—How a Business Exec Led Agile, Lead, CI/CD."
- 3. Campbell-Pretty, "DOES14—Em Campbell-Pretty—How a Business Exec Led Agile, Lead, CI/CD."
- 4. Campbell-Pretty, "DOES14—Em Campbell-Pretty—How a Business Exec Led Agile, Lead, CI/CD."
- 5. Campbell-Pretty, "DOES14—Em Campbell-Pretty—How a Business Exec Led Agile, Lead, CI/CD."
- 6. Campbell-Pretty, "DOES14—Em Campbell-Pretty—How a Business Exec Led Agile, Lead, CI/CD."
- 7. Campbell-Pretty, "DOES14—Em Campbell-Pretty—How a Business Exec Led Agile, Lead, CI/CD."
- 8. "Version Control History," Plastic SCM (website).
- 9. Davis and Daniels, Effective DevOps, 37.
- 10. Velasquez et al., 2014 State of DevOps Report.
- 11. Sharwood, "Are Your Servers PETS or CATTLE?"
- 12. Chan, "OWASP AppSecUSA 2012."
- 13. Fowler, "Trash Your Servers and Burn Your Code."
- 14. Willis, "Docker and the Three Ways of DevOps Part 1."
- 15. Forsgren et al., 2020 State of the Octoverse.
- 16. Holmes, " How A Hotel Company Ran \$30B of Revenue in Containers."
- 17. Holmes, " How A Hotel Company Ran \$30B of Revenue in Containers."
- 18. Holmes, " How A Hotel Company Ran \$30B of Revenue in Containers."
- 19. Holmes, " How A Hotel Company Ran \$30B of Revenue in Containers."
- 20. Holmes, " How A Hotel Company Ran \$30B of Revenue in Containers."
- 21. Holmes, " How A Hotel Company Ran \$30B of Revenue in Containers."
- 22. Holmes, " How A Hotel Company Ran \$30B of Revenue in Containers."
- 1. Gary Gruver, personal correspondence with Gene Kim, 2014.
- 2. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 3. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 4. "Imposter Syndrome," Wikipedia."
- 5. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 6. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 7. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 8. Potvin and Levenber, "Why Google Stores Billions of Lines of Codes in a Single Repository."
- 9. Messeri, "What Goes Wrong When Thousands of Engineers Share the Same Continuous Build?"
- 10. Messeri, "What Goes Wrong When Thousands of Engineers Share the Same Continuous Build?"
- 11. Potvin and Levenber, "Why Google Stores Billions of Lines of Codes in a Single Repository."
- 12. Potvin and Levenber, "Why Google Stores Billions of Lines of Codes in a Single Repository"; Messeri, "What Goes Wrong When Thousands of Engineers Share the Same Continuous Build?"
- 13. Jez Humble and David Farley, personal correspondence with Gene Kim, 2012.
- 14. Humble and Farley, Continuous Delivery, 3.
- 15. Humble and Farley, Continuous Delivery, 188.
- 16. Humble and Farley, Continuous Delivery, 258.
- 17. Fowler, "Continuous Integration."
- 18. Fowler, "Test Pyramid."
- 19. Fowler, "Test Driven Development."
- 20. Nagappan et al., "Realizing Quality Improvement through Test Driven Development."
- 21. Hendrickson, "On the Care and Feeding of Feedback Cycles."
- 22. "Decreasing False Positives in Automated Testing."; Fowler, "Eradicating Non-determinism in Tests."
- 23. Gruver, "DOES14—Gary Gruver—Macy's—Transforming Traditional Enterprise Software Development Processes."
- 24. Jones, "3 Ways to Get Test Automation Done Within Your Sprints."
- 25. Shoup, "The Virtuous Cycle of Velocity."
- 26. West, "Water scrum-fall is-reality_of_agile_for_most."
- 27. Forsgren et al., Accelerate: State of DevOps 2019.
- 28. Forsgren, Humble, and Kim, Accelerate: State of DevOps 2018.

- 1. Kim, "The Amazing DevOps Transformation of the HP LaserJet Firmware Team."
- 2. Kim, "The Amazing DevOps Transformation of the HP LaserJet Firmware Team."
- 3. Kim, "The Amazing DevOps Transformation of the HP LaserJet Firmware Team."
- 4. Kim, "The Amazing DevOps Transformation of the HP LaserJet Firmware Team."
- 5. Kim, "The Amazing DevOps Transformation of the HP LaserJet Firmware Team."
- 6. Gruver and Mouser, Leading the Transformation, 60.
- 7. Gary Gruver, personal communication with the authors, 2016.
- 8. Kim, "The Amazing DevOps Transformation of the HP LaserJet Firmware Team."
- 9. Kim, "The Amazing DevOps Transformation of the HP LaserJet Firmware Team."
- 10. Kim, "The Amazing DevOps Transformation of the HP LaserJet Firmware Team."

- 11. Kim, "The Amazing DevOps Transformation of the HP LaserJet Firmware Team."
- 12. Atwood, "Software Branching and Parallel Universes."
- 13. Cunningham, "Ward Explains Debt Metaphor."
- 14. Mueller, "2012: A Release Odyssey."
- 15. "Bazaarvoice, Inc. Announces Its Financial Results," Bazaar Voice (website).
- 16. Mueller, "DOES15—Ernest Mueller—DevOps Transformations At National Instruments."
- 17. Mueller, "DOES15—Ernest Mueller—DevOps Transformations At National Instruments."
- 18. Mueller, "DOES15—Ernest Mueller—DevOps Transformations At National Instruments."
- 19. Mueller, "DOES15—Ernest Mueller—DevOps Transformations At National Instruments."
- 20. Mueller, "DOES15—Ernest Mueller—DevOps Transformations At National Instruments"
- 21. Kersten, IT Revolution, and PwC, 2015 State of DevOps Report.
- 22. Brown, et al., State of DevOps Report; Forsgren et al., State of DevOps Report 2017.

- 1. Rossi, "Release Engineering and Push Karma."
- 2. Paul, "Exclusive: A Behind-the-Scenes Look at Facebook Release Engineering."
- 3. Rossi, "Release Engineering and Push Karma."
- 4. Paul, "Exclusive: a Behind-the-Scenes Look at Facebook Release Engineering."
- 5. Rossi, "Ship early and ship twice as often."
- 6. Beck, "Slow Deployment Causes Meetings."
- 7. Prugh, "DOES14: Scott Prugh, CSG—DevOps and Lean in Legacy Environments."
- 8. Prugh, "DOES14: Scott Prugh, CSG—DevOps and Lean in Legacy Environments."
- 9. Prugh, "DOES14: Scott Prugh, CSG—DevOps and Lean in Legacy Environments."
- 10. Prugh, "DOES14: Scott Prugh, CSG-DevOps and Lean in Legacy Environments."
- 11. Prugh, "DOES14: Scott Prugh, CSG—DevOps and Lean in Legacy Environments."
- 12. Prugh, "DOES14: Scott Prugh, CSG-DevOps and Lean in Legacy Environments."
- 13. Puppet Labs and IT Revolution Press, 2013 State of DevOps Report.
- Prugh and Morrison, "DOES15—Scott Prugh & Erica Morrison—Conway & Taylor Meet the Strangler (v2.0)."
- Prugh and Morrison, "DOES15—Scott Prugh & Erica Morrison—Conway & Taylor Meet the Strangler (v2.0)."
- Prugh and Morrison, "DOES15—Scott Prugh & Erica Morrison—Conway & Taylor Meet the Strangler (v2.0)."
- 17. Tim Tischler, personal conversation with Gene Kim, FlowCon 2013.
- 18. Puppet Labs and IT Revolution Press, 2013 State of DevOps Report.
- 19. Forsgren et al. Accelerate: State of DevOps 2019.
- 20. Dickerson, "Optimizing for Developer Happiness."
- 21. Sussman and Denker, "Divide and Conquer."
- 22. Sussman and Denker, "Divide and Conquer."
- 23. Sussman and Denker, "Divide and Conquer."
- 24. Sussman and Denker, "Divide and Conquer."
- 25. Sussman and Denker, "Divide and Conquer."
- 26. Kastner, "Quantum of Deployment."
- 27. Fitz, "Continuous Deployment at IMVU."

- Fitz, "Continuous Deployment at IMVU"; Hrenko, "DOES15—Michael Hrenko—DevOps Insured By Blue Shield of California."
- 29. Humble and Farley, Continuous Delivery, 265.
- 30. Ries, The Lean Startup.
- 31. Bosworth, "Building and testing at Facebook"; "Etsy's Feature Flagging," GitHub (website).
- 32. Allspaw, "Convincing Management."
- 33. Rossi, "Release Engineering and Push Karma."
- 34. Protalinski, "Facebook Passes 1.55B Monthly Active Asers."
- 35. Protalinski, "Facebook Passes 1.55B Monthly Active Users."
- 36. Letuchy, "Facebook Chat."
- 37. Letuchy, "Facebook Chat."
- 38. Letuchy, "Facebook Chat."
- 39. Jez Humble, personal correspondence with Gene Kim, 2014.
- 40. Jez Humble, personal correspondence with Gene Kim, 2014.
- 41. Jez Humble, personal correspondence with Gene Kim, 2014.
- Forsgren, Humble, and Kim, Accelerate State of DevOps 2018; Forsgren et al., Accelerate State of DevOps 2019.
- 43. Prugh and Morrison, "When Ops Swallows Dev."
- 44. Prugh and Morrison, "When Ops Swallows Dev."
- 45. Prugh and Morrison, "When Ops Swallows Dev."
- 46. Prugh and Morrison, "When Ops Swallows Dev."
- 47. Prugh and Morrison, "When Ops Swallows Dev."
- 48. Prugh and Morrison, "When Ops Swallows Dev."

- 1. Humble, "What is Continuous Delivery?"
- 2. Kim et al., "Exploring the Uncharted Territory of Microservices."
- 3. Kim et al., "Exploring the Uncharted Territory of Microservices."
- 4. Kim et al., "Exploring the Uncharted Territory of Microservices."
- 5. Shoup, "From Monolith to Microservices."
- 6. Betz, Architecture and Patterns for IT Service Management, 300.
- 7. Shoup, "From Monolith to Micro-services."
- 8. Shoup, "From Monolith to Micro-services."
- 9. Shoup, "From Monolith to Micro-services."
- 10. Vogels, "A Conversation with Werner Vogels."
- 11. Vogels, "A Conversation with Werner Vogels."
- 12. Vogels, "A Conversation with Werner Vogels."
- 13. Vogels, "A Conversation with Werner Vogels."
- 14. Jenkins, "Velocity Culture."
- 15. Exner, "Transforming Software Development."
- 16. Fowler, "Strangler Fig Application."
- 17. Lublinsky, "Versioning in SOA."
- 18. Hammant, "Introducing Branch by Abstraction."
- 19. Fowler, "Strangler Fig Application."
- 20. Huang, "Blackboard CEO Jay Bhatt on the Global Future of Edtech."

- 21. Ashman, "DOES14—David Ashman—Blackboard Learn—Keep Your Head in the Clouds."
- 22. Ashman, "DOES14-David Ashman-Blackboard Learn-Keep Your Head in the Clouds."
- 23. Ashman, "DOES14—David Ashman—Blackboard Learn—Keep Your Head in the Clouds."
- 24. Ashman, "DOES14-David Ashman-Blackboard Learn-Keep Your Head in the Clouds."
- 25. Ashman, "DOES14—David Ashman—Blackboard Learn—Keep Your Head in the Clouds."
- 26. Forsgren et al., State of DevOps Report 2017.
- Forsgren, Humble, and Kim, Accelerate: State of DevOps 2018; Forsgren et al., Accelerate State of DevOps 2019.

- 1. Kim, Behr, and Spafford, The Visible Ops Handbook, Introduction.
- 2. Kim, Behr, and Spafford, The Visible Ops Handbook, Introduction.
- 3. Kim, Behr, and Spafford, The Visible Ops Handbook, Introduction.
- 4. "Telemetry," Wikipedia.
- 5. Rembetsy and McDonnell, "Continuously Deploying Culture."
- 6. Rembetsy and McDonnell, "Continuously Deploying Culture."
- 7. John Allspaw, personal conversation with Gene Kim, 2014.
- 8. Malpass, "Measure Anything, Measure Everything."
- 9. Kersten, IT Revolution, and PwC, 2015 State of DevOps Report.
- 10. Forsgren et al., Accelerate: State of DevOps 2019.
- 11. Turnbull, The Art of Monitoring, Introduction.
- 12. Cockcroft, "Monitorama."
- 13. Prugh, "DOES14: Scott Prugh, CSG-DevOps and Lean in Legacy Environments."
- 14. Figureau, "The 10 Commandments of Logging."
- 15. Dan North, personal correspondence with Gene Kim, 2016.
- 16. Chuvakin, "LogLogic/Chuvakin Log Checklist."
- 17. Kim, Behr, and Spafford, The Visible Ops Handbook, Introduction.
- 18. North, "Ops and Operability."
- 19. John Allspaw, personal correspondence with Gene Kim, 2011.
- 20. Agile Alliance, "Information Radiators."
- 21. Ernest Mueller, personal correspondence with Gene Kim, 2014.
- 22. Gupta, "Visualizing LinkedIn's Site Performance."
- 23. Wong, "Eric the Intern."
- 24. Wong, "Eric the Intern."
- 25. Wong, "Eric the Intern."
- 26. Ed Blankenship, personal correspondence with Gene Kim, 2016.
- 27. Burrows, "The Chubby Lock Service for Loosely-Coupled Distributed Systems."
- 28. Lindsay, "Consul Service Discovery with Docker."
- 29.. Mulkey, "DOES15—Jody Mulkey—DevOps in the Enterprise: A Transformation Journey."
- 30. Forsgren et al., Accelerate: State of DevOps 2019.

Chapter 15

1. Netflix Letter to Shareholders.

- 2. Roy Rapoport, personal correspondence with Gene Kim, 2014.
- 3. Hodge and Austin, "A Survey of Outlier Detection Methodologies."
- 4. Roy Rapoport, personal correspondence with Gene Kim, 2014.
- 5. Roy Rapoport, personal correspondence with Gene Kim, 2014.
- 6. Roy Rapoport, personal correspondence with Gene Kim, 2014.
- 7. Boubez, "Simple Math for Anomaly Detection."
- 8. Limoncelli, "Stop Monitoring Whether or Not Your Service Is Up!."
- 9. Boubez, "Simple Math for Anomaly Detection."
- 10. Dr. Nicole Forsgren, personal correspondence with Gene Kim, 2015.
- 11. Jacobson, Yuan, and Joshi, "Scryer: Netflix's Predictive Auto Scaling Engine."
- 12. Jacobson, Yuan, and Joshi, "Scryer: Netflix's Predictive Auto Scaling Engine."
- 13. Jacobson, Yuan, and Joshi, "Scryer: Netflix's Predictive Auto Scaling Engine."
- 14. Chandola, Banerjee, and Kumar, "Anomaly Detection: A Survey."
- 15. Tarun Reddy, personal interview with Gene Kim, Rally headquarters, Boulder, CO, 2014.
- 16. "Kolmogorov-Smirnov Test," Wikipedia.
- 17. Boubez, "Simple Math for Anomaly Detection."
- 18. Boubez, "Simple Math for Anomaly Detection."

- 1. Walsh, "Ad Firms Right Media."
- 2. Nick Galbreath, personal conversation with Gene, 2013.
- 3. Galbreath, "Continuous Deployment."
- 4. Galbreath, "Continuous Deployment."
- 5. Galbreath, "Continuous Deployment."
- 6. Canahuati, "Growing from the Few to the Many."
- 7. Lightbody, "From Inception to Acquisition."
- 8. Chakrabarti, "Common Ops Mistakes."
- 9. Sussna, "From Design Thinking to DevOps and Back Again."
- 10. Anonymous, personal conversation with Gene Kim, 2005.
- 11. Limoncelli, "SRE@Google."
- 12. Treynor, "Keys to SRE."
- 13. Limoncelli, "SRE@Google."
- 14. Limoncelli, "SRE@Google."
- 15. Limoncelli, "SRE@Google."
- 16. Tom Limoncelli, personal correspondence with Gene Kim, 2016.
- 17. Tom Limoncelli, personal correspondence with Gene Kim, 2016.

- 1. Humble, O'Reilly and Molesky, Lean Enterprise, Part II.
- 2. Intuit, Inc., "2012 Annual Report."
- 3. Cook, "Leadership in an Agile Age."
- 4. Cook, "Leadership in an Agile Age."
- 5. Cook, "Leadership in an Agile Age."

- 6. "Direct Marketing," Wikipedia.
- 7. "Fighting Poverty With Actual Evidence: Full Transcript," Freakonomics (blog).
- 8. Kohavi, Crook, and Longbotham, "Online Experimentation at Microsoft."
- 9. Kohavi, Crook, and Longbotham, "Online Experimentation at Microsoft."
- 10. Jez Humble, personal correspondence with Gene Kim, 2015.
- 11. Wang, "Etsy's Culture Of Continuous Experimentation."
- 12. O'Reilly, "How to Implement Hypothesis-Driven Development."
- 13. Kim, "Organizational Learning and Competitiveness."
- 14. Kim, "Organizational Learning and Competitiveness."
- 15. Kim, "Organizational Learning and Competitiveness."
- 16. Kim, "Organizational Learning and Competitiveness."
- 17. Kim, "Organizational Learning and Competitiveness."
- 18. Kim, "Organizational Learning and Competitiveness."
- 19. Kim, "Organizational Learning and Competitiveness."

- 1. Chacon, "GitHub Flow."
- 2. Douglas, "Deploying at GitHub."
- 3. Allspaw, "Counterfactual Thinking, Rules, and the Knight Capital Accident."
- 4. Allspaw, "Counterfactual Thinking, Rules, and the Knight Capital Accident."
- 5. Staats and Upton, "Lean Knowledge Work."
- 6. Forsgren et al., Accelerate State of DevOps 2019.
- 7. Forsgren et al., Accelerate State of DevOps 2019.
- 8. Velasquez et al., 2014 State of DevOps Report.
- 9. Randy Shoup, personal interview with Gene Kim, 2015.
- 10. Özil, "Ask a programmer."
- 11. Cornago, Yadav, and Otto, "From 6-Eye Principle to Release at Scale adidas Digital Tech 2021."
- 12. Cornago, Yadav, and Otto, "From 6-Eye Principle to Release at Scale adidas Digital Tech 2021."
- 13. Cornago, Yadav, and Otto, "From 6-Eye Principle to Release at Scale adidas Digital Tech 2021."
- 14. Cornago, Yadav, and Otto, "From 6-Eye Principle to Release at Scale adidas Digital Tech 2021."
- 15. Cornago, Yadav, and Otto, "From 6-Eye Principle to Release at Scale adidas Digital Tech 2021."
- 16. Cornago, Yadav, and Otto, "From 6-Eye Principle to Release at Scale adidas Digital Tech 2021."
- 17. Cornago, Yadav, and Otto, "From 6-Eye Principle to Release at Scale adidas Digital Tech 2021."
- 18. Cornago, Yadav, and Otto, "From 6-Eye Principle to Release at Scale adidas Digital Tech 2021."
- 19. Cornago, Yadav, and Otto, "From 6-Eye Principle to Release at Scale adidas Digital Tech 2021."
- 20. Özil, "Ask a programmer to review 10 lines of code.".
- 21. Messeri, "What Goes Wrong When Thousands of Engineers Share the Same Continuous Build?"
- 22. Thomas and Kumar, "Welcome to the Google Engineering Tools Blog."
- 23. Kumar, "Development at the Speed and Scale of Google."
- 24. Randy Shoup, personal correspondence with Gene Kim, 2014.
- 25. Atwood, "Pair Programming vs. Code Reviews."
- 26. Atwood, "Pair Programming vs. Code Reviews."
- 27. "Pair Programming," ALICE Wiki page.
- 28. "Pair Programming," ALICE Wiki page.
- 29. Hendrickson, "DOES15—Elisabeth Hendrickson—Its All About Feedback."

- 30. Hendrickson, "DOES15-Elisabeth Hendrickson-Its All About Feedback."
- 31. Hendrickson, "DOES15—Elisabeth Hendrickson—Its All About Feedback."
- 32. Hendrickson, "DOES15—Elisabeth Hendrickson—Its All About Feedback."
- 33. Hendrickson, "DOES15—Elisabeth Hendrickson—Its All About Feedback."
- 34. Ryan Tomayko, personal correspondence with Gene Kim, 2014.
- 35. Ryan Tomayko, personal correspondence with Gene Kim, 2014.
- 36. Ryan Tomayko, personal correspondence with Gene Kim, 2014.
- 37. Ryan Tomayko, personal correspondence with Gene Kim, 2014.
- 38. Ryan Tomayko, personal correspondence with Gene Kim, 2014.
- 39. Cockcroft, Ducy, and Clanton, "Adrian Cockcroft of Battery Ventures."
- 40. Pal, "DOES15—Tapabrata Pal—Banking on Innovation & DevOps."
- 41. Cox, "Disney DevOps."
- 42. Clanton and Mickman, "DOES14-Ross Clanton and Heather Mickman-DevOps at Target."
- 43. Clanton and Mickman, "DOES14-Ross Clanton and Heather Mickman-DevOps at Target."
- 44. Clanton and Mickman, "DOES14—Ross Clanton and Heather Mickman—DevOps at Target."
- 45. Clanton and Mickman, "DOES14-Ross Clanton and Heather Mickman-DevOps at Target."
- 46. Clanton and Mickman, "DOES14—Ross Clanton and Heather Mickman—DevOps at Target."
- 47. John Allspaw and Jez Humble, personal correspondence with Gene Kim, 2014.

- 1. Spear, The High-Velocity Edge, Chapter 1.
- 2. Spear, The High-Velocity Edge, Chapter 1.
- 3. Pepitone, "Amazon EC2 Outage Downs Reddit, Quora."
- 4. Morgan, "A Rare Peek into the Massive Scale of AWS."
- 5. Cockcroft, Hicks, and Orzell, "Lessons Netflix Learned from the AWS Outage."
- 6. Cockcroft, Hicks, and Orzell, "Lessons Netflix Learned from the AWS Outage."
- 7. Cockcroft, Hicks, and Orzell, "Lessons Netflix Learned from the AWS Outage."
- 8. Dekker, "Just Culture," 152.
- 9. Dekker, "DevOpsDays Brisbane 2014—Sidney Decker—System Failure, Human Error: Who's to Blame?"
- 10. Allspaw, "Post-Mortems, Sans Finger-Pointing."
- 11. Allspaw, "Blameless PostMortems and a Just Culture."
- 12. Malpass, "DevOpsDays Minneapolis 2014-Ian Malpass, Fallible Humans."
- 13. Milstein, "Post-Mortems at HubSpot: What I Learned from 250 Whys."
- 14. Randy Shoup, personal correspondence with Gene Kim, 2014.
- Google, "Post-Mortem for February 24, 2010 Outage"; Amazon Web Services, "Summary of the Amazon DynamoDB Service Disruption and Related Impacts in the US-East Region."
- 16. Macri, "Morgue."
- 17. Macri, "Morgue."
- 18. Forsgren, Humble, and Kim, Accelerate: State of DevOps 2018.
- 19. Edmondson, "Strategies for Learning from Failure."
- 20. Spear, The High-Velocity Edge, Chapter 4.
- 21. Spear, The High-Velocity Edge, Chapter 4.
- 22. Spear, The High-Velocity Edge, Chapter 3.
- 23. Roberto, Bohmer, and Edmondson, "Facing Ambiguous Threats."

- 24. Roberto, Bohmer, and Edmondson, "Facing Ambiguous Threats."
- 25. Roberto, Bohmer, and Edmondson, "Facing Ambiguous Threats."
- 26. Roy Rapoport, personal correspondence with Gene Kim, 2012.
- 27. Roy Rapoport, personal correspondence with Gene Kim, 2012.
- 28. Roy Rapoport, personal correspondence with Gene Kim, 2012.
- 29. Nygard, Release It!, Part I.
- 30. Barr, "EC2 Maintenance Update."
- 31. Wong and Kalantzis, "A State of Xen—Chaos Monkey & Cassandra."
- 32. Wong and Kalantzis, "A State of Xen—Chaos Monkey & Cassandra."
- 33. Wong and Kalantzis, "A State of Xen—Chaos Monkey & Cassandra."
- 34. Roy Rapoport, personal correspondence with Gene Kim, 2015.
- 35. Adrian Cockcroft, personal correspondence with Gene Kim, 2012.
- 36. Robbins, "GameDay."
- 37. Robbins et al., "Resilience Engineering."
- 38. Robbins et al., "Resilience Engineering."
- 39. Robbins et al., "Resilience Engineering."
- 40. Robbins et al., "Resilience Engineering."
- 41. Robbins et al., "Resilience Engineering."
- 42. Krishman, "Learning Continuously From Failures' at Google."
- 43. Krishnan, "Weathering the Unexpected."
- 44. Krishnan, "Weathering the Unexpected."
- 45. Morrison, "How We Turned Our Company's Worst Outage into a Powerful Learning Opportunity."
- 46. Widely attributed to Peter Senge.

- 1. Newland, "ChatOps at GitHub."
- 2. Newland, "ChatOps at GitHub."
- 3. Mark Imbriaco, personal correspondence with Gene Kim, 2015.
- 4. Newland, "ChatOps at GitHub."
- 5. Newland, "ChatOps at GitHub."
- 6. Newland, "ChatOps at GitHub."
- 7. Newland, "ChatOps at GitHub."
- 8. Osterweil, "Software Processes are Software Too."
- 9. Arbuckle, "What Is ArchOps: Chef Executive Roundtable."
- 10. Arbuckle, "What Is ArchOps: Chef Executive Roundtable."
- 11. Arbuckle, "What Is ArchOps: Chef Executive Roundtable."
- 12. Metz, "Google Is 2 Billion Lines of Code—and It's All in One Place."
- 13. Metz, "Google Is 2 Billion Lines of Code—and It's All in One Place."
- 14. Metz, "Google Is 2 Billion Lines of Code—and It's All in One Place."
- 15. Messeri, "What Goes Wrong When Thousands of Engineers Share the Same Continuous Build?"
- 16. Randy Shoup, personal correspondence with Gene Kim, 2014.
- 17. Limoncelli, "Yes, You Can Really Work from HEAD."
- 18. Forsgren et al., Accelerate: State of DevOps 2019.
- 19. Forsgren et al., Accelerate: State of DevOps 2019.

- 20. Mell and Grance, The NIST Definition of Cloud Computing, 6.
- 21. Forsgren et al., Accelerate: State of DevOps 2019.
- 22. Loura, Jacques, and Garcia, "DOES15—Ralph Loura, Olivier Jacques, & Rafael Garcia—Breaking Traditional IT Paradigms."
- 23. Rembetsy and McDonnell, "Continuously Deploying Culture."
- 24. Rembetsy and McDonnell, "Continuously Deploying Culture."
- 25. McKinley, "Why MongoDB Never Worked Out at Etsy."
- 26. Cundiff, Geinert, and Rettig, "Crowdsourcing Technology Governance."
- 27. Cundiff, Geinert, and Rettig, "Crowdsourcing Technology Governance."
- 28. Cundiff, Geinert, and Rettig, "Crowdsourcing Technology Governance."
- 29. Cundiff, Geinert, and Rettig, "Crowdsourcing Technology Governance."
- 30. Cundiff, Geinert, and Rettig, "Crowdsourcing Technology Governance."
- 31. Cundiff, Geinert, and Rettig, "Crowdsourcing Technology Governance."

- 1. "Kaizen," Wikipedia.
- 2. Spear, The High-Velocity Edge, Chapter 8.
- 3. Spear, The High-Velocity Edge, Chapter 8.
- 4. Mickman and Clanton, "(Re)building an Engineering Culture."
- 5. Ravi Pandey, personal correspondence with Gene Kim, 2015.
- 6. Mickman and Clanton, "(Re)building an Engineering Culture."
- 7. Pomeranz, "Queue Inversion Week."
- 8. Spear, The High-Velocity Edge, Chapter 3.
- 9. Stillman, "Hack Days."
- 10. Associated Press, "Number of Active Users at Facebook over the Years."
- 11. Zhao, "HipHop for PHP."
- 12. Metz, "How Three Guys Rebuilt the Foundation of Facebook."
- 13. Metz, "How Three Guys Rebuilt the Foundation of Facebook."
- 14. Steve Farley, personal correspondence with Gene Kim, January 5, 2016.
- 15. Gaekwad, "Agile 2013 Talk."
- O'Donnell, "DOES14—Glenn O'Donnell—Forrester—Modern Services Demand a DevOps Culture Beyond Apps."
- 17. Smart et al., Sooner Safer Happier, 314.
- 18. Nationwide, 2014 Annual Report.
- 19. Steve Farley, personal correspondence with Gene Kim, 2016.
- 20. Pal, "DOES15—Tapabrata Pal—Banking on Innovation & DevOps."
- 21. Pal, "DOES15-Tapabrata Pal-Banking on Innovation & DevOps."
- 22. Tapabrata Pal, personal correspondence with Gene Kim, 2015.
- 23. "All About Target," Target (website).
- 24. Mickman and Clanton, "(Re)building an Engineering Culture."
- Van Leeuwen and Buytaert, "DOES15—Evelijn Van Leeuwen and Kris Buytaert—Turning Around the Containership."
- 26. Mickman and Clanton, "(Re)building an Engineering Culture."
- 27. "DevOps Culture: How to Transform."
- 28. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."

- 29. Bland, "DOES15-Mike Bland-Pain Is Over, If You Want It."
- 30. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 31. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 32. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 33. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 34. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 35. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 36. Bland, "Fixits, or I Am the Walrus."
- 37. Bland, "Fixits, or I Am the Walrus."
- 38. Bland, "Fixits, or I Am the Walrus."

- 1. Wickett, "Attacking Pipelines-Security Meets Continuous Delivery."
- 2. Wickett, "Attacking Pipelines-Security Meets Continuous Delivery."
- 3. Pal, "DOES15—Tapabrata Pal—Banking on Innovation & DevOps."
- 4. Justin Arbuckle, personal interview with Gene Kim, 2015.
- 5. Justin Arbuckle, personal interview with Gene Kim, 2015.
- 6. Antani, "IBM Innovate DevOps Keynote."
- 7. Galbreath, "DevOpsSec: Applying DevOps Principles to Security, DevOpsDays Austin 2012."
- 8. Galbreath, "DevOpsSec: Applying DevOps Principles to Security, DevOpsDays Austin 2012."
- 9. "OWASP Cheat Sheet Series," OWASP (website).
- 10. Collins, Smolen, and Matatall, "Putting to your Robots to Work V1.1."
- 11. "What Happens to Companies That Get Hacked? FTC Cases," Giant Bomb forum.
- 12. "What Happens to Companies That Get Hacked? FTC Cases," Giant Bomb forum.
- 13. Collins, Smolen, and Matatall, "Putting to your Robots to Work V1.1."
- 14. Twitter Engineering, "Hack Week @ Twitter."
- 15. Twitter Engineering, "Hack Week @ Twitter."
- Corman and Willis, "Immutable Awesomeness—Josh Corman and John Willis at DevOps Enterprise Summit 2015."
- 17. Forsgren et al., 2020 State of the Octoverse.
- 18. Forsgren et al., 2020 State of the Octoverse.
- 19. Verison, 2014 Data Breach Investigations Report.
- 20. Verizon, 2021 Data Breach Investigations Report, 20.
- 21. Sonatype, 2019 State of the Software Supply Chain Report.
- 22. Sonatype, 2019 State of the Software Supply Chain Report.
- 23. Sonatype, 2019 State of the Software Supply Chain Report.
- 24. "Jeremy Long: The (Application)Patching Manifesto."
- 25. "Jeremy Long: The (Application)Patching Manifesto."
- 26. Sonatype, 2019 State of the Software Supply Chain Report.
- 27. Sonatype, 2019 State of the Software Supply Chain Report.
- 28. Sonatype, 2019 State of the Software Supply Chain Report.
- 29. Sonatype, 2019 State of the Software Supply Chain Report.
- 30. Sonatype, 2020 State of the Software Supply Chain Report.
- 31. Sonatype, 2020 State of the Software Supply Chain Report.
- 32. Geer and Corman, "Almost Too Big to Fail."

- 33. Forsgren et al., 2020 State of the Octoverse.
- 34. Temple-Raston, "A 'Worst Nightmare' Cyberattack."
- 35. Naraine, "Twilio, HashiCorp Among Codecov Supply Chain Hack Victims."
- 36. Kash, "New Details Released on Proposed 2016 IT Spending."
- 37. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 38. Bland, "DOES15—Mike Bland—Pain Is Over, If You Want It."
- 39. Zia, Ramírez, and Kunin, "Compliance Masonry."
- 40. Marcus Sachs, personal correspondence with Gene Kim, 2010.
- 41. "VPC Best Configuration Practices," Flux7 blog.
- 42. Galbreath, "Fraud Engineering, from Merchant Risk Council Annual Meeting 2012."
- 43. Galbreath, "DevOpsSec."
- 44. Galbreath, "DevOpsSec."
- 45. Galbreath, "DevOpsSec."
- 46. Galbreath, "DevOpsSec."
- 47. Galbreath, "DevOpsSec."
- 48. Galbreath, "DevOpsSec."
- 49. Claudius, "Attacking Cloud Services with Source Code."
- 50. Johnson, et. al., "How Fannie Mae Uses Agility to Support Homeowners and Renters."
- 51. Johnson, et. al., "How Fannie Mae Uses Agility to Support Homeowners and Renters."
- 52. Johnson, et. al., "How Fannie Mae Uses Agility to Support Homeowners and Renters."
- 53. Johnson, et. al., "How Fannie Mae Uses Agility to Support Homeowners and Renters."
- 54. Johnson, et. al., "How Fannie Mae Uses Agility to Support Homeowners and Renters."
- 55. Johnson, et. al., "How Fannie Mae Uses Agility to Support Homeowners and Renters."
- 56. Kimberly Johnson, personal correspondence with the authors, 2021.

- 1. Axelos, ITIL Service Transition, 48.
- 2. Axelos, ITIL Service Transition, 48 and 68.
- 3. Matthew and Mangot, "DOES14-Reena Mathew and Dave Mangot-Salesforce."
- 4. Mangot and Rajan, "Agile.2013.effecting.a.dev ops.transformation.at.salesforce."
- 5. Mangot and Rajan, "Agile.2013.effecting.a.dev ops.transformation.at.salesforce."
- 6. Mangot and Rajan, "Agile.2013.effecting.a.dev ops.transformation.at.salesforce."
- 7. Matthew and Mangot, "DOES14-Reena Mathew and Dave Mangot-Salesforce."
- 8. Matthew and Mangot, "DOES14—Reena Mathew and Dave Mangot—Salesforce."
- 9. Matthew and Mangot, "DOES14—Reena Mathew and Dave Mangot—Salesforce."
- 10. Matthew and Mangot, "DOES14—Reena Mathew and Dave Mangot—Salesforce."
- 11. Matthew and Mangot, "DOES14—Reena Mathew and Dave Mangot—Salesforce."
- 12. Bill Massie, personal correspondence with Gene Kim, 2014.
- 13. "Glossary," PCI Security Standards Council website.
- 14. PCI Security Standards Council, Payment Card Industry (PCI) Data Security Stands.
- 15. Bill Massie, personal correspondence with Gene Kim, 2014.
- 16. Bill Massie, personal correspondence with Gene Kim, 2014.
- 17. Bill Massie, personal correspondence with Gene Kim, 2014.
- 18. Basu, Goyal, and Hansen, "Biz & Tech Partnership Towards 10 'No Fear Releases' Per Day."
- 19. Basu, Goyal, and Hansen, "Biz & Tech Partnership Towards 10 'No Fear Releases' Per Day."

- 20. Basu, Goyal, and Hansen, "Biz & Tech Partnership Towards 10 'No Fear Releases' Per Day."
- 21. Basu, Goyal, and Hansen, "Biz & Tech Partnership Towards 10 'No Fear Releases' Per Day."
- 22. Basu, Goyal, and Hansen, "Biz & Tech Partnership Towards 10 'No Fear Releases' Per Day."
- 23. Basu, Goyal, and Hansen, "Biz & Tech Partnership Towards 10 'No Fear Releases' Per Day."
- 24. Shinn, "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations."
- 25. Shinn, "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations."
- 26. Shinn, "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations."
- Shinn, "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations."
 Shinn, "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations."
- Shinn, "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations."
- Shinn, "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations."
- Shinn, "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations."
- 32. Shinn, "DOES15—Bill Shinn—Prove it! The Last Mile for DevOps in Regulated Organizations."
- 33. DeLuccia, Gallimore, Kim, and Miller, DevOps Audit Defense Toolkit.
- 34. Mary Smith (a pseudonym), personal correspondence with Gene Kim, 2013.
- 35. Mary Smith (a pseudonym), personal correspondence with Gene Kim, 2013

Conclusion

1. Robbins, "Hacking Culture at VelocityConf."

Afterword

- 1. Forsgren et al., 2020 State of the Octoverse.
- 2. Wiseman, 2021 Work Trend Index: Annual Report.
- 3. Forsgren et al., "The SPACE of Developer Productivity."

Appendix

- 1. Ries, The Lean Startup.
- 2. Beck et al., "Twelve Principles of Agile Software."
- 3. Humble and Farley, Continuous Delivery.
- 4. Fitz, "Continuous Deployment at IMVU."
- 5. Rother, Toyota Kata, Introduction.
- 6. Rother, Toyota Kata, Introduction.
- 7. Ries, The Lean Startup.
- 8. Kim, Behr, and Spafford, The Phoenix Project, 365.
- 9. Besnard and Hollnagel, Some Myths about Industrial Safety, 3.
- 10. Besnard and Hollnagel, Some Myths about Industrial Safety, 4.
- 11. Besnard and Hollnagel, Some Myths about Industrial Safety, 6.
- 12. Besnard and Hollnagel, Some Myths about Industrial Safety, 8.
- 13. Besnard and Hollnagel, Some Myths about Industrial Safety, 9.
- 14. Besnard and Hollnagel, Some Myths about Industrial Safety, 11.
- 15. Shook, "Five Missing Pieces in Your Standardized Work (Part 3 of 3)."
- 16. "Post Event Retrospective—Part 1," Rally Blogs.

- 17. Macri, "Morgue."
- 18. Cockcroft, Hicks, and Orzell, "Lessons Netflix Learned."
- 19. Cockcroft, Hicks, and Orzell, "Lessons Netflix Learned."
- 20. Rachitsky, "7 Keys to a Successful Public Health Dashboard."

Index

Note: Figures are indicated with *f*; footnotes are indicated with *n*; tables are indicated with *t*.

SYMBOLS

%C/A percent complete and accurate), 11, 85*f*, 95

NUMBERS

2/4 outage, 318–320 18F team, 369–371

A

A/B testing conclusions on, 280 feature planning and, 278 feature testing and, 276-277 feature toggles and, 196–197 history of, 275 need for, 273 our release and, 277-278 practical guide to, 276 Accelerate: The Science of Lean and DevOps, xln, 14, 219 acceptance stage, 152f, 153 acceptance test-driven development (ATDD), 159 acceptance tests, 36, 155-156 accountability, as cultural pillar, 76 Adams, Keith, 338 Adidas, 286-287 Agile DevOps as continuation of, xxv Infrastructure Movement, 5-6, 410 Manifesto, 4-5, 410 Agile Software Development with Scrum, 122n Aisen Seiki Global, 52 Alcoa, 50, 313 alert fatigue, 247

Algra, Ingrid, 343n Allspaw, John, xxiii, 5, 6, 197, 234, 283, 284, 298, 308, 319, 360, 385n, 410 Amazon architecture transformation at, 210, 212-213 continuous delivery at, 200–201 DevOps myths and, xxiv-xxv market-oriented teams at, 102 "master of disaster" at, 316, 317, 399 post-mortem, 311n service-oriented architecture (SOA) at, 109-110 two-pizza teams at, 110, 111 Amazon Auto Scaling (AAS), 251-252 Amazon Reboot of 2014, Great, 315-316 Amazon Web Services (AWS) CloudWatch, 391 outage, 305-306, 315 security solutions architect at, 389 ambiguous threats, 313-314 American Airlines brownfield transformation, 68 case study, 15-18, 74-77 new vocabulary at, 74-77t anchoring bias, 388 Anderson, David J., 22 Andon button, 38n Andon cord description of, 37-39 Excella, 39-41f for low-risk releases, 181 work stoppage and, 416–417f anomaly detection advanced, 255-257

anomaly detection (continued) defined. 253 Kolmogorov-Smirnov test, 254, 255, 256f, 257 smoothing for, 253-254 Antani, Snehal, 354 antifragility, 52 APIs cleanly defined, 214 enablement, 112-114 loosely coupled architecture and, 209 microservice-based architecture driven bv. 388 self-service, 230 versioned, 214 application logging telemetry, 231-233 application performance monitors, 235n application-based release patterns dark launching, 190, 197-199 defined, 190 feature toggles, 190, 195-197 Arbuckle, Justin, 324, 354 Architectural Review Board (ARB), 333 architecture Amazon, 212–213 Blackboard Learn, 215–217f conclusions on, 218 downward spiral in, 208-209 eBay, 207-208 loosely coupled, 26-27, 209, 217 monoliths versus microservices, 210-212 overly tight, 26-27 service-oriented, 109, 210 strangler fig application pattern and, 70, 208, 213-217, 218 Architecture and Patterns for IT, 208 Art of Monitoring, The, 228, 230f Ashman, David, 215, 216, 217f ATM cash machines, 392 Atwood, Jeff, 170, 171, 293 Audit Defense Toolkit, DevOps, 391 auditors and compliance officers ATM cash machines and, 392 PCI compliance at Etsy, 385-387 proving compliance, 389-391 tension between IT and, 389 Austin, Jim, 245 automated testing Andon cord and, 163-165 categories of tests, 155-156 conclusions on, 166 constraints and, 26 deployment pipeline infrastructure, 151-154

essential components of, 165-166 fast and reproducible tests, 166 at Google, 148-151 green build and, 154, 163, 166 ideal testing pyramid, 157f-158 need for, 147-148 non-functional requirements and, 162-163, 328 observability and, 147 performance testing environment, 161–162 reducing reliance on manual tests, 160-161 research supporting, 165-166 running tests in parallel, 158f-159 test-driven development (TDD), 159, 161, 327 automation, DevOps and, xxvi Ayers, Zack, 39, 41f

B

bad apple theory, 307 Baker, Bill, 141 banks, as IT companies, xxxv batch sizes, reducing, 9, 22-24f, 409 Bazaarvoice, 173-175, 176 Beck, Kent, 159, 178 Beedle, Mike, 122n Behr, Kevin, xlii, 225, 233n, 415f Bell Labs, 54-55 Besnard, Denis, 416 Betz, Charles, 208 Beyond The Phoenix Project, 57 Big Fish Games, 115-117 big-bang approach, 73 bimodal IT, 71, 72 Blackboard Learn, 215-217f blameless post-mortems CSG case study, 318-320 defined, 308 inviting Ops engineers to, 117, 123-124 for organizational learning, xxxix, 48-49 publishing reports of, 311-312 sample agenda, 418-419 scheduling of, 308-310 Bland, Mike, 148, 149, 344, 345, 370 Blank, Steve, 411 Blankenship, Ed, 240 blitz, improvement, 335, 336 blue-green deployment pattern description of, 190, 191f-192 for point-of-sale system, 193 Bohmer, Richard M. J., 313 Booch, Grady, 6n Bosu, Biswanath, 388 bottlenecks

DevOps and, 405–406 generalists and, 106t handoffs, queues, and, 414 Boubez, Toufic, 249, 250f, 251f, 255, 256 bounded contexts, 109 Bouwman, Jan-Joost, 343n Brakeman, 359, 362f-363 branches, feature, 170 branching by abstraction, 214n branching strategies, 167, 170-171 Brittain, Mike, 241f, 262f Brook, Frederick, xli Brooke's Law, 286 brownfield projects case study, 69-71 defined, 67 greenfield projects versus, 66-69 technical debt and, 67 Building Blocks, Blackboard Learn, 215-217f Building the Future: Big Teaming for Audacious Innovation, 312 bureaucratic organizations, 42, 47, 48t bureaucratic processes, cutting, 42-43, 296-297, 299 Burgess, Mark, 6n burnout, decreased, xl, 15, 175 business relationship manager, 116 Buytaert, Kris, 343n

С

Cagan, Marty, 90 Campbell-Pretty, Em, 135-136 Canahuati, Pedro, 105, 263 canary release pattern, 190, 194f-195 canary release test, 177n Capital One biz and tech partnership at, 387-389 case study, 342-343, 387-389 Got Goo? program, 296 internal conferences, 342–343 cardholder data breaches, 364, 371 cardholder data environment (CDE), 385-386 case studies Adidas, 286-287 Amazon, 212-213 American Airlines, 15–18, 74–77 ATM cash machines, 392 Bazaarvoice, 173-175, 176 Bell Labs, 54-55 Blackboard Learn, 215–217f Capital One, 342-343, 387-389 CSG International, 181-183, 201-206, 318-320

Dixons Retail, 193 Etsy, 186-188, 332, 373-375, 385-387 Excella, 39-41f Facebook, 198-199 Fannie Mae, 376-378 Federal Government agencies, 369-371 Google, 269-271, 290-291 hospital system, 29-32 hotel company, 143-144 Kessel Run refueling system, 66-69 LinkedIn, 91-93, 237-238 Nationwide Building Society, 124-127 Nationwide Insurance, 342 Netflix, 251-253 Pivotal Labs, 293-294 Salesforce.com, 383-384 Target, 112-114, 333-334, 342, 343 tax collection agency for UK, 77-80 Twitter, 360-363 Yahoo! Answers, 278–280 Chacon, Scott, 281, 282f Chakrabarti, Arup, 263 Change, John Shook's Model of, 205 change advisory board (CAB), 380, 382 change approval processes case studies, 383-384, 385-387, 387-389 dangers of, 283-284 at Etsy, 385-387 normal changes, 380, 381-382 at Salesforce.com, 383-384 security and compliance in, 379-380 standard changes, 379-381, 383-384 three categories of changes, 379-380 urgent changes, 380 change control failure, 283 change freezes, 292 Chaos Gorilla, 420 Chaos Kong, 420 Chaos Monkey, 52, 55, 306-307, 315 Chapman, Brent, 347 Chapman, Janet, 124, 125, 126f chat rooms announcing changes with, 288 drawbacks of, 94-95 Hubot at GitHub, 321-323 organizational knowledge and, 321-322 shared goals and, 94 as watercooler, 322 Chuvakin, Anton A., 232 Clanton, Ross, 74, 75f, 76, 77t, 297, 335, 336.343 Claudius, Jonathan, 375 Clemm, Josh, 91-93

cloud computing, five characteristics of, 330-331 cloud native, 306 Cloud System Administration, The Practice of, 325 Cloud.gov, 370, 371 cluster immune system, 190, 195n coaching kata, 53 Cockcroft, Adrian, xxxit, 102n, 231, 296 code infrastructure as, 6n libraries, 356 maintainability, 326 repositories, 355-357 signing, 359-360 Code Climate, 359 code commits automated tests on, 160, 166 daily, 172 gated commits, 172 Google, 150, 290 Pivotal Labs, 294 security and, 357 strangler fig application pattern, 215-217f code reviews. See also change approval processes ATM systems and, 392 change reviews versus, 289n defined. 288 e-mail pass around, 290 forms of, 289-290 Google, 290-291 guidelines for, 288-289 learning-based culture and, 339 "over the shoulder," 290 pair programming, 289, 292-294 Pivotal Labs, 293-294 requesting, 282 separation of duty and, 184, 384-385, 386 size of change and, 289 tool-assisted, 290 unauthorized access and, 375, 376 Codecov security breach, 368-369 Cohen, Joshua, 39, 41f Collins, Justin, 360, 361 Columbia space shuttle, 313 commit stage, 152f, 153 Common Vulnerabilities and Exposures (CVE), 365 communities of practice, 343-345 compliance officers and auditors ATM cash machines and, 392 Payment Card Industry Data Security Standards (PCI DSS), 385

PCI compliance at Etsy, 385-387 proving compliance, 389-391 separation of duty and, 379, 384-387 tension between IT and, 389 complicated-subsystem teams, 112 conferences internal, 342–343 sharing experiences from DevOps, 341-342 Conformity Monkey, 420 Conrad, Ben, 78 constraint identification, 25-27 Constraints, Theory of, 4, 412-413 Consul. 242n containers, 143-144, 152, 153n contextual inquiry, 264 continual learning and experimentation, 3, 45-56 continuous delivery. See also deployment process; low-risk releases, enabling continuous deployment versus, 199–201 defined, 133, 200 elite performance and, 201 infrastructure monitoring and, 243 low-risk releases and, 199-201 Continuous Delivery: Reliable Software Releases Through Build, Test, Deployment Automation, xxiv, 151, 193, 199, 219 Continuous Delivery Movement, 6, 410-411 continuous integration (CI) and testing defined, 36, 151n deployment pipeline infrastructure and, 151 - 154continuous integration practices Andon cord and, 163–165 at Bazaarvoice, 173-175, 176 catching errors early, 157-163 reliable automated validation test suite, 154-156 three capabilities required for, 154 trunk-based development, 167–176 convergence of DevOps, 409-412 Conway, Melvin, 97 Conway's Law conclusions about, 114 defined, 61, 97-98 at Etsy, 98-100, 108 organizational archetypes and, 100-101 Target case study and, 112–114 team boundaries in accordance with. 108 two-pizza team and, 110–111 Cook, Richard, 58, 319 Cook, Scott, 274

core conflict in IT operations, xxxii-xxxiii, 412f-413 Corman, Josh, 353, 363, 368, 412 Cornago, Fernando, 286 costs of IT-related deficiencies, xxxvi-xxxvii COTS software, 417-418 counterfactual thinking, 283n, 310 COVID-19 pandemic call centers during, 126–127 remote work and, 108n UK's financial support package, 77-80 Cox, Jason, 107, 119-120, 296 crowdsourcing technology governance, 333-334 CSG International blameless post-mortem, 318-320 brownfield transformation, 68 case study, 181-183, 201-206, 318-320 daily deployments at, 181-183 generalists, 106–107 organizational transformation, 201-206 culture, organizational high-trust, xxxix, 45, 48, 150 importance of, 46 just culture, 47, 307-308 learning-based, 48-49, 339 safety culture, 46-49 three pillars of, 75-76 three types of, 47, 48t culture of causality, 225, 233n Cundiff, Dan, 333 Cunningham, Ward, xxxii, 171 customer acquisition funnels, 240, 273, 275, 278 customers, two types of, 43

D

daily work, improvement of, 49-50 daily work of development at Big Fish Games, 115–117 conclusions on, 127-128 embedded Ops engineers in, 116, 119-120 Ops liaisons in, 116, 117, 120-121 shared services in, 117-118 team rituals in, 117, 121-124 dark launch defined, 190, 197-198 Facebook Chat, 198–199 dashboard Adidas, 287 creating a simple, 237n daily work and, 234 Etsy, 227 public health, 421

Data Breach Investigation Report (DBIR), 364 database changes, dealing with, 192 database syntax error, 374 Davenport, Shawn, 295*n* Debois, Patrick, xxiii, 5, 6, 405–406, 410 DEBUG level, 232 dedicated release engineer, 116 defects, as waste, 28 DeGrandis, Dominica, 22, 58 Dekker, Sidney, 34, 47, 58, 307, 347, 395 (Delicate) Art of Bureaucracy, The, 299 Deming, W. Edwards, 38, 54 demonstrations, compliance by, 354 dependency scanning, 359 deploy code button, 186 Deployinator console, Etsy, 187–188f deployment lead time, 8–11f, 409 deployment pipeline foundations Andon cord and, 163-165 containers, 143-144, 152, 153n defined, 6, 151, 152f definition of "done," 144-145 Enterprise Data Warehouse, 135-137 goal of deployment pipeline, 153 infrastructure, 151-154 on-demand creation of test environments, 137-138 rebuilding instead of repairing infrastructure. 141-143 single repository of truth, 139–141, 150, 324-327 deployment pipeline protection ATMs and production telemetry, 392 auditors and, 389-391 Capital One, 387-389 change advisory board (CAB) and, 380, 381, 382 change approval processes and, 379-380 Etsy, 385-387 for normal changes, 380, 381-382 separation of duty, 379, 384-387 for standard changes, 379-381, 383-384 deployment process Andon cord and, 181 automating, 179-181 continuous deployment, 199-201 CSG International, 181-183, 201-206 decouple deployments from releases, 189-199 Dixons Retail, 193 Etsy, 186-188 Facebook, 177-179 release versus, 189

deployment process (continued) self-service deployments, 184-188 smoke testing deployments, 180, 187 deploys per day per developer, xxxviii, xxxxixf-xl destructive testing, 383-384 Dev team rituals, 121-124 Dev tests, 36 developer productivity DevOps practices and, xlif measuring, 401-403 shared services for, 117-119 development, daily work of at Big Fish Games, 115-117 conclusions on, 127-128 embedded Ops engineers in, 116, 119-120 Ops liaisons in, 116, 117, 120-121 shared services in, 117-118 team rituals in, 117, 121–124 development, hypothesis-driven at Intuit, 273–275 need for, 273, 280 development, test-driven (TDD). See also automated testing building incrementally with, 161 defined. 159 shared libraries and, 327 study on, 159n development, trunk-based adopting practices of, 172 at Bazaarvoice, 173-175, 176 benefits of, 175 conclusions on, 175-176 gated commits and, 172 for HP's LaserJet Firmware division. 168 - 170need for, 167, 171 DevOps breaking downward spiral with, xxxv-xxxvii business value of, xxxvii-xxxix convergence of, 409-412 core conflict in IT operations and, xxxii-xxxiii, 412f-413 developer productivity and, xlif-xlii downward spiral in IT and, xxxiii-xxxvii, 413t-414t history of, 3-6 myths about, xxiv-xxv outcomes created by, xxviii, xxix principles underpinning, 12-14 DevOps Days. See DevOpsDays DevOps Enterprise Summit, xxxix, 341-342, 343 DevOps journeys. See case studies

DevOpsDays, xxiii, xxiv, 6, 341, 343n, 353, 355.410 Dickerson, Chad, 99 Dignan, Larry, 110, 111 direct response marketing, 275 Disaster Recovery Program (DiRT), Google's, 317 Disney, 107, 119-120, 296 Dixons Retail, 193 Doctor Monkey, 420 Dojo, DevOps, 335-336 Dojos, Getting Started With, 58 "done," modifying definition of, 144-145, 172 DORA, State of DevOps Reports from, 5, 14, 57, 103, 140, 165, 166, 175, 185, 201, 217, 227, 243, 285, 312, 326, 331, 343 downward spiral in IT description of, xxxiii-xxxvii, 413t-414t DevOps for breaking, xxxvii-xxxix Drucker, Peter, 75, 80 Dunbar, Robin, 111n Dunbar's number, 111*n* Dweck, Carol, 107 dynamic analysis, 354

E

early adopters defined, 72f, 73 finding, 73-74 eBay, 90, 207-208 economic costs of IT-related deficiencies, xxxvi-xxxviii Edmondson, Amy C., 312, 313, 347 Edwards, Damon, 10f, 28, 118 Eli Lilly, 312 Eltridge, Patrick, 124, 125, 126f, 127 email pass around code review, 290 emergency change advisory board (ECAB), 380 employee burnout, lower rates of, xl, 175 enabling teams, 112 Encasement Strategy, 70 Eno, Brian, 54, 55 Enterprise Data Warehouse, 135-137 envelope game, simulation of, 23, 24*f* environment-based release patterns blue-green deployments, 190, 191f-192, 193 canary releases, 190, 194f-195 cluster immune systems, 190, 195 defined. 190 ERROR level, 232 ethics of DevOps, xxxvii-xxxix Etsy blameless post-mortems, 310, 311, 419

brownfield transformation, 68-69 case studies, 186-188, 332, 373-375, 385-387 cluster immune system, 195n continuous delivery at, 201 Conway's Law and, 98-100, 108 designated Ops liaison at, 120-121 DevOps myths and, xxiv-xxv DevOps transformation at, 226–227 experimentation, 277-278 functional orientation, 101n, 104 LAMP stack, 226 learning-based culture, 308 Morgue tool, 311-312 organizational learning, 49 PCI compliance at, 385-387 PHP run-time warning, 262f retrospective meetings, 310, 311, 419 self-service deployment, 186-188 separation of duty, 385–387 standardizing technology stack at, 332 StatsD metric library, 234, 235f transformation projects, 63 Evans, Eric J., 109 Evans, Jason, 338 event router, 229, 230f Excella, Andon cord at, 39-41f Expanding Pockets of Greatness: Spreading DevOps Horizontally in Your Organization, 129 experimentation, rapid A/B testing, 273, 275-278 customer acquisition funnel and, 240, 273, 275, 278 at Etsy, 277-278 at Intuit, 273-275 need for, 273 at TurboTax, 274 at Yahoo! Answers, 278-280 exploratory testing, 36 Explore It !: Reduce Risk and Increase Confidence with Exploratory Testing, 36, 219 extra features, 28 extra processes, 28

F

Facebook canary release pattern, 194*f*-195 case study, 198–199 code deployment, 177–179 continuous delivery, 201 experimentation, 279 feedback, 105 Gatekeeper, 196*n*, 199 hackathon, 338 Facebook Chat, dark launch, 198-199 fail fasts. 316n failure parties, 312 failures blameless post-mortem at CSG, 318–320 calculated risk-taking and, 314-315 game days to rehearse, 316–318 no fear of. 55 publishing post-mortem reports, 311-312 redefining, 314-315 rehearsing and injecting, 315-316 retrospective meetings after, 308–310 weak failure signals, 313-314 fallbacks, 316n Fannie Mae, 376-378 Farley, David, 6, 151, 152f, 156, 158f, 193, 199, 214, 410 Farley, Steve, 339, 342 Farr, Will, 295n Farrall, Paul, 115-116 fast and reproducible tests, 166 FATAL level, 232 Fearless Organization, The, 347 feature branches, 170 feature flags. See feature toggles feature freezes, 175 feature toggles, 195-197, 277 features extra, 28 planning, 278 testing, 276-277, 279 user excitement and, 241f Federal Government agencies, 369–371 feedback. See also The Second Way (principles of feedback) Andon cord and, 37-41 customer observation, 264–265 cycle times, 37f fast and constant, 10, 13-14, 33 optimizing for downstream work centers, 43-44 quality control closer to the source, 42-43 pager rotation duties, 263-264 principles of, 3, 13-14, 33 production telemetry and, 261-262 safe deployment of code and, 259–261 safety within complex systems, 33-34 seeing problems as they occur, 35-36, 244 self-management by developers, 265-271 stakeholder, 36 swarming, 37-41 types and cycle times, 36–37f user, 36, 37f Fernandez, Roberto, 73, 100

Fifth Discipline, The, 35, 49 First Way, The. See The First Way (principles of flow) Fitz, Tim, 6, 192n, 199, 411 Five Dysfunctions of a Team: A Leadership Fable, 347 fix forward, 262 fixed mindset, 107 fixits, 345 Flickr, 197-198, 278, 360 flow, principles of, 3. See also The First Way (principles of flow) flow metrics, 11-12 focusing steps, five, 26, 32 following work downstream, 264-265 Forsgren, Nicole, xxxix-xl, 5, 14, 140, 228f, 363, 401-403, 404 Fowler, Martin, 156, 157, 213, 214, 220, 306n fraud, defined, 373 Fryman, James, 295n full-stack engineer, 106, 386 functional-oriented organizations defined, 100 DevOps outcomes in, 103-104 market orientation versus, 100-101, 103f problems of, 101-102 funding services and products, 107–108 Furtado, Adam, 69, 70

G

Gaekwad, Karthik, 339 Galbreath, Nick, 259-261, 355, 373-375 Gall's Law, 70 Gallimore, Jeff, 41 game days, 52, 316-318 Ganglia, 226, 229 gated commits, 172 Gatekeeper, Facebook's, 196n, 199 Gauntlt security tool, 353, 357 Gaussian distribution, 247f, 249, 253 GE Capital, 324, 354 Geer, Dan, 368 Geinert, Levi, 333 General Electric, CEO of, xxxi-xxxii General Motors manufacturing plant, 35, 38.45 generalists, 105-107 generative organizations, 47-48t, 57 Gertner, Jon, 54, 55 GitHub functional orientation, 101n, 104 Hubot, 321-323 Octoverse Report, 57, 368, 401

peer review, 281-283 pull request processes, 295-296 vulnerability timeline, 368 GitHub Flow, 282 The Goal: A Process of Ongoing Improvement, xlii, xliii, 30, 406 goals global, 19, 21 improvement, 88 Goldratt, Eliyahu M., xxxiii, xlii, 25, 26, 32, 406 Goldratt's Theory of Constraint, 25-26, 32 Google architecture, 209f-210 automated testing, 148-151 case study, 269-271, 290-291 code reviews, 290-291 continuous delivery, 200-201 DevOps myths and, xxiv-xxv disaster recovery program, 317-318 grouplets, 344, 345 launch and hand-off readiness review, 269-271 retrospective documents, 311 service-oriented architectures (SOA), 109-110 shared source code repository, 300, 324-326 Testing on the Toilet newsletter, 149n, 344 Web Server team, 148-151 Google Cloud Datastore, 209f, 210 Got Goo? program, 296 Gothelf, Jeff, 411 Govindarajan, Vijay, 86, 87 Goyal, Rakesh, 387, 388 Grafana, 79, 234, 254, 255 Gramm-Leach-Bailey Act, 389 Graphite, 226, 234, 235f, 236, 254, 255, 322, 374f Gray, Jim, 212 green build, 154, 163, 166 greenfield vs. brownfield services, 66-69 grouplets, 344, 345 growth mindset, 107 Gruver, Gary, 43, 148, 160, 168, 169, 170 guardrails, 79 Gupta, Prachi, 237, 238

Н

Haber, Eben, 5 hackathon defined, 337*n* Facebook, 338 Hamilton, James, 306*n* Hammant, Paul, 214*n*

Hammond, Paul, xxiii, 5, 6, 360, 410 hand back mechanism, 268-269 hand-off readiness review (HRR), 269, 270, 271f handoffs, 20, 24-25, 414-415 hardships and waste, 27-29 healthcare organizations generative cultures and, 47-48t HIPAA requirements, 390-391 hospital case study, 29–32 helplessness, learned, xxxvi Hendrickson, Elizabeth, 36, 160, 219, 293-294,300 heroics, 10, 28-29, 170 High-Velocity Edge, The, xxxv HIPAA, 390-391 HipHop virtual machine project (HHVM), 338 history A/B testing, 275 DevOps, 3-6 software delivery, xxx-xxxit HMRC tax collection agency, 77-80 Hodge, Victoria J., 245 holdouts, identifying, 74 Hollnagel, Erik, 416 Holmes, Dwayne, 143, 144 hospital system case study, 29-32 HP LaserJet, 69 HP's LaserJet Firmware division, 168-170, 176 HSBC bank, xxxvn Hubot, at GitHub, 321-323 Humble, Jez, xxii-xxiii, xl, 6, 151, 152f, 156, 158f, 191f, 199, 200, 207, 219, 273, 277, 284, 404-405, 406, 410 Hyatt, Matt, 78, 79, 80 hybrid schedules, 108n hypothesis-driven development at Intuit, 273-275 need for, 273, 280

I

Idea Factory: Bell Labs and the Great Age of American Innovation, The, 54 ideal testing pyramid, 157f–158 Imbriaco, Mark, 322 Immelt, Jeffrey, xxxi–xxxii immersive learning opportunities, 16, 18 immutable infrastructure, 142 immutable services, 214 imposter syndrome, 148*n*, 310 improvement blitz, 335, 336 improvement goals, 88 improvement kata, 6 improvement of daily work, 49-50 INFO level, 232 information radiators, 236, 237, 241 information security. See also deployment pipeline protection 18F team, 369-371 bad paths, 358 Brakeman, 359, 362f-363 change approval processes, 379-380 code signing, 359–360 data breaches, 364, 368-369, 371 defect tracking, 355 dependency scanning, 359 deployment pipeline and, 357, 375-376 DevOps and, 353 dynamic analysis, 354 early-stage product demonstrations, 354 Etsy, 373-375, 385-387 Fannie Mae, 376-378 Gauntlt security tool, 353, 357 happy path, 358 **Open Web Application Security Project** (OWASP), 359n, 360 OWASP Dependency Check, 368n OWASP ZAP, 358f, 359 Payment Card Industry Data Security Standards (PCI DSS), 385 post-mortems and, 355 preventive security controls, 355–357 production telemetry and, 371-373 Rugged DevOps, 353 sad and bad paths, 358 separation of duty, 379, 384-387 shared source code repositories and, 355-357 shifting security left, 376-378 silo, 353 software supply chain and, 363-369 source code integrity and code signing, 359-360 SQL injection attacks, 373, 374f static analysis, 358f-359 Twitter case study, 360-363 Infosec. See information security Infosec team, 83 infrastructure centralized telemetry, 227-231 changes, 383-384 as code. 6n deployment pipeline, 151-154 immutable, 142 metrics, 242-243 rebuilding instead of repairing, 141–143

ING technology organization, 74
innovators and early adopters defined, 72f-73 finding, 73-74
integration tests, 156
Intuit, 273-275
IT operations core conflict in, xxxii-xxxiii, 412f-413 developer productivity, xlif-xlii
DevOps and, xxvi downward spiral in, xxxv-xxxix, 413t-414t impact of DevOps on, xxxvii-xxxix
ITIL, xxvii, 139, 263n, 285, 286, 379, 380n
ITIL CMBD, 242n

J

Jacob, Adam, 6n Jacobson, Daniel, 252f, 254f Janitor Monkey, 420 Java automation, 161 Bazaarvoice, 173 dependency scanning, 359 EAR and WAR files, 152 LinkedIn, 91 Maven Central, 364 ORM, 99n JavaScript application logging, 231n client software level, 239 CSG, 318 eBay, 207n Facebook, 199 libraries, 326 NPM, 364 open-source dependencies, 363 StatsD, 234 Jenkins, 152, 153, 180, 187, 205, 322, 358f, 377 JIRA, 355, 377, 381, 382, 386 Johnson, Kimberly, 376, 377-378 Jones, Angie, 161 Jones, Daniel T., 23 Joshi, Neeraj, 252f, 254f just culture, 47, 307-308 Just Culture, 347

K

kaizen blitz, 49, 335, 336 Kalantzis, Christos, 315–316 *Kanban: Successful Evolutionary Change for Your Technology Business*, 22 kanban boards

example, 20f-21 Ops work on, 124 shared goals and, 94 Toyota Production System, 4 Kandogan, Eser, 5 Kanies, Luke, xxiii, 6n Kastner, Erik, 187, 188f Kelly, Mervin, 54, 55 Kersten, Mik, 12, 54 Kessel Run mid-air refueling system, 69–71 Kim, Gene, xxi-xxii, xl, xlii, 13f, 47n, 54, 57, 58, 225, 233n, 265, 284, 295n, 300, 353n, 364, 385n, 403-404, 406, 415f Kissler, Courtney, 63, 64, 65, 66, 81-82, 83 Knight Capital failure, 283 Kohavi, Ron, 276, 277 Kolmogorov-Smirnov test, 254, 255, 256f, 257 Krishnan, Kripa, 317-318 Kumar, Ashish, 291f

L

laggards (skeptics), 72f, 73 LAMP stack DevOps myth and, xxvi Etsy, 226 large batch sizes merges of, 170-171 small versus, 9, 22-24f, 409 Latency Monkey, 420 latent defects, 317 Lauderbach, John, 108n launch guidance, 266-267 launch readiness review (LRR), 269, 270, 271f Lead Architecture Review Board (LARB), 296, 297 lead time defined, 409 focus on, 8, 18 Lean Movement and, 409 of minutes, 10–11f process time versus, 9f queue size and, 22, 415f of three months, 10f leaders role of, 52-54 vocabulary for, 76-77t Lean Enterprise: How High Performance Organizations Innovate at Scale, 278 Lean Manufacturing, xxx, 7, 8, 9 Lean Movement description of, 3, 409 missing element in, 6 Lean Startup, The, 411

Lean UX movement, 411 LeanKit, 381 learned helplessness, xxxvi learning-based culture ASREDS learning loop, 340, 341f communities of practice, 343-345 conclusions on, 346 DevOps conferences, 341-342 Etsy, 49 grouplets, 344, 345 importance of, 46 improvement blitz, 335, 336 internal conferences, 342-343 just culture, 47, 307-308 rituals to pay down technical debt, 336-339 Teaching Thursday, 339 thirty-day challenge at Target, 335-336 safety culture, 46-49 trust and, xli, 45, 48, 150 Leibman, Maya, 15, 16, 17, 75f, 77t Lencioni, Patrick, 347 Lesiecki, Nick, 344 Letuchy, Eugene, 198, 199 Levenberg, Josh, 149, 300 liaisons, Ops assigning, 117, 120-121 purpose of, 115-117 team rituals and, 121-124 two types of, 116 libraries, shared, 326, 327 Lightbody, Patrick, 263 limiting work in progress (WIP), 7, 21-22 Limoncelli, Tom, 248, 270, 271, 325-326 LinkedIn case study, 91-93, 237-238 former monolithic architecture, 210 Operation Inversion, 91-93 self-service metrics, 237-238 Little, Christopher, xxxv, xxvi, 94 logging infrastructure, 231n logging telemetry, 231-233 logs centralized, 229 in monitoring framework, 230f-231 Lonestar Application Security Conference, 353 Long, Jeremy, 365 long-lived private branches, 170 loosely coupled architecture, 26-27, 209, 217 Love, Paul, 353n low-risk releases, architecture for at Amazon, 212-213

at Blackboard Learn, 215-217f conclusion on, 218 downward spiral and, 208-209 at eBay, 207-208 loosely coupled architecture, 26-27, 209, 217 monoliths versus microservices, 210-212 service-oriented architecture, 109, 210 strangler fig application pattern and, 70, 208, 213-217, 218 low-risk releases, enabling Andon cord and, 181 application-based release patterns, 190, 195-199 automating deployment process, 179-181 conclusion on, 206 continuous delivery and, 199-201 at CSG International, 181-183, 201-206 dark launching, 197-199 environment-based release patterns, 190, 191-195 at Etsy, 186-188, 201 at Facebook, 177-179, 201 feature toggles, 195-197 self-service deployments, 184-188 Luyten, Stefan, 24f

M

Macri, Bethany, 49, 419 Magill, Stephen, 364 Maglio, Paul, 5 Making Work Visible: Exposing Time Theft to Optimize Work & Flow, 22, 58 Malpass, Ian, 227, 235f, 310 Mangot, Dave, 383, 384 manual tests automating, 160-161 high reliance on, 10 manufacturing value stream, 7 market-oriented organizations, 100-101 market-oriented teams, 102-103 Marsh, Dianne, 118 Martens, Ryan, 94 Martin, Karen, 7, 9n, 11 mass marketing, 275 Massie, Bill, 385-387 Matatall, Neil, 360, 361 Mathew, Reena, 383, 384 matrix-oriented organizations, 100 Maurer, Dan, 274 Maven Central repository, 364, 365 Maximilien, E. Michael, 159n McChrystal, Stanley, 347

McDonnell, Patrick, 226 McKinley, Dan, 332 mean time to recover. See MTTR means and standard deviations, 246–248 Measuring Software Quality whitepaper, 58 Mediratta, Bharat, 149, 344 mentors, test certified, 345 Messeri, Eran, 150, 290, 325 Metasploit, 359, 369 metrics actionable business, 240-241f application and business, 240-242 flow, 11-12 infrastructure, 242-243 library (StatsD), 234 MTTR and, 227 production, 234-235 self-service, 236-238 Metz, Cade, 338 Mickman, Heather, 112-114, 297, 343 microservices monoliths versus, 210–212 pros and cons of, 211t shifting to, 212-213 Microsoft Excel, 255 Microsoft Office Framework (MOF) Study, 225 Milstein, Dan, 310 Minimum viable product (MVP), 388 mistakes. See also failures examining, 308-310 in learning-based culture, 307-308 redefining failure, 314-315 Model of Change, John Shook's, 205 monitoring frameworks, 227-231 monolithic architectures defined, 210, 211t shifting from, 212-213 Moore, Geoffrey A., 72 Morgue tool, Etsy's, 311-312 Morrison, Erica, 183f, 204, 206, 318 motion waste, 28 MTTR batch size and, 292 culture of causality and, 233n daily deployments and, 183f defined, 16 fact-based problem-solving and, 234 high performers and, 186f, 227, 228f low-risk changes and, 381 Morgue tool for recording, 311 optimizing for, 261n tracking metrics and, 227

Mueller, Ernest, 117*n*, 122, 173–175, 237 Mulkey, Jody, 104–105 Multichannel Digital Tax Platform (MDTP), 78–80 multitasking, 21–22 multivariate testing, defined, 276 MySQL, xxvi, xxxit, 226, 296, 332 *Mythical Man-Month, The*, xli myths DevOps, xxiv–xxvi industrial safety, 416*t*

N

Nagappan, Nachi, 159n NASA, 231, 313, 314 National Institute of Standards and Technology (NIST), 330, 331, 389 National Vulnerability Database, 368 Nationwide Building Society, 124-127 Nationwide Insurance, 339, 342 Naval Reactors (NR), 51 .NET. xxviii. 363 Netflix Archaius library, 196n auto-scaling capacity, 251-253 case study, 251-253 Chaos Monkey, 52, 55, 306-307, 315, 420 cluster immune system, 195n DevOps myths and, xxiv-xxv as digital-native company, 15 market-oriented teams, 102 shared services, 118 Simian Army, 369n, 420 telemetry, 245-246 Newland, Jesse, 321, 323 Nmap, 359, 369 non-functional requirements (NFRs), 89, 90f, 162-163, 328 Nordstrom, 15, 63-66 normal changes, 380, 381-382 North, Dan, 191f, 193, 232 Nygard, Michael, 315, 347

0

object relational mapping (ORM), 99*n* observability, and testing, 147 *Octoverse Report*, 57, 368, 401 O'Donnell, Glenn, 340 O'Neill, Paul, 50, 313 Open Web Application Security Project (OWASP), 359*n*, 360 Operation Desert Shield, 189*n* opinion of the team, 40

opinionated platform, 79 Ops engineers on service teams, embedding, 119-120 Ops liaisons assigning, 117, 120-121 purpose of, 115-117 team rituals and, 121-124 two types of, 116 O'Reilly, Barry, 278 organizational archetypes, 100-101 organizational culture importance of, 46 just culture, 47, 307-308 safety culture, 46-49 three pillars of, 75-76 three types of, 47, 48t trust and, xli, 45, 48, 150 organizational goals, and technology choices, 329-330 organizational knowledge chat rooms for capturing, 321-323 easy-to-reuse, 323-324 Ops user stories and, 328-329 recommend_tech program at Target, 333-334 source code repository for, 139-141, 150, 324-327 organizational learning ASREDS learning loop, 340, 341f communities of practice, 343-345 conclusions on, 346 DevOps conferences, 341-342 Etsy, 49 grouplets, 344, 345 improvement blitz, 335, 336 internal conferences, 342-343 rituals to pay down technical debt, 336–339 Teaching Thursday, 339 thirty-day challenge at Target, 335-336 ORM (object relational mapping), 99n Orzen, Mike, 49 Osterling, Mike, 7, 9n, 11 Otto, Andreia, 287 outages 2/4 outage, 318-320 Adidas, 286 Amazon Web Services (AWS), 305-306, 315 culture of blame around, 233 culture of causality and, 225 mastering, 347 Netflix, 314-315 outlier detection, 245-246

outsourcing contracts, IT, 102*n* "over the shoulder" code review, 290 OWASP (Open Web Application Security Project), 359*n*, 360 OWASP Dependency Check, 359, 365, 368*n* OWASP ZAP, 358*f*, 359 Özil, Giray, 289

P

pager rotation duties, 263-264 pair programmed rollback, 164n pair programming, 289, 292-294 pairing hours, 293n Pais, Manuel, 111, 117, 129 Pal, Tapabrata, 296, 342-343, 353n Pandey, Ravi, 336 Parker, Doug, 76-77 Paroski, Drew, 338 partially done work, 28 passion, as cultural piller, 75 passwords, OWASP guidance on, 360 pathological organizations, 47, 48t Payment Card Industry Data Security Standards (PCI DSS), 385, 386 PayPal, 73n PDCA (Plan, Do, Check, Act) cycle, 38.54 peer review of changes, 288-290 code review basics, 288-291 email pass around, 290 GitHub, 281-283 "over the shoulder" code review, 290 pair programming, 289, 292-294 pull requests, 281-283, 295-296 tool-assisted code review, 290 performance testing environment, 161–162 Perrow, Charles, 34 Phoenix Project, The, xxii, xlii, xxxix, 12, 404, 406, 413, 414 PHP Conway's Law, 98, 99 Etsy, 226, 262f, 332 Facebook, 177n, 178, 199, 338 LAMP stack and DevOps, xxvi as representative technology, xxxit Pinterest, 114 Pivotal Labs, 293–294 planning horizons, short, 88-89 platform teams, 112 Poppendieck, Mary and Tom, 27 Porter, Chris, 376, 377

post-mortems, blameless CSG case study, 318-320 defined, 308 inviting Ops engineers to, 117, 123-124 for organizational learning, xxxix, 48-49 publishing reports of, 311-312 sample agenda, 418-419 scheduling of, 308-310 Potvin, Rachel, 149, 300, 324, 326 powerlessness, employees and, xxxvi pretotyping, 275n preventive security controls, 355-357 problem-solving guided by telemetry, 225, 233-234 seeing problems as they occur, 35-36, 244 swarming, 37-41 process time, lead time versus, 9f product owner, 83 production telemetry. See also telemetry; telemetry analysis ATM systems and, 392 contextual inquiry, 264 in daily work, 234-235 feature toggles, 195-197, 262, 277 fix forward, 262 hand-off readiness review (HRR), 269, 270.271f information security and, 371–373 launch guidance, 266-267 launch readiness review (LLR), 270, 271f LinkedIn, 237-238 pager rotation duties, 263-264 roll back, 262 service hand-back mechanism, 268f-269 site reliability engineers (SREs), 269-271 UX observation, 264-265 productivity, developer DevOps practices for, xlif measuring, 401-403 shared services for, 117-120 Project to Product, 12 prototypes, creating, 275n Prugh, Scott, 72, 106-107, 181, 182, 183, 201-204,231 psychological safety, 347 pull requests defined, 281 evaluating, 295-296 GitHub, 281–283 Puppet Labs, xxiii, xxxix, xlif, 5, 14, 140, 185, 217 Python

Etsy, 332 ORM, 99*n* PyPi for, 364 Zenoss and, 238

Q

quality controls, ineffective, 42–43 queue size, controlling, 22, 414–415*f* Quicken, 274*n* Quora, 278, 306

R

Rachitsky, Lenny, 420 Rajan, Karthik, 383 Rally, 94, 253, 381 Rapoport, Roy, 245, 246, 314 Rational Unified Process, 4 Raymond, Eric S., 97 rebooting servers, 225 Red Hat, xxxit, 138 Reddit, 306 Reddy, Tarum, 253 release, deployment versus, 189 Release It! Design and Deploy Production-Ready Software, 315, 347 release managers, 83 release patterns application-based, 190, 195-199 environment-based, 190, 191-195 releases, architecture for low-risk. See architecture releases, enabling low-risk Andon cord and, 181 application-based release patterns, 190, 195-199 automating deployment process, 179-181 conclusion on, 206 continuous delivery and, 199-201 at CSG International, 181–183, 201-206 dark launching, 197-199 environment-based release patterns, 190, 191-195 at Etsy, 186-188, 201 at Facebook, 177-179, 201 feature toggles, 195-197 self-service deployments, 184-188 Rembetsy, Michael, 63, 226, 332 remote work, Covid-19 pandemic and, 108n request for change (RFC) form, 380, 382 resilience engineering, defined, 316 resilience patterns, 51-52

resilient organizations calculated risk-taking in, 314-315 CSG case study, 318-320 description of, 305 experimental model for, 313 game days in, 316-318 just, learning culture in, 307-308, 320 Netflix example, 305-307 post-mortem reports in, 311-312 retrospective meetings at, 308-310 retrospectives CSG case study, 318-320 defined. 308 Ops engineers at, 117, 123-124 for organizational learning, xxxix, 48–49 publishing reports of, 311–312 sample agenda, 418-419 scheduling of, 308-310 Rettif, Lucas, 333 review and coordination processes Adidas, 286-287 bureaucracy and, 42-43, 296-297 code review basics, 288-291 conclusions on, 297-298 coordination and scheduling, 288 dangers of change approval processes, 283-284 dangers of manual testing and change freezes, 292 dangers of overly controlling changes, 284-285 GitHub, 281-283 Google, 290-291 pair programming, 289, 292–294 Pivotal Labs, 293-294 pull requests, 281-283, 295-296 Rhoades, Lacy, 277–278 Rice, David, 412 Richardson, Vernon, xxxvin Ries, Eric, 6n, 24, 195n, 411 Right Media, 259-261 risk-taking, calculated, 314-315 rituals Dev team, 121-124 technical debt and, 336-339 Robbins, Jesse, 316, 317, 399, 410 Roberto, Michael, 313 roll back, 262 Rossi, Chuck, 177, 178, 198 Rother, Mike, 6, 49, 53, 54, 104, 411 Rouster, 383 Ruby, 152, 231n, 234, 359, 363, 364 Ruby on Rails, xxxit, 99n, 143, 191n, 362

Rugged Computing movement, 412 Rugged DevOps, 353 Rushgrove, Gareth, 358f

S

Sachs, Marcus, 371 safety in complex systems, 33-34 myths about, 416t safety culture, 46-49 Safety Differently: Human Factors for a New Era, 395 Salesforce.com, 383-384 Sarbanes-Oxley Act, 267 Savoia, Alberto, 275n scenius, 54-55 Schmidt, Eric, 69 Schwaber, Ken, 122n Schwartz, Mark, 299 Scott, Kevin, 92, 93 Scrum, 122n, 410 Scrver, 251-253 Second Way, The. See The Second Way (principles of feedback) security. See information security security and compliance, DevOps compatibility with, xxv Security Chaos Engineering, 142n Security Monkey, 420 selflessness, as cultural pillar, 75 Senge, Peter, 35, 49, 320 separation of duty, 379, 384-387 service-oriented architectures (SOAs), 109, 210 settling period, 243 Shafer, Andrew Clay, xxiv, 5, 410 Shared Operations Team (SOT), 181-182n shared services, 117-119 Shewhart, Walter, 54 Shewhart cycle, 38, 54 Shingo, Shigeo, 27 Shinn, Bill, 389-391 Shook, John, 205 Shortridge, Kelly, 142n Shoup, Randy, xli, 109-110, 164, 207, 208, 209f, 210, 211f, 289, 291, 311, 326 silent majority, 74 siloization, 105 silos breaking down, 124-127 functional teams in, 125, 126f learning, 340

Simian Army, 369n, 420 single repository of truth as deployment pipeline foundation, 139-141, 150 for global improvement, 324-327, 334 at Google, 324-326 site reliability engineers (SREs), 269-271 Skelton, Matthew, 111, 117, 129 small batch sizes, 7, 9, 22-24f, 409 small teams, 110-111 Smart, Jonathan, 125, 341f, 403 smoke testing deployments, 180, 187 Smollen, Alex, 360, 361 smoothing, 253 Snover, Jeffrey, xxxii Snyder, Ross, 98, 99, 100 software, COTS, 417-418 software delivery. See also deployment pipeline foundations; low-risk release, enabling faster, xxxiiit history of, xxx-xxxit software supply chain, security of, 363-369 software type, myth about DevOps and, xxvi SolarWinds security breach, 368 Sonatype Nexus Lifecycle, 368n Sonatype State of the Software Supply Chain Report, 364, 365, 366 Sooner Safer Happier, 125, 340, 395, 403 source code repository, shared as deployment pipeline foundation, 139-141, 150 for global improvement, 324-327, 334 at Google, 324-326 preventive security controls and, 355-357 SPACE framework, 403 Spafford, George, xliv, 225, 233n, 353n, 415f Spear, Steven J., xxxvii, 34, 38, 49, 50, 58, 105, 305, 313, 335, 338 specialists, 105-107 Spotify, 15 spring cleanings, 337 sprint planning boards, 20-21 Sprouter, 98-100n, 108 SQL injection attacks, 373, 374f Stack Exchange, 278, 293 stakeholder feedback, 36 standard changes description of, 379-380 at Salesforce.com, 383-384 standups, daily, 117, 120, 122, 410 Starbucks, 15

startups, DevOps for, xxiv State of DevOps Reports, xln, xlii, 5, 14, 15, 57, 67, 71, 103, 140, 165, 166, 175, 185, 201, 217, 227, 243, 284, 285, 312, 326, 331, 343 State of the Octoverse report, GitHub's, 57, 368,401 static analysis, 358f-359 statistical analysis software, Tableau, 253 StatsD metric library, 234, 235f Stillman, Jessica, 338 Stoneham, Jim, 278-279, 280 strangler fig application pattern at Blackboard Learn, 215-217f blog explaining, 220 defined, 70, 208, 213-215, 218 stream-aligned teams, 112 Strear, Chris, 29-32 Sussman, Noah, 187 Sussna, Jeff, 265n swarming case study, 39-41 purpose of, 37-39 systems of engagement, 71, 72 systems of record, 71, 72

T

Tableau, 253 Taleb, Nassim Nicholas, 52 Tang, Diane, 276 Target API enablement, 112–114 approval processes, 296-297 case study, 112-114, 333-334, 342, 343 internal conferences, 342, 343 recommend_tech program, 333-334 thirty-day challenge, 335-336 as traditional enterprise, 15 task switching, 28 tax collection agency, UK's, 77-80 Team of Teams: New Rules of Engagement for a Complex World, 347 team opinion, requesting, 40 team rituals, 121-124 Team Topologies: Organizing Business and Technology Teams for Fast Flow, 111, 117, 129 teams 18F team, 369-371 Conway's Law and, 108 dedicated team, 86-88 four types of, 111-112 functional teams in silos, 125, 126f generalists on, 105-107 Google Web Server (GWS) team, 148-151

long-lived, 126f platform team, 112 Shared Operations Team (SOT), 181-182n stream-aligned, 112 two-pizza team, 110-111 in value stream, 83 technical debt brownfield transformations and, 67 defined, xxxiv-xxxv, 171, 205 reducing, 89-93, 165 rituals to pay down, 336-339 test automation and, 161, 165 technology choices Etsy, 332 organizational goals and, 329-330 Target, 333-334 Technology Enterprise Adoption Process (TEAP), 296, 297 technology value stream. See also value stream; value stream selection description of, 8-12 invisible work in, 19-20 telemetry. See also production telemetry; telemetry analysis application and business metrics, 240-242 application logging, 231-233 centralized telemetry infrastructure, 227-231 conclusions on, 244 culture of causality, 225, 233n daily work and, 234-235 defined, 225-226 detection of problems with, 36 Etsy, 226-227 gaps, 239-240 information radiators, 236, 237, 241 infrastructure metrics, 242-243 monitoring frameworks, 227-231 Netflix, 245-246 problem-solving guided by, 233-234 security-related, 371-375 self-service access to, 236-237 self-service metrics at LinkedIn, 237-238 settling period, 243-244 StatsD metric library, 234, 235f telemetry analysis alerting systems, 248–249, 250f, 255f anomaly detection techniques, 253-257 Gaussian distribution, 247f, 249 Kolmogorov-Smirnov test, 254, 255, 256f, 257 means and standard deviations, 246–248

non-Gaussian distribution, 249-251f outlier detection, 245-246 Scryer tool at Netflix, 251-253 smoothing, 253-254 Tableau software, 253 Terhorst-North, Dan, 191f, 193, 232 test setup and run, 26 test-driven development (TDD), 159, 161, 327 testing, A/B conclusions on, 280 feature toggles and, 196-197 history of, 275 integrating into feature planning, 278 integrating into feature testing, 276-277 integrating into our release, 277-278 need for, 273 practical guide to, 276 testing, automated Andon cord and, 163–165 categories of tests, 155-156 conclusions on, 166 deployment pipeline infrastructure, 151-154 essential components of, 165–166 fast and reproducible tests, 166 at Google, 148-151 green build and, 154, 163, 166 ideal testing pyramid, 157f-158 need for, 147-148 non-functional requirements and, 162-163, 328 observability and, 147 performance testing environment, 161-162 reducing reliance on manual tests, 160-161 research supporting, 165-166 running tests in parallel, 158f-159 test-driven development (TDD), 159, 161, 327 testing, operations, and security, as everyone's job, 104-105 Testing on the Toilet newsletter, 149n, 344 The First Way (principles of flow) case study, 29-32 constraint identification, 25-27 general description of, 13f, 19 handoff reduction, 24-25 kanban boards, 4, 20f-21, 94, 124 limiting work in progress (WIP), 7, 10, 21 - 22making work visible, 19-21, 93, 94, 124 reducing batch sizes, 22-24f small batch sizes, 7, 9, 22–24f waste and hardship elimination, 27-29

The Second Way (principles of feedback) case study, 39-41 description of, 13f-14 feedback types and cycle times, 36–37f optimizing for downstream work centers, 43-44 quality control closer to the source, 42 - 43safety within complex systems, 33-34 seeing problems as they occur, 35-36 swarming, 37-39, 41 The Third Way (continual learning and experimentation) case study, 54-55 conclusions on, 55-56 description of, 13f, 14 global knowledge, 51 improvement of daily work, 49-50 iust culture, 47, 307-308 leader's role, 52-54 resilience patterns, 51-52 safety culture for, 46-49 The Three Ways case study, 15-18 defined, 12-14 high-level principles of, xliv research supporting, 14-15 Theory of Constraints, 4, 412-413 Third Way, The. See The Third Way (continual learning and experimentation) Thoughtworks' Tech Radar, 57 Thrasher, Paula, 129 Three Mile Island, 34 Three Ways, The. See The Three Ways ticket, defined, 381n Ticketmaster/LiveNation, 104, 243 Tischler, Tim, 184 Tomayko, Ryan, 295, 296 tool-assisted code review, 290 Toyota Kata: Managing People for Improvement, Adaptiveness and Superior Results, 6, 49, 104, 411 Toyota Kata Movement, description of, 4, 6, 49, 54, 411 Toyota Production System Andon cord, 37-39, 163, 416-417f core belief in, 285 improvement blitz, 335 improvement kata, 6 information radiators, 236, 237, 241

Lean Movement and, 409 techniques, 4 transparent uptime, 311n, 420-421 Treynor Sloss, Ben, 269 Trimble, Chris, 86, 87 trunk-based development adopting practices of, 172 at Bazaarvoice, 173-175, 176 benefits of, 175 conclusions on, 175-176 gated commits and, 172 for HP's LaserJet Firmware division, 168-170 need for, 167, 171 Trustworthy Online Controlled Experiments: A Practical Guide to A/B Testing, 276 TurboTax, 274, 275 Turnbull, James, 228, 230f Twitter architecture, 210 experimentation, 279 Fail Whale error page, 360 static security testing, 360-363 two-pizza team, 110-111

U

Unicorn Project, The, 219, 333n unit tests, 155 universality of the solution, xlii–xliii urgent changes, 380 US Air Force, 69–71 US Federal Government agencies, 369–371 US Navy, 34n, 51 user acceptance testing (UAT), 152 user feedback, 36, 37f user research, 273, 275n, 277. See also A/B testing user stories, 328–329 UX movement, Lean, 411 UX observation, 264–265

V

value stream. *See also* value stream selection defined, 7, 18 manufacturing, 7 selection of, 61, 63 technology, 8–12 value stream manager, 83 value stream mapping %C/A metrics in, 11, 85, 95 case studies, 15–18, 91–93

conclusions on, 85 creating a map, 84-86 defined, 81, 84-86 dedicated transformation team and, 86-88,95 example of a value stream map, 85f goals and, 88 Lean movement and, 4 members of value stream, 83 at Nordstrom, 81-83 technical debt and, 89-93 Value Stream Mapping: How to Visualize Work and Align Leadership for Organizational Transformation, 7 value stream selection American Airlines' new vocabulary, 74–77t brownfield transformation of refueling system, 69-71 case studies, 69-71, 74-77, 77-80 expanding scope of initiative, 73-77 greenfield vs. brownfield services, 66-69 innovators and early adopters, 72–74 Nordstrom's DevOps journey, 63-66 systems of engagement, 71, 72 systems of record, 71, 72 tax collection agency's journey, 77-80 van Kemenade, Ron, 74 Van Leeuwen, Eveliin, 343n Vance, Ashlee, 92 Velocity Conference, 410 Verizon data breach, 364, 371 version control system, 139-141 versioned APIs, 214 Vincent, John, 247 visibility of automated test failures, 164 visibility of work, 19-21, 93, 94 Visible Ops Handbook, The, 225 Visible Ops Security, 353n visual work boards, 20f-21. See also kanban boards vocabulary, using a new, 76–77t Vogels, Werner, 110, 212

Vulnerabilities and Exposures, Common (CVEs), 364, 365 Vulnerability Database, National, 368

W

Walker, Jason, 333 Wang, Kendrick, 277 WARN level, 232 waste and hardship elimination, 27-29 water-Scrum-fall anti-pattern, 165n weak failure signals, 313-314 Westrum, Ron, 47, 48t, 57 Westrum Organizational Typology Model, 48t Wickett, James, 353 Williams, Branden, xxv Williams, Jeff, 412 Williams, Laurie, 159n, 293 Willis, John, xxiii-xxiv, 3, 5, 57, 406-407, 409,410 Wolberg, Kirsten, 73n Womack, James P., 23, 53 Wong, Bruce, 316 Wong, Eric, 238 work, visibility of, 19-21, 93, 94, 124 work in process (WIP), 7, 10, 21-22 workplace safety, at Alcoa, 50

х

Xu, Ya, 276

Y

Yadav, Yikalp, 286 Yahoo! Answers, 278–280 Yuan, Danny, 252*f*, 254*f*

Z

Zenoss, 229, 238 Zhao, Haiping, 338 ZooKeeper, 242 Zuckerberg, Mark, 338

ACKNOWLEDGMENTS

Gene Kim

Thank you to my wife, Margueritte, and my sons, Reid, Parker, and Grant, for putting up with me being in deadline mode for over five years. And to my parents for helping me become a nerd early in life. Thank you to my amazing co-authors, as well as the IT Revolution team who made this book possible: Anna Noak and especially to Leah Brown for all her heroic work pulling this second edition together!

I am so grateful to everyone who made the first edition possible: John Allspaw (Etsy), Alanna Brown (Puppet), Adrian Cockcroft (Battery Ventures), Justin Collins (Brakeman Pro), Josh Corman (Atlantic Council), Jason Cox (The Walt Disney Company), Dominica DeGrandis (LeanKit), Damon Edwards (DTO Solutions), Dr. Nicole Forsgren (Chef), Gary Gruver, Sam Guckenheimer (Microsoft), Elisabeth Hendrickson (Pivotal Software), Nick Galbreath (Signal Sciences), Tom Limoncelli (Stack Exchange), Chris Little, Ryan Martens, Ernest Mueller (Alien-Vault), Mike Orzen, Christopher Porter (CISO, Fannie Mae), Scott Prugh (CSG International), Roy Rapoport (Netflix), Tarun Reddy (CA/Rally), Jesse Robbins (Orion Labs), Ben Rockwood (Chef), Andrew Shafer (Pivotal), Randy Shoup (Stitch Fix), James Turnbull (Kickstarter), and James Wickett (Signal Sciences).

I also want to thank the many people whose incredible DevOps journeys we studied, including Justin Arbuckle, David Ashman, Charlie Betz, Mike Bland, Dr. Toufic Boubez, Em Campbell-Pretty, Jason Chan, Pete Cheslock, Ross Clanton, Jonathan Claudius, Shawn Davenport, James DeLuccia, Rob England, John Esser, James Fryman, Paul Farrall, Nathen Harvey, Mirco Hering, Adam Jacob, Luke Kanies, Kaimar Karu, Nigel Kersten, Courtney Kissler, Bethany Macri, Simon Morris, Ian Malpass, Dianne Marsh, Norman Marks, Bill Massie, Neil Matatall, Michael Nygard, Patrick McDonnell, Eran Messeri, Heather Mickman, Jody Mulkey, Paul Muller, Jesse Newland, Dan North, Dr. Tapabrata Pal, Michael Rembetsy, Mike Rother, Paul Stack, Gareth Rushgrove, Mark Schwartz, Nathan Shimek, Bill Shinn, JP Schneider, Dr. Steven Spear, Laurence Sweeney, Jim Stoneham, and Ryan Tomayko.

And I am so profoundly grateful for the many reviewers who gave us fantastic feedback that shaped this book: Will Albenzi, JT Armstrong, Paul Auclair, Ed Bellis, Daniel Blander, Matt Brender, Alanna Brown, Branden Burton, Ross Clanton, Adrian Cockcroft, Jennifer Davis, Jessica DeVita, Stephen Feldman, Martin Fisher, Stephen Fishman, Jeff Gallimore, Becky Hartman, Matt Hatch, William Hertling, Rob Hirschfeld, Tim Hunter, Stein Inge Morisbak, Mark Klein, Alan Kraft, Bridget Kromhaut, Chris Leavory, Chris Leavoy, Jenny Madorsky, Dave Mangot, Chris McDevitt, Chris McEniry, Mike McGarr, Thomas McGonagle, Sam McLeod, Byron Miller, David Mortman, Chivas Nambiar, Charles Nelles, John Osborne, Matt O'Keefe, Manuel Pais, Gary Pedretti, Dan Piessens, Brian Prince, Dennis Ravenelle, Pete Reid, Markos Rendell, Trevor Roberts, Jr., Frederick Scholl, Matthew Selheimer, David Severski, Samir Shah, Paul Stack, Scott Stockton, Dave Tempero, Todd Varland, Jeremy Voorhis, and Branden Williams.

And several people gave me an amazing glimpse of what the future of authoring with modern toolchains looks like, including Andrew Odewahn (O'Reilly Media) who let us use the fantastic Chimera reviewing platform, James Turnbull (Kickstarter) for his help creating my first publishing rendering toolchain, and Scott Chacon (GitHub) for his work on GitHub Flow for authors.

Jez Humble

Creating this book has been a labor of love, for Gene in particular. It's an immense privilege and pleasure to have worked with Gene and my other co-authors, John and Pat, along with Todd, Anna, Robyn and the editorial and production team at IT Revolution preparing this work—thank you. I also want to thank Nicole Forsgren whose work with Gene, Alanna Brown, Nigel Kersten and I on the PuppetLabs/DORA *State of DevOps Report* over the last three years has been instrumental in developing, testing and refining many of the ideas in this book. My wife, Rani, and my two daughters, Amrita and Reshmi, have given me boundless love and support during my work on this book, as in every part of my life. Thank you. I love you. Finally, I feel incredibly lucky to be part of the DevOps community, which almost without exception walks the talk of practicing empathy and growing a culture of respect and learning. Thanks to each and every one of you.

Patrick Debois

I would like to thank those who were on this ride, much gratitude to you all.

John Willis

First and foremost, I need to acknowledge my saint of a wife for putting up with my crazy career. It would take another book to express how much I learned from my co-authors Patrick, Gene and Jez. Other very important influencers and advisers in my journey are Mark Hinkle, Mark Burgess, Andrew Clay Shafer, and Michael Cote. I also want to give a shout out to Adam Jacob for hiring me at Chef and giving me the freedom to explore, in the early days, this thing we call Devops. Last but definitely not least is my partner in crime, my *Devops Cafe* cohost, Damon Edwards.

Nicole Forsgren

I'm grateful to Jez and Gene, whose collaboration on the *State of DevOps Reports* (and later DORA) provided a wonderful research lab and foundation for work. The partnership wouldn't have been possible without the insights of Alanna Brown, who spearheaded the initiative and who we were able to work with on those first few reports. I continue to be grateful for those who believe in women—and believe women—with ambition, ideas, and opinions. Some notable people in my life are my parents (sorry for the stress!), my dissertation advisors Suzie Weisband and Alexandra Durcikova (much love and respect to you for taking a chance on me and my pivot), Xavier Velasquez (you've always believed my plans will work), and of course my lady backchannel (thank you for love, support, and for space to rage so it doesn't end up on my Twitter timeline). And as always, Diet Coke.

ABOUT THE AUTHORS

Gene Kim is a bestselling author, researcher, multiple award-winning CTO, and founder of IT Revolution. His books include *The Phoenix Project*, *The Unicorn Project*, and *Accelerate*. Since 2014, he has been the founder and organizer of DevOps Enterprise Summit, studying the technology transformations of large, complex organizations.

Jez Humble has co-authored several bestselling books on software, including Jolt Award-winner *Continuous Delivery* and Shingo Publication Award winner *Accelerate*. He works at Google and teaches at the University of California, Berkeley.

Patrick Debois is Director of DevOps Relations and Advisor at Snyk. He works to bridge the gap between projects and operations by using Agile techniques in development, project management, and system administration.

John Willis is Senior Director of Global Transformation Office at RedHat. Hehas worked in the IT management industry for more than thirty-five years. He is coauthor of *Beyond The Phoenix Project* and host the Profound podcast.

Nicole Forsgren, PhD, is Partner at Microsoft Research leading the Developer Velocity Lab. She is author of the Shingo Publication Award-winning book *Accelerate: The Science of Lean Software and DevOps* and is best known as lead investigator on the largest DevOps studies to date. She has been a successful entrepreneur (with an exit to Google), professor, performance engineer, and sysadmin. Her work has been published in several peer-reviewed journals.