PRAISE FOR
THE VALUE FLYWHEEL EFFECT

"*The Value Flywheel Effect* is a timely and hugely valuable book providing a set of principles and ways of working to navigate and exploit the fast-changing technology and business landscape. Without sense-making capabilities, organizations are increasingly struggling to act effectively in the modern disruptive technology context, so much so that the ideas in *The Value Flywheel Effect* amount to a kind of 'survival handbook' for the next decade or two. *The Value Flywheel Effect* is essential reading for any leader or practitioner who wants to help their organization to survive and thrive."

—**Matthew Skelton**, Co-Author of *Team Topologies: Organizing Business and Technology Teams for Fast Flow*

"I'm delighted to see that David, Mark, and Michael have taken the time and considerable trouble to explain how they architected and delivered serverless innovation at Liberty Mutual. Why? Because change is the only constant, and the pace of technology innovation is accelerating this change. As Jon F. Kennedy said 'those that look only to the past or the present are certain to miss the future'. This book is required reading if you want to leverage serverless technology."

—**Dr. Jacqui Taylor**, CEO and Co-Founder, FlyingBinary, and #15 Most Influential UK Technologist

"The future for company creation and innovation relies on speed, scalability, and sustainability. To nurture thriving products, organizations, and cultures, you need to understand the landscape you're competing in, along with the path to choose to succeed. Packed with real-world examples, case studies, and practical tools applied to complex domains, *The Value Flywheel Effect* is required reading to plot your future and the business you want to be."

—**Barry O'Reilly**, Co-Founder Nobody Studios, Author of *Unlearn* and *Lean Enterprise*

"Required reading for every level of an organization adopting modern cloud, and grounded in unparalleled lived experience of transformation. I'm sure I'll be referencing this for years."

—**Ben Ellerby**, Founder, Aleios, and AWS Serverless Hero

"The journey to the cloud is one of leadership and change management rather than purely a technical pathway. *The Value Flywheel Effect* is a great foundation for any organization on a modern cloud journey. Based on how it's applied in the real world, to real problems, there are great insights into how to navigate waves of change."

—**Seamus Cushley**, VP Product Development, Bazaarvoice

"Serverless architecture combined with clarity of purpose is a lethal combination for winning in today's competitive landscape. *The Value Flywheel Effect* provides powerful insights on how you can leverage these modern practices to navigate an enterprise cloud transformation and accelerate the delivery of value."

—**Drew Firment**, SVP, A Cloud Guru

"*The Value Flywheel Effect* shares a surprising collection of leadership and transformation practices that are sure to help you stir up the right kind of trouble, no matter where you are in the organization. Also an excellent reference for your Wardley Mapping practice, with helpful guidance, plenty of examples, and new things to try as well!"

—**Ben Mosior**, LearnWardleyMapping.com

"*The Value Flywheel Effect* presents a practical road map to innovation, focusing on a modernization strategy that takes full advantage of modern cloud capabilities. Engineers and their leadership will learn real-world techniques based on the authors' experience evolving a value-first approach at a Fortune 100 company to reinvent traditional enterprise development."

—**Sam Dengler**, AWS Principal Developer Advocate

"*The Value Flywheel Effect* is a valuable read for the technology leaders of today and tomorrow. It successfully weaves tools like Wardley Mapping, systems thinking, north star metrics, and the serverless mindset into one cohesive framework for organizational success."

—**Ajay Nair**, Director, Amazon Web Services

"In working together with the authors for several years, I was always inspired and challenged as a leader to ensure engineering excellence was the first point of focus for myself and my teams. This book does a fantastic job laying out the 'flywheel' impact they identified. I can say that while it can perhaps feel elusive, it is real and ended up being far more powerful than I ever imagined. A must read!"

—**John Heveran**, EVP & CIO Global Risk Solutions,
Liberty Mutual Insurance

"There are many resources on the technical 'how' of serverless, but there is very little out there on the higher-level 'why' of choosing a serverless-first approach to software engineering in organizations. If you're a senior technology decision maker and you care about delivering value fast and frequently to your users while also minimizing the total cost of ownership and cultivating an environment in which your engineers can thrive and become self-sufficient, *The Value Flywheel Effect* will provide you with a framework and tools to make the strategic decisions to deliver on these goals."

—**Paul Swail**, Independent Cloud Consultant, ServerlessFirst.com

"*The Value Flywheel Effect* is a must-read for anyone looking to lead successful software organizations in the modern cloud era. Without a doubt, this is one of the most practical guides ever written to help teams create and rally around a shared vision, develop customer-centric road maps to meaningfully impact north star metrics, and execute well-thought-out strategies to achieve extraordinary outcomes. It strikes an excellent balance between teaching you the methods, theories, and practices used, and their implementation using clear examples and real-world case studies. Dave, Mark, and Michael are masters at what they do, and this book imparts their shared expertise into the go-to instruction manual for tomorrow's technology leaders."

—**Jeremy Daly**, AWS Serverless Hero, Publisher of OffByNone.io,
and Host of *Serverless Chats*

"If you are in the business of producing software, no matter your role, you need to read this book. It does more than join the dots of what we already know. Like all the best advancements, it somehow pulls the very clever trick of creating a whole new perspective on software delivery, and yet feels both intuitive and obvious at the same time. How important are the ideas in this book? Well, I would go as far as to say that if you are not using 'the value flywheel philosophy' as the basis of your own delivery, then those that do will win. Simple as that. It's that good. And that important."

—**Tara Simpson**, CEO, Instil Software

the Value Flywheel Effect

POWER THE FUTURE AND
ACCELERATE YOUR ORGANIZATION
TO THE MODERN CLOUD

the Value Flywheel Effect

DAVID ANDERSON
with Mark McCann & Michael O'Reilly

Forewords by Adrian Cockcroft *and* Simon Wardley

IT Revolution
Portland, Oregon

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

First Edition

Printed in the United States of America
27 26 25 24 23 22 1 2 3 4 5 6 7 8 9 10

Cover and book design by Devon Smith/D. Smith Creative, LLC

Library of Congress Control Number: 2022939507

ISBN: 9781950508570
eBook ISBN: 9781950508587
Web PDF ISBN: 9781950508594
Audio: 9781950508600

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

THE VALUE FLYWHEEL EFFECT

To my partner, Treasa, my children, Sarah & Thomas, and all my family for all the love and support. Thanks to all the engineers for taking the time to "see what you think."

—Dave

To Mairead, Aoife, and my immediate family for the love and support, and my inspirational colleagues from whom I am always learning. I am eternally grateful.

—Michael

To my amazing wife, Gillian, and my awesome kids, Isabella and Lucy. Thanks for all the love, support, and inspiration.

—Mark

CONTENTS

PART IV: PHASE THREE: NEXT BEST ACTION

PART V: PHASE FOUR: LONG-TERM VALUE

LIST OF TABLES & FIGURES

FOREWORD
BY ADRIAN COCKCROFT

Back in 2010, I gave my first public presentation trying to explain what Netflix was doing, how we were moving to the cloud using open source, the distributed architecture later known as microservices, chaos testing, and how developers were on-call operating the service, which became one of the flavors of DevOps. The reaction was a mixture of bafflement, a general opinion that we were a weird "unicorn" that no one else could copy, and that we would soon be back in our datacenter running enterprise licensed products when we gave up trying to make it work.

However, there were a few people who were more curious and excited, and there were a lot of discussions on Twitter with a group known as the #clouderati. I met Simon Wardley in that group and explained in detail to him, as part of his research, what we were doing. In return, he explained his mapping techniques to me. These maps helped me understand that even though Netflix wasn't using maps, it did have excellent situational awareness, an appropriate use of doctrine, a systems-thinking approach, and was well aligned with many of the best practices that Simon described.

In the last twelve years, as the ideas moved from crazy to mainstream, a big part of my job has been to try to explain and apply the ideas developed at Netflix and Amazon to other organizations and applications, big and small. This book is the best distillation of how to do that that I've encountered. This book uses Wardley Mapping to take today's best ideas and make sense of when to use what doctrine or technique in which situation.

The ideas we developed at Netflix were synthesized from what came before. Many of us had decades of experience to draw from and were familiar with Frederick Brooks's *The Mythical Man-Month* and Conway's Law. We also absorbed Werner Vogels's "Run What You Wrote" *ACM Queue* story from 2006[1] and were inspired by open-source development practices. The Netflix team brought ideas we'd developed at Sun Microsystems, Xerox PARC, and the early days of eBay, Google, and Yahoo. Netflix CEO Reed Hastings deeply understands software, and he encouraged us to start afresh and build an architecture that would scale and support innovation.

Some attempts to copy the ideas we developed at Netflix missed that they are artifacts of a dynamic underlying system, where the parts reinforce each other using principles and doctrine to organize in a fluid way, rather than rules and best-practice patterns that are more rigid and evolve slowly. This often led to "architecture theater" and failed attempts by organizations trying to copy Netflix's success.

Many of these organizations also struggled with the question of how to get "The Business" to let the technology organization work more like Netflix or Amazon. One CIO told me that they couldn't copy Netflix because they didn't have the super talented engineers we did. I responded that we had just hired someone from his organization.

Talent isn't the problem. The problem is the very idea of having a separate business organization. No one at Netflix or Amazon ever talks about "The Business." When I worked there, Netflix was organized as a single-product organization. Everyone in product management and development reported to the chief product officer. There wasn't a CIO or CTO in charge. Amazon is organized into a huge number of independent service teams—none of them report to the CIO or CTO.

At the end of 2014, AWS Lambda was launched. I thought it was interesting then, but at the end of 2016, I joined AWS and was judging a one-day AWS re:Invent hackathon. I was surprised to see every team choosing to build serverless architectures using Lambda, and I was also amazed to see what they were able to build from scratch in one day. I started to tell this story in my microservices workshops and found some audience members responded with similar stories of huge amounts of functionality being developed by tiny teams in very little time and with very low operating costs.

This was eye opening, but the problem was that serverless seemed like a fairytale. When it worked, the results were better by an order of magnitude or more—a ludicrous improvement that most people discounted as fantasy or something that only "unicorns" could use.

Like with the ideas I expressed in my initial Netflix talks in 2010, most people remained baffled or dismissive, but a few latched onto the idea of serverless. Simon Wardley was also paying attention, mapped the evolution of the cloud, and declared that serverless was the future. Meanwhile, AWS was systematically removing every objection that customers raised as to why they couldn't do serverless, and I started giving talks titled "Serverless-First," which expressed my opinion that organizations should try to build everything using serverless as their first attempt and only then fall back on containers and specialized instance types when they really have to.

I was talking to lots of AWS customers at that time, and at some point, I connected to Liberty Mutual and discovered David Anderson and his team. We clicked immediately and set up a regular meeting that we continued over several years. The

systematic approach David and his team were taking at Liberty Mutual put together the latest best practices, including serverless, and used Wardley Maps to make sense of how to apply these practices.

The amazing thing was that this old insurance company had built one of the most innovative and fast-moving development organizations I was aware of. They were going so fast and at such a low cost that product teams stopped looking at competing technology platforms until they had tried serverless-first. When some people talked about using Kubernetes by default to avoid lock-in, the response was, "Why should I spend ten times as much in time and money? If we need to make it portable later for some reason, we will spend the time and money then." They also told me that the bottleneck for delivering products had moved to the product managers, as ideas were being built faster than new ideas were being figured out. They explained the advantages of serverless-first all the way up the management chain as a core advantage for the company.

I recently retired, left Amazon, and don't have to explain what we did at Netflix or Amazon as my day job anymore. I'm happy to pass that baton to David, Mark, and Michael, who via this excellent book, will spend the next few years explaining what they did at Liberty Mutual to the next generation of baffled and curious audiences.

—Adrian Cockcroft
2022

FOREWORD
BY SIMON WARDLEY

All entities strive for success in the game of life. But success assumes there exists some form of competition, either with the environment or other entities or both. Competition (the act of seeking together) comes in many forms—conflict, cooperation, and collaboration. The way we undertake these forms are governed by the age we live in. Competition today is not the same as competition yesterday. It is instead sculptured by the technology and practices available to us and the age in which we live.

This book explores the practices and technology of the modern age and the beginning of what many have dubbed the Fourth Industrial Revolution. It examines how the business and technology can communicate seamlessly together, the importance of technology changes (such as serverless), and why situational awareness of our landscape matters.

Competition in the past was often some form of conflict over territory described through a map of the physical landscape. Those ideas of conflict often extended within the organization, with departments jostling for control. Today's competition is over the supply chains that underpin our technological marvels. This book explains how to map that new territory, the landscape of components that make our businesses, and how to achieve this through collaboration that does not conflict between business and technology.

This book also examines these practices, not from the standpoint of theory or some consultancy's favorite "PowerPoint strategy deck," but from the very act of practice itself, covering the different journeys of a traditional $40 billion revenue, one-hundred-plus-years-old insurance company to a modern software start-up that sold for almost $2 billion. This diversity of experience matters because there are no copycat lists or "one and done" exercises for success; instead, there are sets of practices that need to be applied to your context. The business of insurance is not the same as the business of training, but there are practices that can be applied across both.

At the heart of this is a concept known as the Value Flywheel Effect (the title of this book), derived from a paper napkin sketch by Jeff Bezos, founder of Amazon, which itself was an adaptation from the work of Jim Collins. If you wish to survive and thrive in this Fourth Industrial Age, if you wish to understand the landscapes you are competing in, if you wish to understand how modern practices will change your organization, then I would recommend that you read and study this book carefully.

I'm certain Laura and Clive would agree.

—Simon Wardley
2022

INTRODUCTION

After twenty-five years in the technology industry, I can now look back and empathize with all the people I drove crazy—mostly IT managers. Let's face it, software engineers are usually hired to build things quickly, not fix the sociotechnical issues of the larger organization. And yet, that is what I have found myself drawn to time and time again. After all, I have always believed that if you fix the system, then everyone can build quicker, and the business can deliver value sooner. The power of the group is always greater than the power of one.

Whether I was working on improving security, introducing Agile working methods, creating good engineering principles, improving enterprise architecture, building a machine learning (ML) capability, modernizing systems, designing cloud platforms, fixing the developer experience—you name it—I always believed that improving the larger system would improve everything else.

Over the years, I've lost count of the number of times a manager has asked me, "Why are you doing *that* task? I need you to write more code!" I would always be polite and point out that writing code is important, but improving the system is more important. Of course, plenty of code was written, but code is a liability. When we eliminate the burden of something like infrastructure concerns for software developers, it creates room for focus elsewhere, such as developing software assets.

I have been writing code since I was nine; therefore, you could say I've always been a coder. But being a *software engineer** is different. A software engineer should be hired to solve problems and create value for the business. Most software engineers code as a pastime, not as a job. Code written "on the job" should be part of a larger value creation effort, not an effort to write X lines of code in X hours.

---

* Mathematician Margaret Hamilton was the first programmer hired by MIT to work on the NASA Apollo missions and coined the job title "software engineer." She had been tasked with designing software programs for the guidance computers on Apollo. To legitimize the importance of her work, she immediately changed her title from programmer to software engineer, as she felt she was "just as much an engineer as the men who were building the spacecraft."[1]

Unfortunately, too many people in IT write code, build systems, and perform tasks without any idea why. They obsess over function and forget about purpose.* And too many people from the business see software developers and engineers as nothing more than programmers who should just make the system do what the business says it should.

In my experience, I've taken huge risks by focusing on these larger sociotechnical systems. I've had to deal with a lot of push back from middle management for not doing what I was supposed to (i.e., write more code). But I always had the conviction to push on. It was like I was in a poker game and sitting on a hand that would pay big. When you see that kind of opportunity, you must take it. Taking that risk, playing that hand, ended up paying off big.

Serverless Transformation at Liberty Mutual

In 2013, Liberty Mutual Insurance, the sixth-largest property and casualty insurance company, started to move its services to the cloud, and I was lucky enough to find myself part of the transformation.

I had joined the company back in 2007, spending a few years designing and building a large eCommerce platform with co-authors Mark McCann and Michael O'Reilly as well as many other talented engineers and leaders.

I was impressed with the quality of individual engineers at Liberty Mutual, but I could see there was still a significant opportunity to create value. I could see that the connection between the business and technology needed work. And (most importantly) I could see that the engineers were crying out for change. But, like many legacy enterprises, Liberty Mutual was like a big oil tanker—any attempt to steer it in a new direction was going to be difficult and very slow.

By 2013, I moved into a CTO position in Belfast, Ireland. I had built a small team of architects and a solid technical leadership community. Liberty Mutual had begun exploring solutions for security and test data in the cloud using AWS. This was a significant opportunity. I quickly realized that the cloud was not just another datacenter; it could offer a transformational way of working. I just didn't quite know what that was yet. I decided that my team and I would try and figure out a way to build better software in the cloud. The cloud was new to many of us, and "application development" was our area of expertise. I could sense a paradigm shift and knew it was time to start exploring.

---

* Simon Sinek elaborates on this eloquently in his depiction of the golden circle and in his book *Start with Why*.[2]

I wondered, "What does cloud application architecture look like in this brave new world?" I knew it would take a few years for the foundational capabilities (security, governance, infrastructure, processes) to be ready at scale, but I was certain which direction we should take. We had a window of opportunity. I just needed a way to map a path, even if that map would change along the way.

Mark and I had begun following the work of Simon Wardley. We were enjoying his technique, called Wardley Mapping, which is a method for building situational awareness to map out a potential business strategy. We didn't fully understand it yet, but it felt exciting and described the evolution we knew was coming. Mapping allowed us to ask questions about how things could evolve; it let us peek into the future and turn hunches into strategies that we could then either test or look for early signs. We decided to attempt to map out what we thought could happen with Liberty Mutual's shift to the cloud.

We asked ourselves hard questions:

- Will we still write thousands of lines of code in this new place?
- Will infrastructure as code happen? What will it look like?
- What happens when continuous delivery is in place?
- How will the cloud providers like AWS evolve?
- What things do we do now that we won't do in the future?
- What will be valuable for our business when all of this is complete?

We spent many hours writing lots of rubbish on dry-erase boards. The architect team grew, and we experimented and worked with peers across the organization to try and better understand the cloud landscape. We didn't know it then, but we were building situational awareness and informing our maps.

Eventually, the team could sit in a small room and discuss the entire technology landscape of a Fortune 100 company and use Wardley Mapping to predict what might happen in the next year, five years, etc. We had become a sensemaking machine. This, fortuitously, coincided with the launch of AWS Lambda.

AWS Lambda was a significant innovation in cloud technology and provided the opportunity for a huge mindset shift for developers. Previously, organizations moving to the cloud still had to manage the infrastructure (costing hours and hours of developer time). But with Lambda, there was now the option to leave the infrastructure management to the cloud vendor! This could move the cloud from a product to a commodity the business consumed like electricity. This would give our teams more time to focus on creativity and innovation, including operational constraints, performance constraints, the total cost of the solution, and user experience, instead

of just keeping the lights on. Teams could start to write systems rather than simply applications. And there were clear cost savings for the business as well.

This was the beginning of what came to be known as serverless computing, where companies no longer managed their cloud operations themselves but gave that toil over to the cloud vendors. With this model, an organization could run an application when needed, shut it down when needed, and pay only when it was being used.

But we had a problem: How could we use the cloud to create business value? The answer had presented itself: serverless. The technology was very raw, but we could see the potential. My team and I believed we had a map for the future of technology. We could see the trends that would disappear. We could see the capabilities that would be critical in the future, and we could guess how the cloud providers might evolve. We had that winning poker hand that I alluded to earlier. We decided to take the plunge and experiment in this new serverless world. Our experiments quickly accelerated our engineering team's ability to focus on things other than infrastructure management. And the more we experimented in this space, the more we started to see a flywheel effect, small wins accumulating over time to drive momentum. We were delivering more value into the hands of our business partners faster. And we started to see that the cloud was more than just another datacenter. It provided a transformational way of working.

We tried to map out what we were experiencing. First, we had a clear purpose (Phase 1). Next, we had the right environment in which to thrive (Phase 2). And serverless-first architecture provided us with the next best action (Phase 3) we could take to create long-term value (Phase 4) for our organization.

We deployed this same pattern many times, spinning the flywheel again and again, creating more and more momentum and less and less inertia, and the success was evident. Engineers were moving faster, creating lower-cost solutions and more innovative approaches with a better connection between technology and the business.

As was reported by *TechRepublic*, a single web application at Liberty Mutual was rewritten as serverless and resulted in reduced maintenance costs of 99.98%, from $50,000 a year to $10 a year.[3] That small savings is hugely powerful when you have hundreds or thousands of similar applications running at the same time. I've seen this type of successful pattern repeated at Liberty Mutual and across different industries.

Thanks to serverless and our flywheel, we were also able to release applications quicker, which meant getting feedback from users and customers sooner. This in turn gave us an advantage in the market to more rapidly respond to customer demands and changes. Serverless also opened new possibilities that seemed too

costly or difficult before, such as integration with AI and data services or event streaming services.

This new mindset shifted our perspective of code as well. Instead of an asset, we began to see that more code was a liability. The less code we wrote, the better. And the code we did write must have demonstrable business value. Surprisingly, our software engineers loved this shift to writing less code. Many didn't want to go back to "the old ways" of doing things.

We started to adopt what came to be known as *serverless-first architecture*. In other words, a team's first implementation choice should be serverless, and if that's not a good fit, then you work backward (i.e., introduce more infrastructure, like containers).

By 2019–2020, things on the cloud front had progressed significantly. At one point, I had four AWS Heroes (individuals recognized as AWS community experts and who enjoy legendary status!) in my extended team, which was unheard of at the time. Many of the engineers on my team were giving talks and keynotes about our successes at major AWS and technology conferences. And the business metrics delivered became simply unbelievable: 95%+ runtime cost-savings, new functionality delivered months ahead of schedule, global roll out in weeks instead of years, and innovative features leading the market and deploying multiple times a day.

Also, we created a software accelerator using the AWS Cloud Development Kit (AWS CDK), an open-source software development framework in which engineers can use familiar programming languages to define cloud application resources, to help deploy new applications quickly. These acted like code templates that could be used by software engineers to rapidly build projects rather than writing the code from scratch.

I was frequently challenged at external events, as these metrics seemed far-fetched (for example, cost savings of 99.5% and similar). Amazon CTO Werner Vogels even started to praise Liberty Mutual and our serverless-first architecture, calling it organizational nirvana.[4] Our "Serverless-First Enterprise" concept was starting to take hold.

In a global organization with thousands of people, the sociotechnical element of driving a paradigm shift like this is significant; moving to the cloud and embracing a new way to write software might only happen once in your career. It's not enough to have some cool tech. Winning hearts and minds is the real challenge.

By this time, we had been evolving our methods and practices for ten years, so we decided to encapsulate them in a set of principles we referred to as the "Serverless-First Organization Strategy." The running joke I had with the tech leads was that it was impossible to measure if the principle was true, but blindingly obvious when

it was not! We tried to paint a picture of our ideal software development team in a set of principles as follows:

A high-performing, serverless-first team will:

1. chase a business outcome (KPI)
2. be secure by design
3. keep high throughput of work
4. reliably run a high-stability system
5. rent/reuse, with build as the final option
6. continuously optimize the total cost
7. build event-driven via strong APIs
8. build solutions that fit in their heads

We knew from experience that a successful team needed to know what they were doing and should be able to recite the business metrics from memory (principle #1).

We knew that security and threat modeling were table stakes; it's everyone's job (principle #2).

DORA's metrics for high-performing teams (as illustrated in the book *Accelerate* by Dr. Nicole Forsgren, Jez Humble, and Gene Kim),[5] provided clear standards for code quality (principles #3 and #4).

We worked to spread the attitude of the evolution of technology and the idea that code is a liability (principle #5).

Trying to encourage builders that you don't always need to build is a significant challenge. The great cloud principle and mindset change needed is cost and OpEx. Teams need to be aware of the cost, not to save the company money, but to think frugally and efficiently (principle #6).

We set the bar high for integration patterns and encouraged clean design (principle #7).

And finally, borrowed from our friend and mentor Dan North, we held that software should fit in your head.[6] It shouldn't be overcomplicated (principle #8).

We often joke that it took ten years to write these eight bullet points, but each represents a lesson learned. The principles landed well, and after considerable risk, many engineers learned in this environment, contributed, and succeeded. Even today, serverless is still not widely accepted. Learning a brand-new way of writing code could set an engineer's career back considerably, especially if the technology doesn't "win." It's always a leap of faith when moving to a new technology.

The Value Flywheel Effect Materializes

Of course, lockdown changed everything, and we all retired to our home offices and conference calls in early 2020. The "Serverless-First Organization Strategy" still stood up. It gave the engineering teams a clear focus, which was build well for engineering value. The collaboration continued, if in a different way. But it was clear that the flywheel we had discovered (having a clarity of purpose, the right environment, a clear next best action, and creating long-term value) was fully in effect now. Not even a pandemic could stop the flywheel from spinning.

As I made sense of this "Value Flywheel Effect" approach that we had worked out, I knew I needed to stress test the thinking with true industry thought leaders—people I respected and who I had followed for years. Life is full of gambles, but I figured that in lockdown, people might have more time on their hands. I (honestly) had only two names on my list—Simon Wardley, who had created Wardley Mapping, and Adrian Cockcroft, VP of Cloud Architecture at AWS (at that time).

I had been listening to and reading the work of both leaders for over ten years, but I didn't have a personal relationship with either of them. After a bit of effort, I tracked them down and asked for their feedback.

My question was simple: I think this idea of the Value Flywheel Effect is good, but why is no one else doing this? What am I missing? Both (who I later learned are friends) separately told me: "No, you're not mad. This is good stuff. Let's talk more."

That positive response and the invitation to collaborate had not been on my map. I sat in Belfast deciding what to do next. Two of my heroes had just told me I'd hit proverbial gold. Well, this book is the result.

In this book, I've distilled this set of practices (what I've termed the Value Flywheel Effect) learned from people "not doing what they were supposed to." The practices are all real. They are derived from real experiences, real scars, real success. This is not a book about technology (even if there's a healthy dose in it). And it's not a book about job functions. This is a book for all the business leaders, at any level, who aren't afraid of not doing what they're supposed to. This is a book for leaders of the future who will forage a clearer alliance between the business and technology to navigate the unknown waters ahead of us.

But before we go any further, I must ask you to forget everything you know about the IT departments and technology teams you've worked in or with in the past. To succeed on our journey, we must get away from the mental and physical image of IT as a separate entity or department. Instead, we must create a shared, ambitious goal for technology and the business, as they are increasingly the same. Today, every leader is a technology leader.

The inherent need for IT and the business to unite has been accelerating at breakneck speed for more than a decade, fueled by advances in technology and drastic changes in the way people work. In fact, many have dubbed this the Fourth Industrial Revolution. Unlike the Third Industrial Revolution, which used electronics and information technology to automate production, this new revolution is characterized "by a fusion of technologies that is blurring the lines between the physical, digital, and biological spheres."[7]

In fact, it's becoming increasingly clear from the "velocity, scope, and systems impact"[8] of technologies today that we're in a wholly different era. According to the World Economic Forum, "the speed of current breakthroughs has no historical precedent . . . evolving at an exponential rather than a linear pace. Moreover, it is disrupting almost every industry in every country. And the breadth and depth of these changes herald the transformation of entire systems of production, management, and governance."[9]

It should come as no surprise, then, to anyone reading this book that the technological advances of today have a significant impact on businesses. My own story of exploring serverless technologies at Liberty Mutual is but one example. Global leaders and business executives say that "the acceleration of innovation and the velocity of disruption are hard to comprehend or anticipate and that these drivers constitute a source of constant surprise, even for the best connected and most well informed."[10]

This digital revolution is

> significantly disrupt[ing] existing industry value chains . . . [and] flowing from agile, innovative competitors who, thanks to access to [global talent and] global digital platforms for research, development, marketing, sales, and distribution, can oust well-established incumbents faster than ever by improving the quality, speed, or price at which value is delivered. . . . Overall, the inexorable shift from simple digitization (the Third Industrial Revolution) to innovation based on combinations of technologies (the Fourth Industrial Revolution) is forcing companies to reexamine the way they do business."[11]

As we enter this new era of business and technology, it is irresponsible for modern organizations to ignore or waste the potential that effective technology brings to the business and the power and potential of serverless and the modern cloud—both represent software in its purest form, without hardware. Executives must learn to harness today's technology to drive innovation and power change. They must challenge their own assumptions, continuously innovate, and adapt to their changing environment.

But even with this evidence, some leaders continue to ask if technology is genuinely driving their business forward. Honestly? I imagine if you asked Jeff Bezos or Elon Musk that question, they'd probably reply, "Yes, but we could be doing better." If digital-native unicorns like Amazon and Tesla think they can do even better, what hope is there for the rest of us?

The question every business leader must ask themselves is this: Is technology really driving your business? There is a significant culture change required to truly achieve this. Just lifting and shifting into the cloud will only give you a nicer datacenter. Simply writing more code only increases your organization's liability; it doesn't guarantee you'll win in the marketplace. You won't really be benefiting from what these technologies have to offer your business unless you embrace a deeper mindset shift.

The organizations of today and tomorrow need a mechanism to accelerate the business and technology evolution. They must move away from the siloization of IT from the business, which creates an inherent lack of focus. IT departments are valuable parts of the company and have huge potential to create value. They are not costs to be squeezed. Integration of the departments—technology with the rest of the business—is fundamental to the success of the whole organization.

The organizations that recognize this will create a space for innovation. Small successes will breed larger success and spread through the organization like wildfire. This power and momentum will increase, and the path forward will become smoother. This is the Value Flywheel Effect, when the business and technology strategies power and drive each other, turning the organization into a sensemaking machine with the ability to easily pivot to the challenges of today and to whatever the next great transformation will be.

Organizations that achieve true alignment between the business and technology will find themselves riding a wave of continuous momentum thanks to the Value Flywheel Effect. To achieve continuous momentum means to be in the lead, to break new boundaries. In the mechanical world, when a power source is inconsistent, a flywheel is used to absorb energy and evenly distribute it so the machine runs smoothly. I believe that both business and technology drivers must merge in this same way to ensure smooth progress forward.

As business technology leaders, the next wave of technology will not worry about servers, instances, and traditional models—it will be serverless. When we free ourselves of the operational burden of managing infrastructure and think about capability more abstractly, the organization can move more quickly. When we let go of the constraints from yesteryear and forget about IaaS, PaaS, and FaaS,* we

* Infrastructure as a Service, Platform as a Service, Function as a Service.

accelerate. We know we need to consume capabilities and assemble systems that will drive our business forward.

The Value Flywheel Effect exists in every organization, but it will turn very slowly if you lock all your engineers in the basement and demand that they crank out code. There is a better way. I have seen it, experienced it, and now I am sharing it with all of you.

I hope you enjoy it and learn from it. Mark, Michael, and I have worked very hard to learn these lessons and distill them into this book. We are so pleased that you are taking the time read it, challenge it, and (hopefully) evolve it.

How to Read This Book

We have broken this book down based on the elements of the four phases of the Value Flywheel Effect.

- Part I focuses on an introduction to the Value Flywheel Effect, including the use of the Value Flywheel and Wardley Mapping.
- Part II focuses on Phase 1 of the Value Flywheel Effect: Clarity of Purpose and Vision. We start with a clarity of purpose or a north star, which helps measure key metrics. Time to value (as a version of lead time) is essential to capture here. To flesh out the purpose, a competitive map of the market is beneficial. What are your differentiators?
- Part III focuses on Phase 2 of the Value Flywheel Effect: Challenge and Landscape. An early assessment of psychological safety is crucial for understanding your organization's "sociotechnical" elements. An important question regarding the way of working is, "Does challenge exist in your organization, *challenge* being a healthy inquiry and debate of critical components?" Mapping the capability of your organization is essential. Do you have the people and capability to do what you need to do?
- Part IV focuses on Phase 3 of the Value Flywheel Effect: Next Best Action. At this stage, there is a purpose and situational awareness. A robust technical strategy is required to start improving your time to value. A frictionless developer experience is an excellent place to start, and we recommend a serverless-first approach. After all, code is a liability! We must start as we mean to go on, so we must create the correct mindset here. What's the next best action we can take to make progress? A popular map here is "mapping the tech stack." What's good or bad about the

tech stack? You may find that your engineers waste vast amounts of time on a flawed process or an outdated solution.

- Part V focuses on Phase 4 of the Value Flywheel Effect: Long-Term Value. As the Value Flywheel starts to turn, the longer-term value becomes essential. Well-architected systems and sustainability combine nicely to create a culture of problem prevention. Companies can use any of the three maps discussed throughout the book (market, capability, or tech stack) to troubleshoot issues, communicate, or discuss options. Mapping should be a constant and quick exercise through which we build competency.

This book contains plans, case studies, and advice for creating and accelerating your Value Flywheel Effect, helping you build confidence in the fast-changing ecosystem that is the modern cloud. We discuss twelve tenets of the Value Flywheel that provide further guidance on the four phases, depending on your specific role in the organization.

Remember, the Value Flywheel Effect is a cycle (identify a purpose, challenge existing procedures, act to improve time to value, and sustain your efforts), so you won't accomplish everything on the first pass. Once you get into a cadence of identify, challenge, act, and sustain, meaningful change will happen quickly.

This book is not just about technology or moving faster; it's about generating value for your organization. Used correctly, going serverless (using the modern cloud) will deliver more value than you have ever imagined, but you must behave like a next-generation company to unlock it. When the modern cloud is used appropriately, it will tighten up your bottom line, but the real benefit is driving the growth of the top line of your business.

Technology changes and market opportunities are coming thick and fast. Your organization needs to have the capability, agility, and cohesion to leverage the transformation opportunity. The "Great Digital Transformation" was not a one-off event. The iPhone 1 from Apple in 2007 was not the end of the mobile revolution from the previous fifteen years; it simply marked an acceleration that started the next fifteen years of evolution, and not every company could keep up.

We tend to think of technological advancements as events that have discrete starts and ends. But technology moves forward due to rapid progress and fast iterations. There is an essential question here before you start your journey: Are you happy to *consume* the event or use your Value Flywheel Effect to *create* the event? If you're one of the ones looking to *create* the event, then read on. The Value Flywheel Effect will help you get there.

CHAPTER 1
THE VALUE FLYWHEEL EFFECT

Momentum is a strange thing. It's difficult to imagine what it will feel like and takes a great deal of effort to achieve.

When we learn to ride a bicycle, for example, it feels clunky and awkward at first. It's hard to get the wheels turning in the beginning, and our frustration is often evident. But our teacher assures us that it will pass. When we finally start to build momentum, the exhilaration takes our breath away. Every push of the pedal gets easier and takes less effort. Suddenly we can focus on the larger experience of gliding through a beautiful forest or tree-lined street. The value of our hard work is evident, and we can now continue to reap the benefits with less and less toil.

However, just as the work becomes second nature, new challenges present themselves—namely, turning, shifting, and, importantly, stopping.

This cycle of nerves, uncertainty, confidence, drive, and then back to nerves is repeated by everyone, from five-year-olds learning to ride for the first time to Olympic cyclists racing in their hundredth or thousandth race. When it comes to riding a bike or any similar task, we know this cycle exists. We overlook the initial challenges of getting started because we know the value we will reap in the end. We have seen others do it, so we steel ourselves for the journey, not just the start.

It should be no different in business, especially in a world where we are constantly and with increasing speed being thrown new and more difficult challenges. And yet, too many organizations are frozen in the initial phases of nerves and uncertainty, content to remain still instead of pushing to move forward.

The Value Flywheel Effect can help organizations escape this state of stagnation and fear in the face of new challenges. For example, one of the most ubiquitous challenges organizations face today is when, if, and how to start a cloud transformation. The modern cloud (serverless) promises speed, low friction, and reduced costs, but most organizations don't realize that achieving these results requires more than just lifting and shifting their existing architecture into the cloud.

For Liberty Mutual, as was illustrated in the Introduction, it took focusing on a clarity of purpose, creating an environment for success, and experimenting with the next best action we could envision (serverless) to finally realize the long-term value that was promised. Then, using one small win (one pump of the wheel) we were able to experiment again and achieve another success (another spin of the wheel). With each success (each spin of the flywheel), the effort and work it took to earn another success was reduced. We built the momentum that eventually shot us into the forefront of technological innovation and excellence and drove huge business growth.

It can be challenging to start turning the flywheel. There are nerves and uncertainty, and it requires significant effort. Technology and product drivers are necessary, and the flywheel needs to absorb both. But as the flywheel starts to turn, it releases energy that will drive your organization forward. It achieves momentum. The systemic forces are now hard to stop but easier to guide.

Unlike many frameworks, the Value Flywheel Effect is not a "one and done" exercise. The entire idea of a *project* conjures up the image of a start and an end, point A to point Z. The fast iteration of a flywheel, however, is more akin to *Atomic Habits*, the seminal book by James Clear.[1] By breaking larger actions down into smaller efforts and moving through the four phases of the Value Flywheel, progress may seem insurmountable, but it will be faster.

Origin of the Value Flywheel Effect

The inspiration for the Value Flywheel Effect comes from three sources: Amazon's Virtuous Cycle, James Collins's flywheel concept from *Good to Great*, and from our own experiences.

In *The Everything Store: Jeff Bezos and the Age of Amazon*, journalist Brad Stone explains that the "flywheel effect" in the company's early stages worked like this:

> Lower prices led to more customer visits. More customers increased the volume of sales and attracted more commission-paying third-party sellers to the site. That allowed Amazon to get more out of fixed costs like the fulfillment centers and the servers needed to run the website. This greater efficiency then enabled it to lower prices further. Feed any part of this flywheel, they reasoned, and it should accelerate the loop.[2]

The flywheel story described is also known as the Amazon Virtuous Cycle and, as you would expect, focuses on the customer (see Figure 1.1).

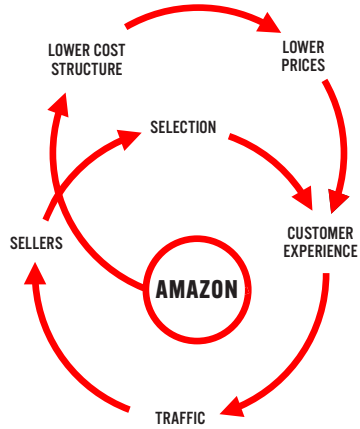

Figure 1.1: The Amazon Flywheel (also known as the Amazon Virtuous Cycle)
Source: The Everything Store: Jeff Bezos and the Age of Amazon *by Brad Stone.*

Bezos's Virtuous Cycle is itself an adaptation of Jim Collins's flywheel from the book *Good to Great* (see Figure 1.2). As Collins describes it,

> No matter how dramatic the result, good-to-great transformations never happen in one fell swoop. In building a great company or social sector enterprise, there is no single defining action, no grand program, no one killer innovation, no solitary lucky break, no miracle moment. Rather, the process resembles relentlessly pushing a giant, heavy flywheel, turn upon turn, building momentum until a point of breakthrough, and beyond.[3]

Collins's model drives organizational change and is primarily focused on leading change in the organization.

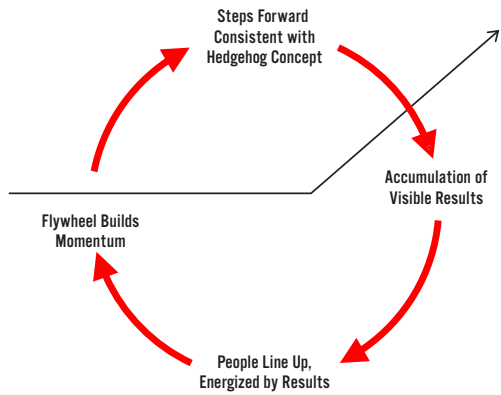

Figure 1.2: The Flywheel Concept from Jim Collins
Source: Good to Great: Why Some Companies Make the Leap and Others Don't *by Jim Collins.*

Bezos's flywheel tells us how a new business can energize the customer as a virtuous force for change. Collins's flywheel shows how to use a small, successful organizational change to foment a larger transformation within a struggling business. But how do we build momentum and continue to transform an already successful company?

The Value Flywheel Effect described in this book (see Figure 1.3) is the third iteration of the concept of the organizational flywheel, combining elements from Bezos's Virtuous Cycle and Collins's flywheel and applying much of our own technical leadership lessons learned.

This Value Flywheel Effect reveals the rapid acceleration and transformation that is possible when technology and business strategies intertwine. If the focus is only on the customer (as with Bezos) or only on the organization (as with Collins), organizations will continue to struggle to bring technology and business together. They will struggle to improve how long it takes them to realize value. In the modern digital age, technology and the business must become one. As we evolve the works of Collins and Bezos, this Value Flywheel Effect will be the catalyst for all our future endeavors.

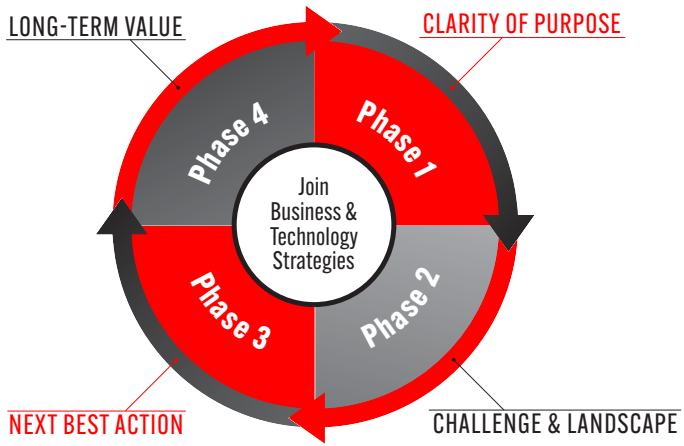

Figure 1.3: The Value Flywheel Effect

The Four Phases of the Value Flywheel

The Value Flywheel has four phases, starting with *clarity of purpose*. A clear and definitive purpose is often hard to achieve (like starting a mechanical flywheel), but it's a critical starting point. As Simon Sinek points out with his "Golden Circle" concept, we have to "start with why" (from his book of the same name). Once you have

clarity on your why (your purpose), only then can the flywheel start turning. As you saw in the introduction to this book, Liberty Mutual first had to align to three main areas of digital transformation: customer centricity, agility, and cloud-native development. This became the north star we used to direct all our future efforts.

Once a purpose is selected, this will inevitably lead to challenges (the second phase of the Value Flywheel). How are we going to achieve this? What actions do we need to take to reach our goal? Challenge is good. It helps an organization question its business-as-usual attitudes. It helps lead them out of stagnation. Only through embracing and confronting challenges can an organization hope to grow and meet the demands of an ever-changing market.

In today's landscape, solutions to challenges need to be executed quickly. Instead of waiting for the perfect solution, the goal should be to quickly identify and implement the next best action (the third phase of the Value Flywheel). There is no need to over-plan or stall. Instead, the organization must ask itself, "What's the next most useful thing we can do to provide some value?" and then act accordingly. Overcoming challenges demands speed, so this is not the time for over-engineering or deep analysis.

This next best action must have the goal of turning small wins into lasting, long-term value (the fourth phase of the Value Flywheel) to ensure that your organization creates a sustainable practice early instead of building quickly but badly, creating further complications down the line.

But the journey is not over. Once the wheel has spun, it is essential to return to clarity of purpose (the first phase of the Value Flywheel). A fast experiment should have supplied some valuable feedback so that the whole iteration (a turn of the flywheel) can start again. And again. And again. The faster your organization moves through these iterations (turns of the flywheel), the more you will learn, the more value you will deliver, and the more opportunities that will present themselves. The business strategy never stands still. There will be new ideas, new opportunities, and new developments. There will also be improvements to the technology stack and to the teams themselves. The Value Flywheel works as it absorbs changes from the business and technology (both never stop evolving).

The Value Flywheel Effect always intends to move and absorb challenges, energized through pragmatic decision-making. It's crucial to maintain momentum—to never slow down or stop. Note that the Value Flywheel Effect is neither a hybrid strategy nor operational efficiency. It's about creating a true bias for action, aligned with the pragmatic and proven ways of working that we've seen work in our own experiences.

The Value Flywheel is designed to spin many times, so don't feel that you need to do everything in phase two before moving on to phase three. Momentum and bias for action are more important than anything else. The flywheel gets its energy through feedback and ideas from both product and technology, so getting moving

is critical. The flywheel then levels out these requests and maintains a steady flow of power throughout the organization.

The Value Flywheel Effect in Action

Let's look at the Value Flywheel in action. The first phase of the Value Flywheel is *clarity of purpose* when a new requirement emerges. Can we line it up to a north star? Are the benefits of this ask clear? What is the value proposition that this ask will deliver, and is the time to value clear? A well-thought-out ask will quickly move you through this phase.

Next is the *challenge*. An ask will always require specific capabilities. Is there a safe environment to challenge and explore this ask? Can we pick at it and get behind it? Is the opportunity cost of doing this over another thing clear? Are the teams well-positioned to do this work? From a sociotechnical perspective, is our system set up for success, or could this ask be a breaking change?

Once we have confidence that the ask aligns with our capability, we move on to the third phase, finding and acting upon the *next best action*. This is one of the most critical stages of the Value Flywheel. Most organizations that are moving to serverless will eventually use public cloud providers as their platform. Cloud platforms exist to enable and provide acceleration. The smart organizations that leverage the cloud properly will see the benefit of acting quickly. If the engineers have a frictionless developer experience and execute against a serverless-first strategy, they will build well and fast.

Finally, *long-term value* is our check and balance against technical debt, which slows the flywheel down. This phase benefits from seeds planted at the start and aims to prevent longer-term issues. A problem-prevention mindset is often forgotten when we go to market quickly, and we rarely come back to clean up the mess. Facilitating and investing in well-architected and sustainable engineering/product development will ensure that we think ahead and keep the flywheel turning smoothly.

If the ask moves through these four phases smoothly, we have ensured that our flywheel will output value and is ready to turn again. The momentum generated in the organization by building this way is invaluable.

Key Tenets of the Value Flywheel

The Value Flywheel Effect, enabled by cloud adoption, will accelerate your business. Each phase of the Value Flywheel is anchored by three key tenets (twelve in total), as detailed below and in Figure 1.4. These tenets will help guide you through

each of the phases of the Value Flywheel (and the remaining Parts II through V of this book are organized according to these tenets).

That said, it's essential to understand that we are constantly evolving—these tenets may not hold in a few years. For that reason, we will also illustrate them using Wardley Maps. You should map your context and adapt these principles to work in your environment.

We've also broken these tenets down based on personas, or the role in an organization that would be most concerned with each phase of the Value Flywheel. The persona listed for each section is not the sole owner of these tenets, but the individual who would sleep easy if their three tenets were followed.

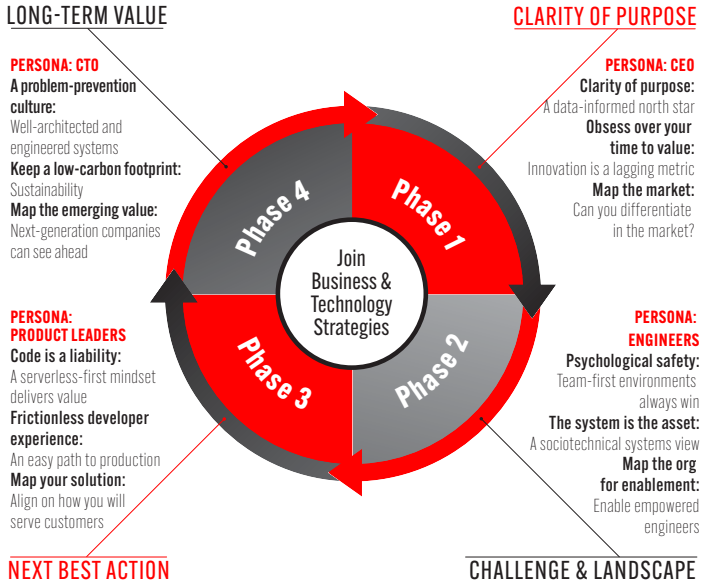

Figure 1.4: The Value Flywheel Effect & Key Tenets

Phase 1: Clarity of Purpose (Persona: CEO)

1. Clarity of purpose: A data-informed north star.
2. Obsess over your time to value: Innovation is a lagging metric.
3. Map the market: Can you differentiate in the market?

Phase 2: Challenge & Landscape (Persona: Engineers)

4. Psychological safety: Team-first environments always win.
5. The system is the asset: A sociotechnical systems view.
6. Map the org for enablement: Enable empowered engineers.

Phase 3: Next Best Action (Persona: Product Leaders)

7. Code is a liability: A serverless-first mindset delivers value.
8. Frictionless developer experience: An easy path to production.
9. Map your solution: Align on how you will serve customers.

Phase 4: Long-Term Value (Persona: CTO)

10. A problem-prevention culture: Well-architected and engineered systems.
11. Keep a low carbon footprint: Sustainability.
12. Map the emerging value: Next-generation companies can see ahead.

Let's discuss the key tenets in relation to the four phases of the Value Flywheel in more detail.

Phase 1: Clarity of Purpose

From a company perspective, the CEO is the individual we can use as the persona most concerned with the first three tenets. Though these tenets affect everyone in the organization, the CEO has the interest of the company at heart and can ensure these three tenets are met.

Clarity of purpose is the number one job of the CEO. The company must have a vision and not just a few words written on the wall. Clarity of purpose can be tested by creating a north star–model using the North Star Framework from Amplitude (more on this in Chapter 5). Ideally, the north star is a lagging metric (one that takes a long time to measure), and you should be able to identify the leading metrics (actions that lead to an outcome) and the effort that will drive its success.

Many CEOs *demand* innovation, which often leads to innovation theater and little actual innovation. If the CEO tracks time to value instead—which means reducing the time taken from "idea conception" to "value in the hands of customers"—then innovation will happen. Innovation is a lagging metric. Rather than focusing on the nebulous idea of innovation, improve the leading metrics that you can control. (We'll explore this in Chapter 6.)

Related to the clarity of purpose is the intellectual property of the organization. Is there clarity regarding the market you are operating in? Performing a Wardley Map on your value chain will help distinguish your differentiators and your enablers. (We'll explore this more in Chapter 7.)

Phase 2: Challenge & Landscape

The software engineer in an organization is focused on a different set of tenets than the CEO. The engineer's responsibility is to build well, so there are specific tenets that will help set them on the right path.

Psychological safety is critical here, as it is the foundation for an environment that fosters success. Engineering requires collaboration, challenge, vulnerability, calculated risk-taking, and skill. A highly charged political environment will negatively impact the team's success. Alternatively, a team-first environment, like in many sports, will lead to better results and engagement all around. (More on this in Chapter 9.)

Often, engineers will obsess with the code while non-engineers will consider the people. But the key contributors to any software system are the people that interact with technology. It is this combination and interrelationship between the socio (the people) and the technology that is of vital importance. If the sociotechnical system is valued and understood, then engineers can make huge impacts. If it is not, inertia will slow down your flywheel. (We'll explore more on sociotechnical systems in Chapter 10.)

The top issue for engineering teams is often friction. Decision-makers in a business often try to govern and ensure compliance by restricting teams. If we Wardley Map the engineering environment, it should be clear that certain functions are stuck in the wrong phase. This map can be a valuable source of continuous improvement that will enable instead of frustrate engineering teams. (We'll explore this map in Chapter 11.)

Phase 3: Next Best Action

There are many flavors of business or product roles, but they should all represent customer value. For the third phase of the Value Flywheel Effect, the product leader, who represents the customer, is the driver. They ask the question: How can we optimize for maximum customer value? It's important to recognize the depth of the product discipline and the many important techniques available. In *The Value Flywheel Effect*, we'll focus on speed—deciding what to build is a whole other set of books!

One of the biggest misunderstandings in the world of software is the value of code. But code is a liability, as we'll say repeatedly in this book. The more code we write, the more complexity and risk we generate for ourselves. In the modern cloud, it's important to offload as many capabilities to the provider as possible. Less code allows teams to move faster. Taking advantage of serverless is the clearest next best action for many modern organizations. (We'll explore the benefits of a serverless-first mindset more in Chapter 13.)

When teams do release new features, it's critical that there is a frictionless developer experience. Organizations must make it easy for the engineers to make changes quickly and in a safe, secure manner to deliver value for the business and keep the flywheel moving. Automation is a key enabler in reducing developer friction. (We'll explore this more in Chapter 14.)

To embrace a serverless-first mindset (offloading infrastructure management to the cloud) it's a valuable exercise to Wardley Map the existing technology stack with engineers. With this map, it will quickly become clear which components either slow the team down, generate little value, or are easily replaced by a cloud service. (We'll explore this mapping technique more in Chapter 15.)

Phase 4: Long-Term Value

The final persona driving the Value Flywheel Effect is the CTO (chief technology officer or similar), who represents the architecture of the system. Often misunderstood, the architecture of the system should support future changes, reduce risk, and meet the business need. Like security, good architecture often results in bad things not happening, which is often difficult to measure. And good architecture leads to sustainable, long-term value versus short-term gains.

Many organizations reward teams for fixing problems. An alternative model should be to create a culture of *preventing* problems: reward the teams that use well-architected and strong engineering practices to prevent issues from *ever* occurring and lead to more reliable systems in the long term. (We'll explore problem-prevention culture in Chapter 17.)

Good architecture is often hard to define and measure; therefore, efficiency can be a strong measure here. And efficiency can also be represented as sustainability. Cloud providers are starting to measure the amount of carbon burned in a specific workload or system. Quite simply, if a team can reduce their carbon burn, they are providing a benefit to the customer, the company, and the environment. (We'll explore this more in Chapter 18.)

A key role of architecture is looking ahead and anticipating change. One thing in technology that is certain is there will be an evolution of capability. Wardley Mapping provides the perfect mechanism to map how key capabilities in your value chain will evolve and what emerging capabilities or needs will surface in the future. Once mapped, you start preparing for evolution today instead of waiting for the future to hit you in the face. (We'll explore this map more in Chapter 19.)

Avoiding Inertia with Your Value Flywheel

There's a very good reason why the mechanism we present throughout this book is a flywheel: to succeed, inertia must be avoided at all costs. Inertia is the resistance of matter to change. This includes changes to speed and direction. Organizations today cannot afford to suffer this inertia. They must have the ability to increase their speed (improve time to value) and change direction (adapt).

With sequential change, there is always inertia that will either slow or misdirect effort. It's very hard to avoid. The Value Flywheel Effect provides a tight feedback loop, making it possible to smash inertia before it becomes an issue. The inertia of a flywheel opposes and moderates fluctuations in the speed of an engine, or in this case an organization, and then stores that excess energy for intermittent use. An organization can draw upon this stored energy in times of great change without having to exert more effort.

Change is difficult, but the Value Flywheel Effect will keep things moving. Sometimes a simple motto like "code is a liability" or "improve time to value" will keep engineers moving more effectively than a fifty-five-minute town hall with the same presentation that was delivered in some boring meeting last week.

At an executive level, there is often the need to gradually cascade strategy. You do not need to do this here. Once the high-level strategy is in place, you can let the teams advance at their own pace. To prevent inertia from slowing down your flywheel, give your engineers permission to move quickly. The flywheel nearly removes the need for executive oversight. In fact, command and control is the biggest creator of inertia in this approach.

With cohesive feedback loops, progress is transparent. Transparency can help leadership craft the right language and drive the Value Flywheel Effect. The more your teams understand this feedback loop—what worked and what didn't work—the quicker your progress will be.

Takeaways

The concept of the flywheel is not new; it has been used to describe company change and customer interaction for decades. Now we're showing organizations how to use the Value Flywheel Effect to inform impactful strategy as the business and technology combine. With the current technology picture, this flywheel has never been so central or needed to turn so fast—and it's not slowing down. It's time to find that flywheel in your organization and recognize the four phases: Clarity of Purpose, Challenge and Landscape, Next Best Action, and Long-Term Value.

To help you on your journey, we've provided twelve key tenets based on personas in your organization to help you see areas of improvement and build situational awareness.

Some of the traditional models for creating and analyzing strategies may not work in today's fast-paced environment. The Value Flywheel Effect helps to seamlessly distribute power throughout the organization, turning technology and product needs into momentum and value.

But power without direction will leave any organization spinning in circles. Wardley Mapping—creating a visual representation of who you serve, what they need, and how you fulfill that need—helps the organization navigate the rough waters ahead and find direction. In the next chapters, we'll explore the method of Wardley Mapping in more detail to help you practice it in your own organization and build necessary situational awareness.