build better systems faster

INDUSTRIAL DEVOPS

Dr. Suzette Johnson
and **Robin Yeman**

Forewords by **Mik Kersten** and **Dean Leffingwell**

IT Revolution
Independent Publisher Since 2013
Portland, Oregon

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

First Edition
Printed in the United States of America
28 27 26 25 24 23 1 2 3 4 5 6 7 8 9 10

Cover design by Devon Smith Creative
Book design by Devon Smith Creative

Library of Congress Control Number: 2023027080

ISBN: 9781950508792
eBook ISBN: 9781950508808
Web PDF ISBN: 9781950508815
Audio: 9781950508822

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

INDUSTRIAL DEVOPS

CONTENTS

FIGURES & TABLES

Chapter 9

Chapter 10

Chapter 11

Chapter 12

Chapter 13

Appendix C

Foreword *by Mik Kersten*

With the acceleration of software and the advent of generative AI, we are witnessing the fastest pace of change in our careers. This puts tremendous pressure on both organizations and individuals to learn and adapt in a rapidly shifting technology landscape.

Since the 2000s, we have seen digital natives create the delivery and feedback loops that have enabled them to capture a massive portion of the global market. Since then, much has been written about digital transformations, and an abundance of guidance provided for how large enterprises should go about becoming software innovators. However, comparatively little has been written about how organizations building physical systems should transform. That's where Industrial DevOps comes in.

A small number of organizations that take a software-centric approach to building physical systems have won a disproportionate part of the market for physical products. Apple and Tesla alone have disrupted countless slower moving incumbents and amassed a dizzying amount of market share in the process. What we need for a thriving economic landscape is for the rest of the world's organizations to match this pace of product delivery and innovation.

What *Industrial DevOps* does so eloquently is make the case for why this shift is not only possible but critical for your organization to thrive. In addition to providing concrete guidance, this book has a wealth of case studies that will show you what good looks like regardless of your industry or scale. The Joby Aviation case study demonstrates how these principles apply to newer and nimbler companies wanting to move fast against entrenched competitors. At the other end of the spectrum, the Lockheed Martin U-2 Dragon Lady case study demonstrates how the same principles can be applied to bring machine learning and DevOps via Kubernetes containers to a reconnaissance aircraft created back in the 1950s. *Industrial DevOps* will provide you and your organization with numerous examples that ground the principles presented in guidance that you can start actioning today.

There is no one I know better suited to put forth these best practices and methodologies than Robin and Suzette. Their close involvement with the IT Revolution DevOps Enterprise community for the past decade has given them a perspective that few others have. More importantly, thanks to their groundbreaking work applying the principles of Agile and DevOps to cyber-physical systems, they have continually expanded our communities' understanding of how to bring these benefits across horizons that span physical systems and the longer time horizons necessitated by hardware.

Suzette and Robin have collected that wisdom in this book, and they demonstrate how the principles of DevOps can not only move mountains within your organizations but can go as far as moving interstellar objects. Quite literally, as one of the case studies in the book is NASA's DART mission, which applied continuous delivery (CD) automation to hardware-in-the-loop (HIL) and software-in-the-loop (SIL) systems to successfully change the trajectory of an asteroid. This was the first time that humanity had changed the motion of an interstellar object. If changing the direction of your organization seems like a harder task than changing the direction of interstellar objects, Robin and Suzette provide an invaluable summary on how to surmount organizational objections and make the economic case to leadership.

I have had the pleasure of working with manufacturing organizations that have participated in the DevOps Enterprise community early and were some of the first adopters of the principles outlined in this book. The journey of one of those organizations, BMW Group, is chronicled in *Project to Product*, where I recount my visit to the Plant Leipzig.

I happened to be reading the manuscript for *Industrial DevOps* while making a visit to BMW Group's Munich plant, hosted by Ralf Waltram (VP IT Delivery DevOps for Production at BMW Group). Ralf helped me come up with the title of *Project to Product* as he was leading the BMW Group's shift to Agile. He made this anniversary tour even more intimate, showing me an amazing combination of digital workstations combined with active car assembly via human and robot collaboration.

I could not believe the progress that I saw since visiting the Plant Leipzig exactly six years earlier. The manufacturing plant was producing electric, hybrid, and combustion engines all on a single line, just in sequence (JIS) of orders. I had not fathomed that could be possible given the extremely different physical components and calibrations required for one body that housed a massive battery being followed by another that contained a combustion engine.

As Ralf pointed out, managing this kind of real-time complexity would simply not be possible without a software-centric approach based on digital twins, simulation, and platform architectures. It will be exactly these and other leading-edge practices of product development innovators that you will learn about in this book.

Each technological revolution builds on the pioneering work of the organizations that paved the previous one. We have entered the Age of AI, which I believe will provide productivity gains of at least an order of magnitude for software development and architecture. Organizations that take a software-centric approach to their product development will be able to quickly reap those massive productivity gains and to deliver value to their customers or citizens at an unprecedented rate. Those restricted by the constraints of waterfall development will not have the infrastructure for innovation needed to leverage the benefits of these tectonic shifts.

Industrial DevOps is your guide for the changes that you need to put in place to reap the benefits of digitization and AI. The guidance in the book ranges from architecting around flow and value streams to the simulation and automation needed to 10x your hardware and software feedback loops. I firmly believe that the organizations adopting the practices in this book will outperform and outlast those that do not lean into the change and learn to move at the speed of DevOps. I hope you use this book to build the road map for your team's journey. Whether that leads you to the stars or to your organization's north star, you are about to embark on it with two of the best and most proven guides in the industry: Robin and Suzette.

Enjoy!

—Mik Kersten
CTO of Planview and author of *Project to Product*

Foreword *by Dean Leffingwell*

Sometimes major breakthroughs in technology development happen quietly. They might not initially make headlines, but their impact on how we innovate is indelible. Agile development is a great example.

It's hard to believe it's been over twenty years since Agile, and later DevOps, emerged as a better way of building software and systems. Thankfully, I can now say that I've spent more time working in Agile than in my prior years applying waterfall. But I must admit, it took me a couple of years to join the Agile party.

My earlier developer and executive years were spent primarily developing high-assurance systems such as medical devices, DNA sequencers, heavy-duty industrial robotics, patient monitoring systems, aerospace systems, and the like. The development of those systems gave me great respect for documented requirements and design, traceability, meeting imposed regulations, and all the other aspects of delivering systems that did exactly what they were supposed to do and no more.

Agile values and principles didn't address those issues; initially, I wasn't sure they would be useful for that kind of work. But I then had the opportunity to coach several pure software startups. Free of all those high assurance constraints, I learned and coached Agile with several new companies. One was Rally Software, where I was mentored and influenced by Ryan Martens, an early and influential Agile proponent.

Simply put, I was blown away by the power of Agile. It was unlike the methods that came before. It was a quantum leap forward. What's more, once I experienced being on an Agile team, I vowed never again to work in a situation where I couldn't be on one. That led me to a new purpose: bringing the power of Agile to those building the world's largest and most important systems. This new passion led me to write *Scaling Software Agility: Best Practices for Large Enterprises* and *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*.

Even then, the application of Agile—and scaled Agile—was primarily constrained to software systems. The challenge of building high-assurance and cyber-physical systems with Agile never left my mind. I continued to research and observe how scaled Agile methods could be applied to these systems. And when it became apparent that Agile development methods produced natively and fundamentally higher quality outcomes, I reached another tipping point.

Working with Dr. Harry Koehnemann and other industry experts, we branched the Scaled Agile Framework® (SAFe®) into a beta version of SAFe for Lean Systems Engineering. But as early testers soon discovered, it became evident that the principles and practices of building these systems were strikingly similar to large-scale software development. With that knowledge, we instead folded those learnings into the new SAFe Enterprise Solution Development competency.

Two of the industry experts who helped guide us through that exploration journey are the authors of this book, Robin Yeman and Dr. Suzette Johnson. I found their perspectives and experiences to be invaluable to my new purpose. In their respective roles at Lockheed Martin and Northrop Grumman, they had been working—both cooperatively and competitively—to extend Agile, DevOps, Lean, and SAFe to build everything from satellites to submarines, as Robin likes to say.

I have always been impressed by their experience, insights, grace, and gumption. You can imagine it was no easy journey. They worked for very large, very conservative, and very successful companies, who, in turn, delivered systems to the world's largest bureaucracy, the US government. Undoubtedly, they found significant impediments blocking their progress every day. And they were not just working the "easier" pure software stuff; they were applying these innovative and advanced methods to very large-scale and safety-critical cyber-physical systems. These systems contained combinations of software, mechanical, firmware, electrical, electronic, hydraulic, propulsion, optical, sensors, actuators, and physical devices. Moreover, these systems interacted with other equally complex systems in extensive communication networks. Tough stuff, to put it mildly.

Meanwhile, the DevOps movement was gaining momentum. I vividly remember the fourth annual DevOps Enterprise Forum in Portland, Oregon, in 2018. This private event was sponsored by Gene Kim and IT Revolution. Invited experts met and collaborated in what Gene described as a "scenius" (communal genius). Teams collaborated, wrote, and published (mantra: "in DevOps, we ship") guidance papers on the most challenging and relevant topics to the industry.

I joined a working group instigated by Robin and Suzette that was keen to work on this topic. Initially, the task was daunting. How could anyone describe the application of Lean, Agile, DevOps, and Systems Thinking bodies of knowledge to building systems that were so diverse and complex that the systems themselves were hard to describe in any detail, much less document and validate with any rigor? The conundrums were many: continuous integration meets the laws of physics; fast iterations meet the laws of overloaded machine shops; fuzzy user stories meet the laws of documented software requirements specifications.

But a breakthrough came when the working group agreed that while the problems and experiences were vastly different, and comprehensive guidance might be unattainable, there was another way to move the industry forward via the application of common principles. This W. Edwards Deming quote inspired us: “[The problems] are different, to be sure, but the principles that will help to improve the quality of product and service are universal in nature.” That’s when we knew we could create something worthwhile. We’d take a stand on agreeing to a set of principles—underlying truths that could reliably inform the methods and practices needed to build these diverse and complicated systems.

That effort resulted in the first published paper on the topic entitled *Industrial DevOps: Applying DevOps and Continuous Delivery to Significant Cyber-Physical Systems*. Further papers ensued, and Robin and Suzette wrote this book as a logical and much-needed outcome. Part II of this book highlights the evolved principles which form the core of Industrial DevOps.

But perhaps one of the most valuable aspects of this book is that it is not just about abstract principles. The ideas described here are augmented by real-world and exemplary case studies, practical tips, and guidance. I can say for certain that applying Lean, Agile, DevOps, and Systems Thinking to the creation of the world’s most important systems is now more attainable. Doubters and naysayers may well abound in industry and government, and perhaps some will never give up their more prescriptive and stage-gated waterfall habits. But those with an open mind can now see the possibilities of building these systems faster, with higher quality and lower cost, and, importantly, enhancing the joy in doing.

And what could be more important than applying these principles to the very systems that provide society with security and safety? Surely that would truly make the world a better and safer place. We will soon owe some of that to Robin and Suzette.

This is an important work; as an industry and a society dependent on these future systems, we are lucky to have it.

—Dean Leffingwell
Creator of SAFe®

Preface

Suzette: My first experiences with Agile, specifically Scrum and eXtreme programming, started in 2005. The mission called for faster delivery of software capabilities. We had to find a way to manage the fast pace of change, and the practices uncovered from the Agile community seemed like the right fit for our environment. This was a large-scale implementation of many teams of Agile teams who were very successful at delivering at speed of mission. My Agile experiences had always been at scale.

My journey to Industrial DevOps started around 2015, when I was asked to help a software development team of about sixty people implement Agile and DevOps. The project was a special radar program. Its unique capabilities were designed to help the defense community provide special data and reporting for critical decision-making activities. However, these large radar systems require more than software. The software must run on hardware. And therein was the challenge.

Before I came on board, the project had been using a waterfall approach to development. The project was "in the red," and not only was it running behind schedule, but the slip in schedule was also getting a great deal of attention from customers and management. Concerns were mounting. The risk to try something new was low, and management's mindset was, "Well, we've tried everything else—we may as well try this Agile/DevOps thing." So, we did.

Over the course of a year or so, things turned around. The software team did an outstanding job of adopting the new practices, and the project (from a software perspective) was back on track. But, while the software was reporting green, the project as a whole was still reporting crimson red. How could this be? This is when I realized the misstep that had been made. Agile/DevOps principles and practices had not been applied to the whole value stream. We had neglected to bring the hardware team, and a couple of other teams further down the value stream, along with us as we adopted these new ways of working.

Around this time, I had also been reading the book *The Goal* by Eliyahu Goldratt. *The Goal* is a novel about a Lean manufacturing plant. It teaches concepts such as improving flow, productivity, and continuous improvement on the manufacturing line. The analogy that could be drawn between the learning from Goldratt and what I just experienced was serendipitous. It truly was a moment of epiphany.

Much of the effort I had spent with the team was focused on improving the flow of software development and removing bottlenecks at the *software* integration points. The software teams had shifted to an Agile and DevOps approach while the rest of the functions were still using waterfall practices. While the results from the software were good, the overall effort was not. And that is when I thought back to what I had learned from Goldratt: if you build in all your efficiencies before the bottleneck in the assembly line and you still can't deliver to your customer, then your efficiencies are likely an illusion. The bigger bottlenecks were *between* software and hardware.

The real goal was to deliver the product to the customer, not to improve software development. We had neglected to assess the whole picture. What was needed was a systems view of the problem and to understand how all the teams were organized around the flow of value. By understanding and visualizing the full value stream, we were able to prioritize improvement efforts and focus on the highest-prioritized needs, wherever they fell within the value stream.

In the summer of 2017, I began volunteering at an Agile conference in San Diego. I already had experience with scaling Agile in cyber-physical systems, and I wanted to learn more. I was curious if anyone else was having similar experiences. That led me to meet Joe Justice.

He had been very successful with Wikispeed, an automotive manufacturer of modular cars using Scrum and eXtreme programming practices as applied to hardware. He invited me to a Scrum for Hardware workshop he was giving at Bosch Engineering, just outside of Stuttgart, Germany, later that fall. It was an amazing experience. Bosch was implementing Scrum in their hardware development, and they were gracious enough to include me in their train-the-trainer course and share their approach and steps toward scaling across larger parts of the organization. My assumptions were validated in the art of the possible.

Robin: My experience with Industrial DevOps concepts has been an evolution over many years, beginning with the transition to Agile in 2002. Every system that we build and maintain in the aerospace community is large-scale, cyber-physical, and often safety critical. I found time and time again I would have teams doing an excellent job completing the software rapidly by using Agile practices, only to find that the software would be shelved, waiting for the hardware to be completed. By the time the hardware had been completed, we found that the software needed to be adapted based on the actual hardware and the changes in the mission environment. Bottom line: we were not looking at the whole value stream and addressing the theory of constraints. Any improvement we made in the process was not resulting in changes in the outcome for our customers. This problem was replicated in every environment I supported, from radar to ships to aircraft.

In 2013, I began researching a variety of materials on implementing Agile for hardware from several pioneers in the field, including Joe Justice with team Wikispeed and Gary Gruver with Hewlett Packard.

In 2014, I implemented our first Agile for Hardware pilot building a missile with an existing legacy program. We experienced many benefits using Agile ways of working, including reduced rework, shorter lead times, and increased transparency. At Lockheed Martin, we built an Agile for Hardware workshop and training materials, which allowed us to continue to experiment with Agile practices on hardware projects across multiple domains.

While not every experiment was successful, it became clear that we needed to expand Agile beyond hardware and software and to the entire value stream if we were going to maximize benefits to the end customer.

Over the course of the next few years I was able to identify key lessons learned, which included:

- **Agile is an empirical life cycle:** it needs to include all of product development to succeed.
- **Focus on the principles:** Agile/DevOps principles apply to all types of work.
- **Context matters:** Agile/DevOps utilizes many patterns; not all of them are applied to every area.
- **Begin with the problem** you are trying to solve, not a process.

Suzette and Robin: We soon had the opportunity to discuss what we were seeing at the fourth annual DevOps Enterprise Forum in Portland, Oregon, in the spring of 2018. At this private event, Gene Kim (best-selling author, researcher, and multiple-award-winning CTO) invites industry leaders and experts to come together for a couple of days to discuss the issues at the forefront of the DevOps Enterprise community and to put together guidance to help us overcome and move through those obstacles.

So, we went and made our pitch to the group:

The Big Idea: DevOps for large, complex systems utilizing both hardware and software.

Topic: Shifting large legacy organizations with ingrained traditional management and engineering practices into a DevOps flow of continuous delivery of value in large, complex systems with both hardware and software development.

Problem Statement: We recognize the benefits of DevOps in IT/software-centric environments. Can these same benefits be recognized in the development and engineering of large, complex systems that encompass both hardware and software development (like an autonomous car)?

Following our pitch, a team of leaders* from across the industry joined us to address this problem space. For the next several months, we worked on defining and publishing our first paper on the subject: *Industrial DevOps: Applying DevOps and Continuous Delivery to Significant Cyber-Physical Systems*.[1]

In the development of this paper, we agreed upon several things:

1. These software/hardware systems would be referred to as "cyber-physical systems."
2. We identified the success patterns we were seeing in the application of Lean, Agile, DevOps, and trends in digital engineering in software and hardware.
3. We pulled from existing bodies of knowledge, including Lean, Lean Startup, systems thinking, design thinking, systems engineering, model-based systems engineering, Agile, and DevOps.
4. We named the integration of cyber-physical systems and the success patterns "Industrial DevOps." *Industrial* pulled in the concept of hardware and manufacturing, while *DevOps* emphasized flow and fast feedback from software.

* The initial team included Diane LaFortune, Dean Leffingwell, Harry Koehnemann, Dr. Stephen Magill, Dr. Steve Mayner, Avigail Ofer, Anders Wallgren, and Robert Stroud.

Over five years, we researched and implemented our ideas, producing five papers as we refined the concept of Industrial DevOps. Finally, it was clear that it was time to collect our experiences and the refined concept into a book to help instruct others on how to build cyber-physical systems using Industrial DevOps principles. While this book is the culmination of years of experience, experimentation, failures, and successes, we recognize the journey is not over. As we broaden our experiences and learning, sometimes it feels like it is just the beginning. We expect the concepts presented here will continue to evolve over the coming years, and we firmly believe these principles will help businesses not just survive the digital age but thrive in it.

Introduction

It is no secret that we are living in the digital age. Technological advancements continue to grow at an unprecedented rate. Digital capabilities are impacting our businesses, making competition fiercer and increasing the need to expedite time to market. To succeed today, organizations must rapidly improve, innovate, and adapt to changing market demands or lose their foothold in the market.

A key imperative for companies to be successful is *speed to market*. And one of the clearest ways to achieve this is by updating our old ways of working (waterfall) with more Agile ways of working. In the digital world, countless organizations have achieved this speed through Agile and DevOps practices. But Agile and DevOps ways of working aren't just for software companies or digital natives. Today, these ways of working are crucial for all businesses, including those that create cyber-physical systems (combining software with hardware and firmware). And yet, few of these companies have successfully adopted these ways of working, putting the majority in the dangerous position of falling behind their competition.

The Need for Speed

In the 16th Annual *State of Agile Report*, which surveyed more than three thousand people from across industries and countries, 52% of respondents reported that accelerating time to market was a key benefit of transitioning to Agile ways of working. This benefit was coupled with organizations' desire to move quickly while still remaining predictable. In addition, 47% of respondents reported that Agile teams are measured by the speed of delivery, which highlights organizations' continued desire to deliver in shorter lead times.[1]

Over the past decade, Agile and DevOps have produced dramatic changes in the way software teams develop and deploy value to their customers. According to a 2019 *Harvard Business Review* report, organizations that have implemented Agile and DevOps practices demonstrated "increased speed to market (named by 70%), productivity (67%), customer relevance (67%), innovation (66%), and product/service quality (64%)."[2] Data from a 2020 report from the Standish Group shows that "Agile Projects are 3X more likely to succeed than waterfall projects. And waterfall projects are 2X more likely to fail."[3] In addition, their data indicates that "large agile projects succeed at twice the rate of non-agile projects and fail half as often."[4]

Clearly, the ability to react to market demands and deliver value with speed has become a key differentiator in digital products and companies. But it does not end there. This need for speed is not limited to the "digital" world. Nicola Accialini describes in his book *Agile Manufacturing: Strategies for Adaptive, Resilient and Sustainable Manufacturing* that there is a growing need for flexibility and speed on the production line and the ability to respond to changing markets and increasingly customized offerings, in ever shorter cycle times.[5] We recognize the growing imperative to respond to changing priorities as we build *cyber-physical systems*—that is, those that combine software, firmware, hardware, and manufacturing.

Applying Agile and DevOps ways of working in the development of cyber-physical systems through manufacturing presents an opportunity to reap significant rewards, as these systems are often large, complex, and worth millions to billions of dollars. Companies that embrace these practices can achieve speed to market and can experience increased adaptability, shorter delivery schedules, reduced development costs, increased quality, and higher transparency into delivery.

This need for speed in cyber-physical systems goes beyond the commercial industry. In the 2018 National Defense Strategy, US Secretary of Defense James Mattis called out the need to accelerate the delivery of weapons systems capabilities in response to ongoing threats and the need to ensure the continuation of national security.[6] Specifically, he highlighted "speed of relevance," as it was expressed that "the department will transition to a culture of performance and affordability that operates at the speed of relevance."[7]

Advancements in technologies and capabilities also continue to evolve quickly in the growing space market. According to Benchmark International, in 2021, the space market "was valued at $388.50 billion and is expected to reach $540.75 billion by 2026."[8] Reduced costs of developing capabilities and services with faster deployment are broadening the playing field and increasing competition. Those who don't keep up will quickly be left behind.

Left Behind

The benefits of adopting Agile ways of working are already being realized, and some organizations are anxious to reap these benefits. Even so, according to those responding to the *Competitive Advantage through DevOps* survey conducted by *Harvard Business Review*, only 10% said their organization is very successful at rapid development and deployment of software.[9] This means 90% of the market is at risk of losing out to their competition if they do not evolve and improve more quickly.

The need to change is paramount, as explained by the work of Carlotta Perez, who has been researching the different waves of technological revolutions throughout history. Her research explains how each technological wave has redefined the means of production fundamentally, which triggers an explosion of new businesses, followed by the mass extinction of those that thrived in the previous technology wave but failed to adapt.

Perez suggests that we are in the middle of the fifth technological revolution, known as the "Age of Information and Telecommunications" or the "digital age," characterized by a shift toward more decentralized and flexible production, a greater emphasis on knowledge-based services and the creative economy, and a growing concern for sustainability and social inclusion.[10]

Currently, digital environments and tools are making product development less expensive by enabling us to move physical components into the digital space and allowing product teams to perform multiple, iterative redesigns cheaply before we must acquire the needed physical components. This results in faster learning and savings in cost and schedule by finding defects earlier in the life cycle and reducing rework. Digital engineering, as it is collectively referred to, is providing the opportunity to shorten feedback loops in physical development to the speed of software through models, simulators, emulators, digital twins, and additive manufacturing.

While the new technological revolution brings clear advantages, Perez's work also shows the danger in not transforming. Based on her research, when companies continue to apply infrastructure concepts from the previous technological revolution to the current wave, they inevitably fall behind and fail, becoming casualties of each wave's mass-extinction event.

The government sector is experiencing these same challenges. The world's largest employer, the US Department of Defense, acquires cyber-physical systems in the form of strategic defense and weapons capabilities costing billions of dollars. According to a US Government Accountability Office (GAO) report, over 60% of large weapons systems are not meeting schedule and are over budget.[11] Google's CEO, Eric Schmidt, said the US government's dithering has left the country well behind China in the race to build out 5G technology.[12] China is also threatening US superiority in space. Officials of the Space Force, Defense Innovation Unit, and Air Force Research Laboratory state the United States must act quickly to maintain its advantage over Beijing, which includes using more commercial technology.[13] Based on these trends, it is imperative for the defense community to improve the rate at which they adopt Agile and DevOps ways of working so they can deliver faster than their competitors.

Consumer expectations also continue to grow, forcing both existing and new businesses to be relentless with innovation. According to Eric Ries, author of *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesse*s, the only way to win is to learn faster than anyone else.[14] This includes understanding what the customer wants and delivering it faster than your competition, delivering at the speed of need.

Forbes stated that many companies have failed to meet customer expectations and are struggling to keep up with the pace of the digital age. Since 2000, approximately 52% of the companies on the Fortune 500 list have become obsolete.[15] In addition, Gallup, a global analytics firm, has written many articles regarding "quiet quitting," a trend of increasingly disengaged employees leaving the workforce at alarming rates.[16]

This creates a complex challenge: the need for innovation, the need to be able to respond and keep up with the pace of change, and the need for knowledge workers are all growing. The increasing challenge to stay relevant and meet the growing needs of the knowledge-based workforce requires companies to make culture center stage to attract and retain talent. Knowledge workers are problem solvers who have built their craft through education and hands-on experience. They seek a high level of autonomy and creativity to stay engaged, which requires a change in not only how we manage and build cyber-physical systems but how we inspire the talent that is building those systems. Moving toward Agile/DevOps ways of working helps provide an ideal working environment for knowledge workers.

Given the economic and national imperatives we face, both industry and government need to reimagine how to build, manufacture, and sustain cyber-physical systems. For one, system complexity has grown, and uncertainty is the new normal in product development. Second, the digital revolution has dramatically raised customer expectations. And finally, the workforce to power the digital economy is different from the previous one and requires new ways of working to be successful.

As organizations face these challenges, they must inspect and adapt to the world around them and continuously improve their operating model.

The software industry demonstrated its ability to respond to changing market demands and meet business outcomes using Agile/DevOps. Now it is time for the world's largest enterprises to consider how they will scale these practices across large, complex systems. There is an urgent need to iterate and deploy faster, adapt to changing needs, reduce cycle time for delivery, increase value for money, improve transparency, and leverage innovations.

According to a summary provided by Anthony Mersino of Vitality Chicago, the 2021 Standish Group *CHAOS Report* states that data from the industry continues to demonstrate that Agile projects are three times as likely to succeed as projects run with traditional project management. And waterfall projects are twice as likely to fail.[17]

What is even more interesting is when you correlate Agile adoption with the size of the project. The results of the report conclude that "large agile projects succeed at twice the rate of non-agile projects and fail half as often. 'Medium-agile' projects do not fare that much better (31% versus only 19% for non-agile projects). Only in the small category does non-agile come close to agile."[18]

PROJECT SUCCESS RATES

METHOD	SUCCESSFUL	CHALLENGED	FAILED
Agile	42%	47%	11%
Waterfall	13%	59%	28%

Figure 0.1: Agile vs. Waterfall

Source: Anthony Mersino, "Why Agile Is Better than Waterfall (Based on Standish Group CHAOS Report 2020)."

With cyber-physical systems, manufacturing brings along considerations such as designing for manufacturability, parts management, and supply-chain needs and constraints that were less pervasive in software-only systems. Leveraging tools such as digital twins supports our ability to use telemetry from the factory to optimize product development and the overall flow of value to stakeholders. Similar to product development, the practices and emergence of digital capabilities in manufacturing are also evolving rapidly. Today, as the landscape continues to shift, factories are discovering the need for adaptability *in addition to* improving flow, resulting in increased factory output, better workforce utilization, operational flexibility, reduced production costs, and increased customer satisfaction.

Bridging the Gap

Given the clear advantages of adopting Agile and DevOps ways of working in cyber-physical systems, what is preventing organizations from adopting these ways of working? After all, several companies have already shown what can be achieved with these practices. How can we bridge the gap to show that what works for software and for manufacturing can work for systems that combine the two?

The automotive industry, for instance, has been forging ahead with Lean, Agile, DevOps, and digital capabilities for years, and their experiences can be a great example of what is possible in cyber-physical systems. Many Agile principles, like having a startup mentality and the release of a minimum viable product (MVP) for rapid feedback, have been critical to the success of Porsche since the 1930s. According to Porsche AG, "In order to meet the modern requirements of volatility, uncertainty, complexity, and ambiguity, companies must become more agile (i.e., more flexible, dynamic, and interconnected)."[19] In 2021, Porsche shared how they are scaling Agile across both digital and hardware entities: "Bringing together the two worlds of physical and digital is one of our must-win battles: hardware AND software or digital products and services. Moreover, we shape the transformation with the goal of forming an attractive organization with shared values."[20]

Of course, Porsche is not alone. BMW also has demonstrated success with Agile. In a 2018 article, they highlighted specific improvements they have experienced along their journey to improve their software practices and improve delivery times.[21] The approach at BMW addressed mindset and culture, tools, and practices. As a result, their adoption:

> . . . led to improvements in the way BMW handles the development of products and has helped significantly reduce the time to market—perhaps best demonstrated in the company's new approach to interior design. Historically, Waltram's small in-house R&D team would work using physical mockups of car interiors only twice a year. Under the new approach, his team is now using software to visualize mockups in a 3D game engine, allowing for new integrations every week.[22]

In *Project to Product: How to Survive and Thrive in the Age of Digital Disruption with the Flow Framework*, author Mik Kersten highlights the BMW Group Leipzig and the actions they took to improve flow across their value stream to meet business outcomes. Not only did they address improvements from a software perspective, but they also looked more broadly at how to improve flexibility and adaptability on the production line. Their improvements created the ability to respond to increased demands more easily in the production line by adding more automation or parallelization to the given line to increase the volume of output.[23]

No discussion on Agile and DevOps in the automotive industry can happen without including Tesla. Tesla has been taking Agile adoption into the hardware space using practices often referred to as eXtreme manufacturing.[24]

For example, Tesla used these foundational principles (in particular, optimizing for change) in the construction of their factory in Shanghai in 2019, when they met their aggressive schedule and surpassed industry expectations.[25] Tesla Shanghai used digital technology to improve operational efficiencies, such as the Giga Press, a large-scale casting machine, which can create the "entire front and rear segments of the car."[26] This technological advancement "saves 300 robots, a thousand parts and 30% of the factory space."[27]

But Tesla is not done improving. There are efforts to continue optimizing the layout and operations of the factory. With ongoing investment in new technological capabilities, there is the potential to increase the speed by at least ten times and up to one hundred times. When this is achieved, it will "enable radical factory and production speed improvements."[28] These are just a few of the examples from industry on how Lean, Agile, DevOps, and digital capabilities yield positive results.

Toward a Practice of Industrial DevOps

Speed, flexibility, and adaptability are an imperative across the value stream. When we couple the results of Agile and DevOps implementation in software development with Lean and Agile in manufacturing, we have the foundational success patterns for the development, manufacturing, and deployment of cyber-physical systems in the modern age. The benefits that have been obtained across industries *can* be transferred to the cyber-physical domain, and they have the potential to provide an even greater impact on the delivery of products. This can be achieved through the application of what we have defined as *Industrial DevOps*: a set of proven principles and success patterns for building better systems faster to achieve business outcomes.

In fact, over the years, we have been exploring how to adjust Agile, Lean, and DevOps ways of working to specifically address the concerns and unique challenges of cyber-physical systems. With our over fifty years of combined experience in software and systems engineering of large, complex systems, from satellites, submarines, and spacecraft to large communications and data systems, we have witnessed the evolution of engineering and management practices. We have seen what works and what does not work. Based on these experiences and our collaboration with multiple industry working groups, we have further defined this successful application of Agile, Lean, and DevOps principles in the cyber-physical realm as *Industrial DevOps*.

Industrial DevOps is "the application of continuous delivery and DevOps principles to the development, manufacturing, deployment, and serviceability of significant cyber-physical systems to enable these programs to be more responsive to changing needs while reducing lead times."[29] The nine principles of Industrial DevOps encompass a holistic perspective, applied across the organization, and include the culture and mindsets of an organization leading to the delivery of value to the end customer.

Industrial DevOps is the advancement of Agile and DevOps practices specifically in complex, regulated cyber-physical environments. These environments contain significant challenges, considerations, and success patterns. We have learned from our decades of experience in this industry that the words of W. Edwards Deming still hold true:

> A common disease that afflicts management and government administration the world over is the impression that, 'Our problems are different.' They are different, to be sure, but the principles that will help to improve quality of product and of service are universal in nature.[30]

Industrial DevOps is the embodiment of this belief.

How to Read This Book

Throughout this book, we will show how to successfully adopt Agile/DevOps ways of working at industrial scale through the adoption of Industrial DevOps. Through nine key principles, organizations will learn how to bridge the gap between software engineering and systems engineering, how to build cyber-physical systems with speed and quality, and ultimately, how to stay ahead of their competition to deliver value to their customers.

We have organized the book into several parts. To start, Part I expands on the reasons we must change how we work, explains what Industrial DevOps is, details the nine key principles, and expands on the benefits of adopting Industrial DevOps in your organization.

In Part II, we dive deeper into each of the nine principles, where we explore the underlying concepts and how they are applied to different domains.

In Part III, we bring it all together and help show you how to start adopting these ways of working in your company or on your team. We also dig into the many barriers that currently block adopting Industrial DevOps and show you how to overcome them.

Throughout the book, you'll find case studies of real-life companies adopting these ways of working, coaching tips, questions to ask, and key takeaways.

We routinely use the example of a CubeSat (a cube-shaped miniature satellite) throughout the book to illustrate the concepts being discussed. This running example helps us build a common mental model as we discuss and explore the principles of Industrial DevOps. We chose a CubeSat specifically because it is a cyber-physical system that is relatively simple to understand but can scale in size and number to demonstrate increasing levels of complexity. If you are unfamiliar and need a crash course in CubeSats, head over to Appendix A.

Appendix B outlines the bodies of knowledge that have been combined and used to inform and design the nine key principles of Industrial DevOps. If you are unfamiliar with any of these bodies of knowledge, be sure to check out the materials in Appendix B. Often, it seems like we are all speaking different languages, but to be successful in the next phase of the industry, we must learn to speak the same language. This section will help get us there.

Appendix C summarizes the variety of tools and techniques mentioned through this book. It can serve as a reference for you and support your Industrial DevOps implementation.

Who Should Read This Book

- You're a program manager in the organization who is looking to improve delivery lead time for cyber-physical systems.

- You're a systems architect who wants to understand how Agile and DevOps can be applied to cyber-physical systems.
- You're at the executive level, sponsoring an Agile, DevOps, Lean, or digital transformation initiative across the enterprise. Our research and stories will help you understand the broader application of these principles beyond software and how it impacts the entire value chain to include all functions of the organization.
- You're sitting at the portfolio level, or you're a program manager seeking ideas to evolve and scale your existing Agile software development approach to be more inclusive of hardware teams and Lean and Agile manufacturing teams. You are seeking to improve operational efficiencies, reduce risks, and improve schedule delivery.
- You're a technical manager or leader or a plant or site manufacturing manager focused on operations, and you have noticed a gap between Agile software development teams and the other teams who also have responsibility for the design, build, and delivery of the product. You are interested in exploring new ways of working, from the ideation stage through manufacturing and delivery of products.
- You're a DevOps or Agile coach and a primary change agent for the customer or organization you serve.
- You're curious and looking for principles and ideas to apply in the development of cyber-physical systems.

This book focuses on large-scale, complex environments. However, the principles can be applied in a variety of setting and domains. We are excited to support your learning journey. Our goal is to educate and inform while providing concrete actions that can be taken to lay out a strategy and road map for improvement. This is an opportunity to learn and improve your current ways of working in the cyber-physical realm to improve business outcomes and build happier, more engaged, and empowered teams.

PART I

APPLYING THE SUCCESS OF AGILE/ DEVOPS TO CYBER-PHYSICAL SYSTEMS

CHAPTER 1

TOWARD A PRACTICE OF INDUSTRIAL DEVOPS

> Change is the law of life. And those who look only to the past or present are certain to miss the future.
>
> —John F. Kennedy

The notion of iterating and accepting "vague" requirements for large-scale, cyber-physical, and often safety-critical systems may seem unrealistic. For many cyber-physical organizations, this has been a stopping point to adopting these new ways of working. However, this is exactly how some of our greatest accomplishments have occurred. Consider the Apollo 11 mission that achieved the first human landing on the moon on July 20, 1969. The level of uncertainty throughout that mission was high, and the need to learn fast was critical. But they iterated, learned, and eventually landed on the moon.

In today's fast-paced business environment, organizations must adapt to changing market conditions, emerging technologies, and evolving customer needs to remain relevant. Agile and DevOps practices give us the means to do so. Software may have been the point of entry, but it is not the end. Agile/DevOps practices can be adjusted to fit the unique challenges and needs of building cyber-physical systems by shifting our mindsets and changing the way we work. Industrial DevOps shows us the way.

Bringing Agile/DevOps to Cyber-Physical

Since we first landed a man on the moon in 1969, digital engineering technologies have continued to mature with leaps and bounds. The cyber-physical world *can* take advantage of these digital capabilities to gain flexibility and adaptability even in the physical space, giving companies that create cyber-physical systems the ability to disrupt the market and improve delivery times.

In fact, digital capabilities create the opportunity to shift physical system development into a digital realm using tools such as emulators, simulators, digital modeling, and digital twins. Today, we also have the advantage of computer-integrated manufacturing, 3D printing, and additive materials that reduce costs over traditional methods. These abilities give cyber-physical teams greater flexibility in the design of physical products and the ability to test more frequently.

And let's not lose sight of changes happening in manufacturing. With the emergence of Industry 5.0* and the smart factory, how work is performed on the factory floor is also changing. The factory itself is now a cyber-physical system used to build cyber-physical systems. This is a result of the adoption of the "Internet of Things (IoT), cloud and edge computing, Artificial Intelligence (AI), big data and analytics, blockchain, robotics, drones, 3D printing, Augmented Reality (AR), and Virtual Reality (VR), Robotic Process Automation (RPA) and mobile technologies."[1] These digital capabilities help reduce the cost of traditional manufacturing methods, enabling development to frequently validate and test multiple design options and create iterative capabilities.

Additional practices such as modular hardware designs and production flexibility, robotics, automation on the factory floor, and other enabling digital capabilities are improving the speed of delivery from the ideation phase through development, production, and operations.

So, with these new technological advances, what's holding back companies from adopting new ways of working? The challenge may lie in outdated mindsets. The previous constraints of physicality have been reduced, paving the way for a new way of building.

The traditional development of cyber-physical solutions has been conducted through a serial life cycle flow of design, development, and testing (known as waterfall). The stage-gate milestones of this waterfall life cycle focus on completed documentation (and lots of it) versus validated capabilities. This results in creating a lot of documentation but a nonfunctioning system. The net result is slow time to market, lower quality, cost overruns, and solutions that do not fit their intended purpose. Industrial DevOps does not advocate for zero documentation; instead, it advocates for right-sized documentation in concert with ongoing, iterative development and validated capabilities.

* Industry 5.0 is a new and emerging phase of industrialization that describes humans working in concert with advanced technology and AI-powered robotics.

These waterfall practices have been used for decades. In the past, the cost of change was much higher, so organizations focused on controlling change to keep costs low. Today, the cost of change has diminished thanks to technological advancements. Today the risk of *not* changing is the bigger monster. Jeanne W. Ross of the MIT Sloan Center for Information Systems Research stated, "Clearly, the thing that's transforming is not the technology—it's the technology transforming you."[2]

Digital transformation has also increased customer access to information, which has skyrocketed customer expectations in everything from innovation to the speed of delivery through operations. These expectations are prevalent for digital and cyber-physical products, such as smartphones, wearables, smart appliances, medical devices, vehicles, and even weapons systems.

According to McKinsey, the COVID-19 pandemic further accelerated the digitization of companies and their supply chains by three to four years, with some digitally enabled products accelerated by seven years.[3] It has never been clearer that now is the time for companies that build cyber-physical systems to adopt new ways of working.

Applying the theory, practice, and learnings from Agile and DevOps has the potential to dramatically improve the development and delivery of cyber-physical systems. Companies that solve this problem will increase transparency, reduce cycle time, increase value for money, and innovate faster. Simply, they will build better systems faster, and they will become the ultimate economic and value delivery winners in the marketplace. These practices are especially useful as the systems become increasingly complex with growth in unintended emergent behaviors.

Taking proven principles and practices from Lean, Agile, and DevOps and implementing them at the system level with a common language and mental model is actually a simple idea. That simple idea executed well can cause world-changing ripples in product development.

In 2012, I (Robin) had the opportunity to support fighter jet teams to deliver updates to a legacy cyber-physical system with multiple safety requirements. We had an aggressive but necessary schedule of thirteen months to deliver the updates. Given the schedule constraints, leadership was willing to take a risk on a new approach to development.

I coached an initially reluctant set of teams through an Agile transformation. The teams leveraged a tiered planning approach, decomposing their work by product, working in pairs, developing in timeboxes, holding daily stand-ups, and performing demonstrations of their work. At the close of each time box (sprint), the teams held a retrospective and identified one change they could apply to the next sprint.

The impact the Agile approach had on the system was immense. The system was completed in seven months as opposed to thirteen, with a record-low number of defects in hardware integration. The impact that the Agile approach had on the team and their morale was beyond amazing. Aerospace engineers, who had been building and maintaining aircraft for over thirty years, claimed they had never had more fun during the development process or a greater impact on the system.

As organizations realize the benefits of Industrial DevOps, the opportunity to disrupt the status quo of the entire product life cycle presents itself. For the US Department of Defense (DoD), this is imperative in ensuring the safety and freedom of the United States and its allies. For the space industry, it means getting the edge on space advancements and human discovery. For the broader community, it provides an opportunity to outpace their competition.

According to the National Science Foundation (NSF), "Advances in [cyber-physical systems] will enable capability, adaptability, scalability, resiliency, safety, security, and usability that will expand the horizons of these critical systems. [Cyber-physical system] technologies are transforming the way people interact with engineered systems, just as the Internet has transformed the way people interact with information."[4]

Despite the clear evidence that Agile and DevOps practices have played a key role in the success of software development organizations, many organizations that build cyber-physical systems have the mistaken idea that their systems are too complex to use Agile or DevOps practices. From experience, we recognize that even the simplest of ideas can be difficult to implement when working against cultural norms. What we need is to look at the problem we have been solving from a different perspective. Henry Ford is credited with saying, "If I had asked people what they wanted, they would have said faster horses." People did not consider that an entirely new form of transportation could be made available.

Current State of Cyber-Physical Systems

The term *cyber-physical system* was first used in 2006 by Helen Gill at the US National Science Foundation. According to the NSF, cyber-physical systems "integrate sensing, computation, control and networking into physical objects and infrastructure, connecting them to the Internet and to each other."[5] Building from this definition, we include systems that are part of private, secure networks and communications infrastructures. These systems, including software, hardware, and manufacturing components, are often complex and costly to build. Many cyber-physical systems have safety and security requirements, making these systems even more challenging to adapt to changing priorities and technologies: "Security threats have a high possibility of affecting [cyber-physical systems, and they] can be affected by several cyberattacks without providing any indication of failure."[6]

Cyber-physical systems are everywhere. You see and use these systems as part of your daily activities. They exist across industries in many different forms. Cyber-physical systems can be found in the automotive industry, agriculture and farming, aeronautics and space systems, undersea systems, energy systems, medical and health care, communication devices, smart factories, smart grids, wearable devices, and more.

While there are challenges to the adoption of Agile and DevOps in cyber-physical systems, it is far from impossible. Let's look at some early adopters of Industrial DevOps principles and practices.

The Early Adopters Are Reaping Success

Industrial DevOps practices provide an innovative approach to address many of the unique challenges faced by the development and manufacturing of cyber-physical systems. Where we see pockets of these principles being adopted, the companies are recognizing clear benefits. Early successes of some Industrial DevOps principles can be found in companies such as Tesla, SpaceX, Planet Labs, Bosch, and Saab Aeronautics.

Tesla

To build a competitive presence in the auto industry, Tesla applies what they call "first principles" coupled with attention to how people collaborate. This means when they want to improve a system and innovate faster, they take a systems view. Innovative thinking increases through increased knowledge sharing and strong team collaboration.

Through the efforts of Joe Justice, chair of the Agile Business Institute (ABI) and CEO of Wikispeed Inc., applying an Agile mindset and practices has been fundamental to Tesla's ways of working and their success in building a continuous improvement culture. Tesla CEO Elon Musk has stated the importance of innovation at Tesla: "Speed of innovation is what matters."[7] As reported in the ABI article "Tesla Agile Success," while innovations are ongoing at Tesla, some specific innovations include "designing the battery casing as part of the car structure, the Giga Press, the octovalve and its manifold, and enhancing their car seats from being the worst in the early Model S to being the best on the market (according to Sandy Munro)."[8]

One well-known investment is the Giga Press. The use of the Giga Press to manufacture large parts of vehicles has resulted in increased speed of production and improved operational efficiencies. By using the Giga Press to manufacture parts for the Model Y, the Fremont factory was able to reduce the body shop by 30%, or about three hundred fewer robots, as compared with the manufacturing of the Model 3.[9] Taking a systems view and building teams for improved collaboration helps Tesla build better systems faster. Today, they have a larger market cap than the next six auto companies combined.[10]

SpaceX

Anyone who is familiar with Agile principles and is watching SpaceX recognizes the importance these principles play in the innovative learning cycles of their space systems. As reported in the article "SpaceX's Use of Agile Methods," SpaceX makes vast use of end-to-end systems modeling, and as Elon Musk says, they can "take the concept from your mind, translate that into a 3D object, really intuitively . . . and be able to make it real just by printing it."[11]

The Industrial DevOps principles have been demonstrated by SpaceX through their ability to test often and early and get rapid feedback on their testing. Using these principles, in December 2022, "SpaceX has now edged out Lockheed, which has a valuation of $137 billion, making it the third-most-valuable aerospace and defense franchise in the western world behind only Raytheon Technologies (RTX), which holds the top spot, and Boeing (BA)."[12]

Planet Labs

Planet Labs is an Earth-imaging company that operates a fleet of over two hundred satellites, providing high-resolution imaging of the Earth's surface for a variety of stakeholders, from agriculture to government to energy. Planet Labs refers to their approach as "Agile aerospace," where, over the last decade, they have completed fourteen major iterations of the Dove spacecraft design. Planet Labs explains that getting their first satellites into space quickly enabled them to learn many lessons about constellation management and optical systems before the cost of change was too expensive. These days, Planet Labs launches satellites into space every three to four months. They were able to parlay early learning into a company that is currently estimated to be valued at over $2.8 billion.[13]

Bosch

Bosch is a German engineering and technology company whose products are seen in our everyday life. For several years, the organization has been leveraging Agile and DevOps principles across several areas, such as their autonomous systems and control units, sensors, smart heating controls, and power tool division. Soraia Ferreira, a software engineer supporting Bosch in the area of smart heating and heating systems, stated that by applying Agile practices, “we created a product that has been a smash hit in the market,” and “without innovative physical heating technology, we wouldn’t make any progress with IoT software.”[14]

Saab Aeronautics

The Gripen fighter jet, owned by Saab Aeronautics, has been one of the earliest adopters of Agile, Lean, and Scrum for a large cyber-physical system. For the JAS 39E Saab Gripen, there were two thousand to four thousand individuals working across every level and across software, hardware, and the fuselage. Based on the article “Owning the Sky with Agile: Building a Jet Fighter Faster, Cheaper, Better with Scrum,” Agile practices have enabled Saab “to manage variability and drive performance with clarity and commitment. The result is an aircraft delivered for lower cost, with higher speed, and greater quality.”[15]

NASA

NASA has some of the world’s largest cyber-physical systems in the space domain. They, too, have embraced Agile, Lean, and DevOps principles across some parts of the agency. The Federal News Network captures how their Agile implementation has helped align strategy to execution, with the ability to deliver value incrementally with fast feedback from stakeholders through regular system-level demonstrations. As quoted in the article “Securing Containerized Applications,” Shenandoah Speers, NASA’s director of application and platform services in the office of the CIO, stated that his team has “created a DevSecOps pipeline platform that allows them to do on-demand continuous integration (CI) and continuous deployment (CD) utilizing containerization to automate the build security, scanning and deployment process."[16]

—

Wider adoption of Agile and DevOps principles continues to expand into other functions, as demonstrated through industry professional organizations such as the Project Management Institute, International Council on Systems Engineering (INCOSE), National Defense Industrial Association, and Software Engineering Institute. For example, in June 2022, at the INCOSE 32nd Annual International Symposium, a working session, "SE Modernization Strategy Session Follow-Up," was held to discuss systems engineering modernization to include Agile practices, and the organization continues to promote the practice Agile Systems Engineering. These organizations are taking positive steps to include Agile and DevOps into areas beyond software development.

A 2022 report from the US Government Accountability Office (GAO) highlights the DoD's emphasis on iterative development and continued modernization of its software development efforts. While iterative development and software modernization are on the rise, fast feedback and the frequent release of capabilities for these systems is still suboptimal. The GAO has also reported on the significance of cybersecurity for weapons systems. These capabilities are critical for ensuring the safety and security of freedom.[17]

These examples are not meant to imply that the efforts of these organizations are perfect; however, it is clear that these organizations are experiencing positive results through a set of Lean and Agile principles for the development and production of cyber-physical systems. Companies that solve this problem first will increase transparency, reduce cycle time, increase value for money, and innovate faster. **They will build better systems faster and become the ultimate economic and value delivery winners in the marketplace.** It is evident that we can leverage Agile and DevOps practices to address the unique challenges of cyber-physical systems and obtain similar outcomes.

What Is Industrial DevOps?

Agile and DevOps have been implemented in software for over a decade. This is not a new phenomenon, but adoption in different environments beyond software is still evolving, as we have explored here and in the Introduction. The solution for wider adoption of Agile/DevOps in cyber-physical systems requires a unique set of principles, which we have coined *Industrial DevOps*.

Industrial DevOps is the application of Lean, Agile, and DevOps principles to the planning, development, manufacturing, deployment, and serviceability of significant cyber-physical systems. The practice of Industrial DevOps pulls from multiple bodies of knowledge, including Agile, Lean, DevOps, and systems thinking, as well as from our own personal experience delivering cyber-physical systems in the new technological revolution.

Industrial DevOps bridges the principles and practices of Agile, Lean, and DevOps with the unique needs and challenges of cyber-physical systems through nine principles.* By doing so, Industrial DevOps enables organizations building cyber-physical systems to be more responsive to changing priorities and market needs while also reducing lead times and costs. Through research and experience, we have found that the combined use of these nine principles is effective in successfully delivering cyber-physical systems across industries.

1. **Organize for the Flow of Value:** Organizing for flow provides guidance on how to align your multiple product teams for regular demonstration and delivery of value.
2. **Apply Multiple Horizons of Planning:** Apply multiple horizons of planning to address scaling and complexity while leveraging ongoing experimentation and learning.
3. **Implement Data-Driven Decisions:** Data-driven decisions use current observations and metrics to determine the state, manage the flow of work across systems of systems, and continuously improve with real-time data.
4. **Architect for Change and Speed:** Architecting for change and speed provides information on multiple architecture considerations, which can reduce dependencies and improve the speed of change.
5. **Iterate, Manage Queues, Create Flow:** Iterate, manage queues, and create flow to emphasize the importance of fast feedback, experimentation, and continuous learning.

---

* These principles are a modification from the original eight defined in our earlier publication, *Industrial DevOps: Applying DevOps and Continuous Delivery to Significant Cyber-Physical Systems*. The original principles were created with input from multiple contributors from a wide variety of companies across different industries. The nine principles in this book still align with those original principles but have been refined to provide more clarity as we have continued to learn from new experiences and apply a growth mindset to all that we do, which led us to the additional ninth principle: Apply a Growth Mindset. We are committed to continuously learning and growing in our experiences and in our thinking.

6. **Establish Cadence and Synchronization for Flow:** Establishing cadence and synchronization discusses how these two concepts complement each other to reduce variability and improve predictability.
7. **Integrate Early and Often:** Integrating early and often covers different levels and types of integration points across large, complex systems.
8. **Shift Left:** Shifting left emphasizes a "test-first" mindset encompassing the multiple levels of testing across cyber-physical systems.
9. **Apply a Growth Mindset:** Applying a growth mindset expresses the need to continuously learn, innovate, and adapt to the changes around us in order to stay competitive.

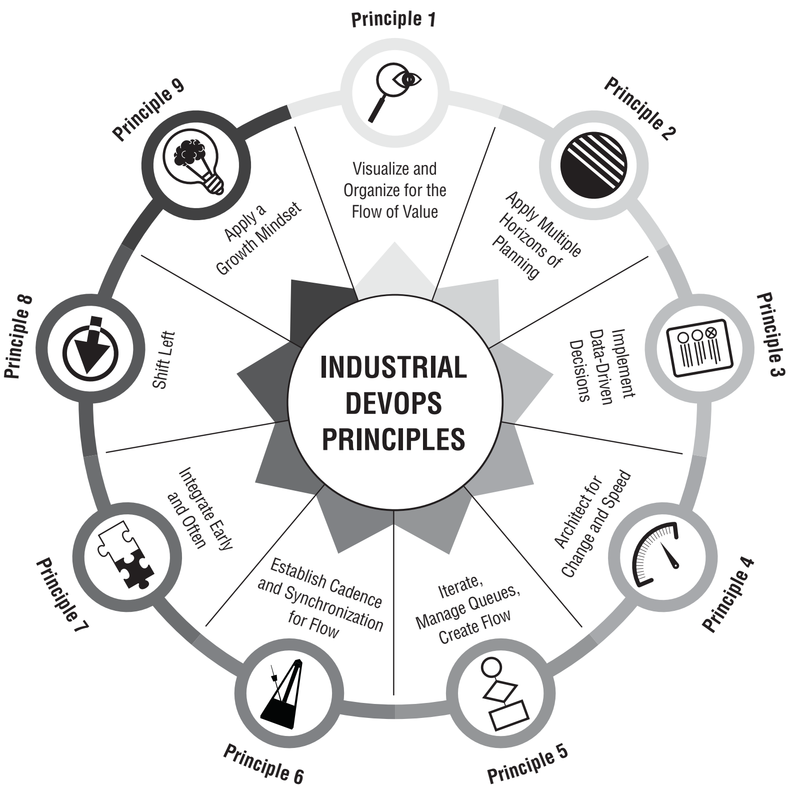

Figure 1.1: Industrial DevOps Principles

Let's take a look at Industrial DevOps in action through a brief glimpse into each of the nine principles. We will take a deeper look into each principle in Part II of this book.

1. Organize for the Flow of Value

The first step in adopting Industrial DevOps is to visualize and organize around the flow of value instead of around functions. This may sound obvious, but many companies are organized around functional activities, such as systems engineering, hardware engineering, software engineering, test engineering, etc. This type of organization creates multiple hand-offs and lots of documentation. Instead, teams should be organized around value streams, and the teams building the systems within the value stream include people with all the skills needed to improve the flow of value and shorten delivery cycle. (See Chapter 4 for more.)

2. Apply Multiple Horizons of Planning

Predictive planning with phase gates has been the most popular approach to building cyber-physical systems. The belief has been that short-term empirical planning, which allows software systems to design and iterate at speed, won't work for cyber-physical systems, with their longer lead times for hardware, dependencies across systems and systems of systems, regulatory controls, and more.

Industrial DevOps employs *multiple horizons of planning* to address this unique challenge. This helps move organizations away from long predictive planning to the short-term planning common with Agile. This approach allows teams to obtain empirical data quickly from planning horizons and apply the knowledge to the next planning horizon, always adjusting the planning based on empirical data. (See Chapter 5 for more.)

3. Implement Data-Driven Decisions

The third principle of Industrial DevOps focuses on using empirical data and leading indicators to better understand the progress and state of the product we are building. The data is then used as input into the decision-making process as the next cycle of work is planned and prioritized. Using data to drive decisions also provides the ability to measure the results of those decisions. (See Chapter 6 for more.)

4. Architect for Change and Speed

Architecture is a critical element in the ability to deliver products and services at speed. Concurrent development is much faster than synchronous development. In Principle 3, we decomposed our systems into smaller components and threads that can be validated and verified, allowing us to learn faster. In Principle 4, we apply a modular architecture to address speed and change, as it enables small teams to build and deliver faster with reduced dependencies between other teams who may be working on the same system module.

Through modularity, a team can change part of the design without impacting the other parts of the system. Architecting the system for serviceability means considering how the system components are updated or enhanced during the ongoing development of the system, whether before or after deployment. This approach makes change easier and increases the speed of delivery. (See Chapter 7 for more.)

5. Iterate, Manage Queues, Create Flow

In Principle 1, we organized our teams for flow. Now, in Principle 5, we create and maintain that flow using iterative and incremental approaches to build out complex solutions. This principle couples Agile's approach of iterative and incremental development with Lean's commitment to reducing batch size to increase flow. (See Chapter 8 for more.)

6. Establish Cadence and Synchronization for Flow

In Principle 6, cadence and synchronization come together to improve flow and establish predictability, which is critical when building cyber-physical systems. A lack of predictability is one of the leading detractors for organizations to adopt more Agile ways of working.

Large cyber-physical systems typically have multiple teams designing, implementing, and deploying multiple interconnected subsystems and components over long periods of time. All of this results in unknowns and variability.

In product development, our goal is to exploit good variability and remove bad variability. The trick is knowing which is good and which is bad. Cadence and synchronization are two tools that aid in removing bad variability while providing the opportunity to exploit good variability. (See Chapter 9 for more.)

7. Iterate, Manage Queues, Create Flow

The more frequently we integrate, the faster we learn and the better our product. However, continuous integration may not be possible for every system. As we scale beyond software, we must consider constraints associated with physical products. For cyber-physical systems, our goal is to integrate as frequently as possible to evolve the system to meet customer needs and get feedback on what is working and where improvements can be made. Some considerations that impact integration are investment of automation, lead times in hardware, expensive test equipment, regulatory compliance rules, and training of employees on the tools and processes. (See Chapter 10 for more.)

8. Shift Left

We must begin with a clear idea of how we are going to test so we can build quality into our products and services. This means thinking first about how we will test so that we have achieved the desired outcome before we start the development of products and services. The importance of early integration and iterative testing is not uncommon in the hardware domain. Models, prototypes, and simulations have been adopted for years. We now align these practices with being test driven, developing iteratively, and organizing around the flow and delivery of value. (See Chapter 11 for more.)

9. Apply a Growth Mindset

Finally, whenever we undertake dramatic change, we must ensure that we approach that change with a growth mindset. Without this final principle in place, none of the other eight matter. This is the principle that glues all the others together. We will discuss what a growth mindset is, why a growth mindset is important, and how to build a growth mindset both individually and as an organization. In addition, we describe the relationship to successfully driving toward Industrial DevOps principles by leveraging the power of a growth mindset. (See Chapter 12 for more.)

Conclusion

The principles of Industrial DevOps have been built from the integration of existing principles. Just like all the innovators before us, we have taken some good ideas and applied them in new ways that we have found to work successfully for our unique needs. Our goal is to provide you with some insights to help you jump-start your approach to delivering products and services better. In the next chapters, we look at the key benefits and misconceptions of applying Industrial DevOps to the development and production of cyber-physical systems.

KEY TAKEAWAYS

- There are companies applying Lean, Agile, and DevOps who are seeing positive results. We can leverage these successes to build more success.
- Industrial DevOps principles extend across the value stream at all levels of the organization.
- Industrial DevOps is a new way to look at building systems of systems with hardware, software, and firmware/manufacturing.
- Benefits of Industrial DevOps include shorter lead times, improved transparency and visibility of progress, increased predictability, improved quality, reduced cost, and improved morale.

QUESTIONS FOR YOUR TEAM TO ANSWER

- Why is it important to extend the Industrial DevOps principles beyond software and across the functional areas?
- Which principles are already being applied in your team environment? Are there any principles being applied at scale in your organization?
- Are there principles that are easier to apply in software than hardware? If so, why? What would your team need to drive the principles into hardware development?

COACHING TIPS

- Industrial DevOps principles are founded upon well-established, understood bodies of knowledge (see Appendix B). Your organization is already applying some of these practices. Use that as a starting point and improve from there.
- Consider which of the principles your organization is struggling with the most or which principles you are most curious about. Write them down and be prepared to reflect on this as you read and learn more in later chapters.

- When applying Industrial DevOps principles to cyber-physical systems, the goal is to define and demonstrate what we can learn instead of what functionality we can deliver. What is something you have recently done at work that you could demonstrate to someone on your team for feedback in the spirit of continuous improvement?