Brought to you by:

IBM

Putting the Ops into DevOps

for dummies®

A Wiley Brand

Deliver maximum value for application releases

Enhance end-user experience

Evaluate technologies and best practices

Dennis Nils Drogseth

Sudhakar V. Chellam

IBM Limited Edition

Putting the Ops into DevOps

IBM Limited Edition

by Dennis Nils Drogseth and Sudhakar V. Chellam

for dummies®
A Wiley Brand

Putting the Ops into DevOps For Dummies®, IBM Limited Edition

Published by
John Wiley & Sons, Inc.
111 River St.
Hoboken, NJ 07030-5774
www.wiley.com

ISBN: 978-1-119-46340-5 (pbk); ISBN: 978-1-119-46335-1 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Publisher's Acknowledgments

Some of the people who helped bring this book to market include the following:

Project Editor: Carrie A. Burchfield
Editorial Manager: Rev Mengle
Acquisitions Editor: Steve Hayes
Business Development Representative: Sue Blessing
Production Editor: Antony Sami

Introduction

"Putting the Ops into DevOps" may suggest many things to many different readers. To some focused on a "No-Ops" model, it may sound irksome and unnecessary — as if to ask, "Why bother?" To others, particularly those in Operations, it may herald a much-needed call-to-arms. To others looking at a yet broader set of requirements for IT-to-business alignment, it might invoke a hopeful sigh of relief. This book is designed to address all three perspectives, although it resides most strongly within the third sigh-of-relief category.

The notion of DevOps has roots that go back to the very foundations of effective IT service management and service delivery. However, in the present era of cloud, microservices, containers, and — also relevant — an increasingly impatient service consumer population largely resident on mobile — DevOps has evolved into an arena of innovation and controversy. The notions of agile computing and continuous delivery have strongly flavored the idea of DevOps with speed and dynamic relevance to a degree that would've been all but inconceivable ten years ago. Moreover, the advent of digital transformation is slowly beginning to redefine how IT organizations are modeled and evaluated — a kind of IT transformation in which evolving approaches to DevOps play a critical role.

Best practices and processes are also changing, as in a growing number of IT organizations, IT Infrastructure Library (ITIL) concepts, and processes for non-disruptive change management are starting to join hands with the more fluid and agile approaches inherent in Scrum in more progressive DevOps initiatives. Finally, changes in IT performance and change and capacity management-related technologies are also playing a role — potentially for the positive — in enabling new levels of predictive dynamic awareness, higher levels of automation, and more compelling, role-aware approaches to visualization and data sharing.

Being aware of all this — and gaining a sense of perspective over and above the marketing hype and the technological clutter — is central to putting the Ops into DevOps, as well as to effective DevOps campaigns more broadly.

About This Book

Putting the Ops into DevOps For Dummies, IBM Limited Edition, is designed to provide insights into the mix of technology trends, processes and best practices, and IT organizational and role shifts impacting Operations overall and DevOps in particular. The goal is to help you stand back and gain a perspective about optimizing operational insights into application/infrastructure behaviors and associated business outcomes, as well as understand what's really occurring in IT organizations, versus some of the hype. Even more importantly, this book is designed to provide you with concrete recommendations for going forward in elevating the role of Operations in DevOps — with benefits for Operations and Development and for the businesses IT serves.

Icons Used in This Book

The following icons are used to point out important information throughout the book:

TIP

Tips help identify best practices in DevOps or ways to save time and money.

WARNING

These icons point out content that you should pay attention to. You can find information here about avoiding common mistakes in DevOps from an Operations perspective, or more generally.

REMEMBER

This icon highlights important information that you should remember.

TECHNICAL STUFF

Information here you don't necessarily need to know for your basic knowledge of DevOps, but you may find these facts interesting.

REAL-WORLD EXAMPLE

Many use cases throughout DevOps can help you put context into real life. This icon shows you real-world use cases or direct quotes from companies putting DevOps to use.

IN THIS CHAPTER

- Seeing the impact of digital transformation on IT Operations and DevOps
- Looking at the various shades of operations management

Chapter 1
Putting the Ops Back Into Operations

The role of IT Operations is changing. But how it's changing is often a matter of opinion, and of course, it varies based on unique business and IT organizational environments. In this chapter, you look at how IT Operations is changing from a variety of perspectives and how each plays a role in putting the Ops into DevOps.

You might feel that the sections in this chapter are all challenging conversations, but these perspectives not only set the stage for the rest of this book, but also they're something you can leverage if you're engaged in Operations-related initiatives at almost any level: senior executive IT management, mid-level IT management, or savvy IT professionals.

The Impacts of Digital Transformation on IT Operations and DevOps

Digital transformation is directed at optimizing business performance and organizational effectiveness through investments in digital services. It has long been the case, of course, that

enterprises and other organizations depended on effective IT service delivery for their day-to-day workplace functionality, as well as some of their business outreach. But digital transformation takes this a step further and looks at business and IT outcomes more creatively. It includes not only improving business efficiencies but also enabling entirely new business models, or increasingly, entirely new businesses — the most obvious of these being a whole crop of Internet innovators.

TECHNICAL STUFF

More than 75 percent of digital transformation initiatives include support for DevOps or agile computing, and most view it as very important or critical to the broader digital transformation initiative.

Operational transformation is a proven foundation for digital transformation. It's directed at optimizing IT performance to more effectively address business or organizational needs or outcomes. Operations can play a central role not only in remediating routine errors but also in integrating and coordinating initiatives that help direct the course of IT as a whole. To do this, Operations needs to invest in both improved process awareness and advanced technologies, such as higher levels of automation, more dynamic and multi-purpose capabilities for service modeling, and above all, advanced analytics for Operations and for IT as a whole.

Not surprisingly, in most cases, operational transformation has been in place longer than digital transformation, which is a good thing. Digital transformation completely depends on some level of effective operational transformation, including effective DevOps (after all, no one can be "transformed" if a service isn't there when it's needed), and business leaders are increasingly looking to IT leaders, both executive and professional, to support them in their business creative decision making.

REMEMBER

When considering operational transformation, because it can support both DevOps and transformational initiatives more broadly, think about the following:

- **Cross-silo teaming:** This is at the heart of operational transformation and often requires attention to processes, technology, and all too often organizational shifts. We could write a whole new *Dummies* book on this topic, but much of *this* book addresses this requirement in a DevOps context.

REAL-WORLD EXAMPLE

In the realm of cross-silo breakage and dealing with siloed tools, a mid-tier, U.S.-based healthcare provider had this to say:

"Our organization had largely domain-focused tools. The processes were very broken. The server team didn't talk to the network team, and the network team didn't talk to storage. Our people weren't trained to share information. So it was hard to get even a temporary fix, let alone an effective resolution of ongoing or recurring problems."

- **More dynamic:** Operations needs to be faster and more responsive to business needs, as well as in assimilating new technologies to meet those needs.
- **More transparent:** IT as a whole needs to become more transparent and run more like a business in the digital age. Operations can and should become a useful center to support transparency by promoting and documenting new levels of IT efficiency, as well as by delivering improved insights into how and why IT services are consumed.
- **More technology aware:** Operations needs to invest in not only improved process awareness but also in advanced technologies such as higher levels of automation, more dynamic and multi-purpose capabilities for service modeling, and above all, advanced analytics for Operations and for IT as a whole. Advanced Operations Analytics, or what EMA calls advanced IT analytics (AIA), are becoming a great unifier for IT organizations willing to work more proactively — responding not just to incidents but to predictive insights that foresee problems before they happen.
- **More service-aware:** Effective Operations will have to move beyond managing isolated components to understanding application service interdependencies across cloud and legacy environments.

REAL-WORLD EXAMPLE

In creating siloed awareness, an international gaming company headquartered in the U.S., was trying to move beyond task by free association:

"We had to get past 'task by free association,' where everyone's on a phone call and begins to speculate about where the problem might lie — and wonder if it was a power loss, or maybe a problem with the network or a service provider — with everybody going off in every direction at once."

- **More business-aware:** This may seem like a stretch for an IT professional caught up in a maelstrom of runbooks gone awry and event tidal waves. But the understanding that IT is no longer separate from the business, but an integral part of the business, is critical for going forward both with DevOps and with digital transformation as a whole.

Looking at the Shades of IT Operations

The truth is that the role of IT Operations varies in different business and organizational environments. In some, it's seen increasingly as a backwater of diminishing importance eroded by investments in the public cloud and LoB-centricity. In other environments, IT Operations is taking a more central leadership position in coordinating both technology and process efficiencies across all of IT, including LoB-centric teams, and increasingly with Development and Security. Although it is the more empowered role of IT Operations that is likely to bring the best value in putting the Ops in DevOps, this book is intended to provide guidance for Operations at any level.

Central IT Operations or LoB-centricity

Both central IT Operations and LoB-driven Operations can become the right strategic centers for putting the Ops into DevOps. A lot depends on size, investment level, and synergies with other teams ranging of course from application development and application owners to integrations with IT service management (ITSM) teams where, workflow, governance, and shared insights on application issues can all be of value.

REMEMBER

When making the choice in leadership between central IT Ops and LoB-driven Ops, consider the following:

- **Ownership of the full application/infrastructure stack is critical.** Operations should work closely with applications development, application owners, and application management groups. These are increasingly LoB-affiliated. But it's a myth that application performance and infrastructure performance should be viewed separately, especially in the age of cloud when complex infrastructure interdependencies

must be understood dynamically and in real time even before releasing a new release. Moreover, most IT environments today are mixed, with large investments in on-premise legacy infrastructure, virtualized infrastructure, and growing dependencies within the public cloud, as well as the Internet and a growing array of third-party service providers. Owning the full infrastructure stack across this full hybrid IT landscape is therefore key, whether for central IT Operations or LoB-centric Ops teams.

- **Affiliation with business stakeholders and the IT executive suite are both needed.** DevOps isn't just about fixing problems when they move into production. What we see more and more given trends such as digital transformation is the need for executive IT, Operations, development, and business stakeholders to work together. While LoB-centric Operations may be closer to business stakeholders, central IT Operations is likely to be closer to the IT executive suite and hence overall IT governance. Both count, and in the end, both are needed for optimizing what Operations can do for DevOps.
- **Being slated for growth is also a plus.** Optimizing Operations' role in DevOps usually requires making investments in good technology (we talk about this more in Chapter 3). These include everything from investments in big data and advanced IT analytics, to more progressive approaches to service modeling, to advanced levels of workflow and automation that can help support and inform both LoB-related operations and all of IT. It may also require investments in best practices and improved processes that can be shared between Operations and Development. Good technologies can promote new ways of working that need to be socialized, agreed on, and, at least at some level, documented.
- **Being cloud-migration ready is also key.** DevOps is becoming increasingly cloud-centric. In general, a unified approach to optimizing cloud across all of IT has proven to be far more efficient than case-by-case, ad-hoc approaches, which may also carry significant security risks. On the other hand, LoB-centric operations teams can often lead the charge, creating a path for the rest of IT to follow.

So in the end, DevOps leadership may come from LoB-centric Operations or central IT Ops, or a mix of both. This checklist is a way of helping you to decide which makes the most sense in your own, unique IT organization.

Operations and application management

Effective operations teams are increasingly bridging visibility into application performance with infrastructure performance through more advanced management software and analytics. This has become all the more critical given the complexities and diversity of cloud-related environments. Here's where having IT Ops reach out to often LoB-centric teams in application development, as well as often more centralized teams devoted to applications support, becomes essential, both for DevOps and for application performance management overall. The digital operations war room should not be just about infrastructure. Instead, it should provide a dynamic vision of changes, anomalies, and critical transaction latencies associated with the full stack of IT infrastructure interdependencies.

Operations and ITSM

Various definitions of operations teams apply, but more often than not the bigger picture includes integrations with ITSM teams, at least at some level. Operations is more likely to own ITSM organizationally than any other single group (the next most likely owner is the IT executive suite). The dialog between ITSM teams and operations staff should be bi-directional because only relevant incidents and events should be passed on from Operations, while workflow, governance, and process requirements typically flow through ITSM.

Empowered by operations expertise and leadership, as well as by technologies such as advanced analytics, ITSM teams may range in direction from support for everything from cross-silo performance and change management, to capacity optimization, to endpoint management, to portfolio planning and, most importantly here, support for DevOps.

BETTER PROCESSES AND EFFICIENCIES

REAL-WORLD EXAMPLE

When ITSM and Operations work together, they create better processes and better efficiencies, as realized by two different types of companies — a global Internet retail company with brick-and-mortar retail outlets:

"First and foremost, we've been able to consolidate our processes for change, incident, and problem management across our entire operation by leveraging one single platform. We've also enjoyed improved visibility into the impacts of changes on service performance and availability, so we can more quickly get to the root cause of many of the issues caused by changes and begin to automate fixes more consistently."

As a part of Operations, ITSM teams often function as a hub of communication across all of IT including Development — a trend that has dramatically accelerated in recent years. Finally, ITSM is increasingly helping Operations become a center of governance, project management, and workflow control for all of IT — all of which figure squarely in putting the Ops into DevOps.

Security and Operations (SecOps)

The role of Security Operations (SecOps) — meaning getting SecOps and Fraud/Compliance to work more actively with IT — is also on the rise, given the growing need to integrate security into performance management, change management, and audit-related activities. The benefits include minimizing disruption to critical IT services and increasing IT operational effectiveness with sometimes dramatic improvements in cost-related efficiencies. IT Operations plays a central role in virtually all SecOps initiatives, especially once you include ITSM teams. In fact, SecOps is most successful when ITSM is involved in coordinating workflows and dialog.

SecOps touches on DevOps in the following ways:

- Minimizes risks in handoffs from Development to Operations
- Creates superior process workflows to support security/compliance-related efficiencies for Development and DevOps
- Educates DevOps on the need for security
- Supports Development directly by ensuring compliant and secure development environments
- Supports Q/A Testing by ensuring secure environments
- Minimizes the time developers spend troubleshooting security-related production issues

IN THIS CHAPTER

- Looking at the DevOps application wheel
- Working with tools in the application life cycle
- Recognizing the power of DEM

Chapter 2
Linking DevOps and Continuous Delivery

The terms *DevOps*, *continuous delivery*, and *agile* are often linked together and for good reason. They combine to support a common initiative directed at accelerating application releases and updates into production without disruption to the business, and with maximum operational efficiencies across IT. However, while some folks in IT equate agile and DevOps and associate both with continuous delivery, these terms are at least three different colors of a common flag:

- **DevOps:** DevOps has the longest heritage. In a sense, it's nothing new. The need to bring applications successfully into production from development is as old as IT itself. Effective DevOps centers on smooth hand-offs between Development and Production that don't disrupt application performance. This requires shared processes and common views and insights. "DevOps" can apply to traditional applications with long development cycles just as much as new, more cloud-ready applications that are made up of many small components and are much more likely to be targeted for continuous delivery.

- **Agile software development:** Agile is directed at adaptive planning and continuous improvement to help accelerate the introduction of critical applications for desired business outcomes. It minimizes documentation and traditional or more rigid forms of project management to support a more fluid, change-aware, dialog-centric way of working. Its roots are also not new; they can be traced back for decades. But the rise of agile has become most dramatic in the last ten years.
- **Continuous delivery:** This is a team-centric approach for software development designed to enable short cycles for the delivery of new releases into production. Continuous delivery can mean many multiple iterations of a single application per day at its most frequent intervals.

The Full Application Wheel

Many applications managed in Operations are not, of course, coming primarily from Development. A recent survey suggests that on average about 40 percent of applications for IT are developed in-house across small, medium, and large enterprises (with little variation due to size). In other words, 60 percent of what Operations has to worry about from an applications perspective isn't necessarily DevOps-related. Needless to say, these average percentages can vary tremendously, and the trend is to accelerate the role of in-house developed applications, which are often the most critical to business outcomes.

Just as third-party applications are becoming more varied in choice and design, in-house developed solutions can take many flavors. Examples of the DevOps application wheel might include

- Traditional web applications
- Web applications optimized for mobile
- Omnichannel applications optimized for multiple endpoints
- Web-services applications
- Cloud-native applications (optimized to run on virtualized environments)

- Cloud-native applications (microservices and containers)
- API-driven applications made up of many components
- Customized commercial applications such as those used for ERP
- Rich media applications and streaming video

Each of these potential DevOps environments — Java, VPNs, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), wireless transmissions, and so on — has unique requirements and interdependencies Operations must be prepared to support a diverse set of applications teams and needs. Because of this, the requirements to support a hybrid environment between traditional application/infrastructures and more agile application types, which some in the industry call Bi-Modal IT (covered more in Chapter 5), has never been greater.

Priorities for in-house applications vary, but some of the most common objectives include the following:

- **External customer satisfaction:** Businesses and organizations have to reach out to a wide set of customers that are external to their organizations, including partners and suppliers, clients, and the most obvious — shoppers. In-house developed applications are often creatively shaped to address these unique needs. Associated with this value are
  - Business competitiveness
  - Customer retention
  - New product creation
  - Business transformation (innovation)
- **Internal customer satisfaction:** Businesses and organizations depend on happy, effective professionals. In-house developed applications can be fine-tuned to business process needs. Associated with this value are
  - Workplace efficiencies
  - Workplace morale
  - Organizational transformation

REMEMBER

The takeaway here is that in-house developed applications can shape both an external business or organizational environment, as well as color the complexion, look, and feel of internal workplace reality. There's a human dimension to DevOps, and understanding and managing that value is essential both for Operations and Development — no matter how far removed it might seem from worrying about a router configuration or CPU utilization on a server. Digital experience management (DEM) can become the critical bridge for bringing humanity into DevOps for both Operations and Development. Check out the later section "Understanding the Rising Role of DEM" for more on DEM.

HOW AGILE IS AGILE?

You may be interested in gauging what's normal for many organizations in the ways they develop code. Here are a few stats:

- **About 25 percent** of organizations and businesses on average are delivering new code multiple times a day.
- **About 20 percent** are delivering new code daily.
- **About 20 percent** are delivering new code multiple times a week.
- **About 25 percent** are delivering new code weekly, biweekly, monthly, or in some cases, every other month.

These averages are of course just that — averages — and not surprisingly depend on who you ask. Development tends to view daily code introductions more frequently than Operations or ITSM teams for instance, but in the end, the percentages are reasonably consistent.

On another note, more than 70 percent of IT organizations have defined a Fast Track for certain applications. And as important as ITSM teams can be in DevOps, these fast-track applications, those with continual updates within a single day, often bypass ITSM. This underscores the need for the overlay team (which we discuss in Chapter 5), where simple handoffs can be made directly between Development and Operations.

Looking at Tools Across the Application Life Cycle

Many tools and technologies are commonly used across the application life cycle and several teams are normally involved. Table 2-1 provides you with an overview of how Operations and Development can come together at different stages of the DevOps process.

TABLE 2-1 Tools Used across the Application Life Cycle

Stage	Lead	Participants	Tools
Design	Development Business stakeholders	Operations	Application life cycle management Requirements management Modeling project management (PM)
Develop	Development	Operations Business stakeholders	Integrated development environments Debug Repository Error scan Build automation DEM Advanced analytics and integrated fault and performance management
Test	Development/QA Operations	Business stakeholders	Test management Test data management API test Integration test (service virtualization) DEM Advanced analytics and integrated fault and performance management

(continued)

TABLE 2-1 *(continued)*

Stage	Lead	Participants	Tools
Deploy/ Release	Development Operations	Business stakeholders	Process orchestration Configuration management Release automation
Manage	Operations	Development Business Stakeholders	Runbook automation Systems management Application management Advanced analytics and integrated fault and performance management DEM Service desk/trouble ticket
Assess	Business stakeholders	Development Operations	Service level management DEM Advanced analytics and integrated fault and performance management

REMEMBER

The opportunity for Operations to share a growing role in the design and assess stages, thanks largely to DEM-related data, is one of the more exciting trends when it comes to putting the Ops into DevOps.

Understanding the Rising Role of DEM

If there is a place to first put a stake in the ground for accelerating the role and value of Operations in supporting DevOps, it's in an evolved, productive, and shared DEM strategy. DEM includes both internal end-user experience management and external customer experience management and allows for a unified approach to both that's central to DevOps success, as well as the success of IT as a whole. In fact, the future indicates a strong role for DEM in all stages of the application life cycle, including application design.

In researching, working, and consulting with various IT organizations, the following values have been achieved from DEM — all of which reflect on DevOps requirements:

- **Business impact:** Monitoring and optimizing IT-delivered business services based on user interactions and business outcomes
- **Performance:** Monitoring and optimizing the effective delivery of business services to their end consumers in terms of application performance
- **Change management/DevOps:** Validating the impacts of change, including impacts from the move to cloud or the introduction of new releases or release changes
- **Design:** Monitoring and optimizing the effects of application design and content on business services for customers and internal end-users
- **User productivity:** Monitoring and optimizing end-user interaction with business services, including critical business processes for internal use or external customer services
- **Service usage:** Monitoring the frequency and outcomes of critical application services to better understand value and relevance

REMEMBER

Typically, DEM requires teamwork just as DevOps does. And often, an effective DEM team can be the best foundation for bringing the operations more pervasively and strategically into DevOps.

REAL-WORLD EXAMPLE

THE POWER OF DEM

To further underscore the many benefits of DEM, we've given you some comments from popular industries in the United States that show a wide range of values.

Getting Proactive with DEM

"Before starting a DEM initiative, the approach used by IT was highly reactive. No metrics were measured, and only a rudimentary understanding of DEM — for example, measuring the call closure rates to

(continued)

(continued)

the Help Desk — was used. We only found out about problems when users complained. We got our hands slapped a lot. Now when we build a trial, we want to do load testing and benchmarking during pre-deployment. This will give us metrics to measure and a better understanding of what will happen with metrics and performance when we roll into production." — *Medical supply company, headquartered in the U.S.*

Evolving Toward Business Impact with DEM

"Measurements moved beyond simple Internet pings to writing 'definitive' transactions — real orders — that encompassed the entire customer experience: shopping, checkout, credit card verification, etc. The goal was to measure availability first and performance second, always with the actual end-user experience in mind." — *U.S.-based e-commerce provider*

Flagging App Design Issues with DEM

"What may seem obvious to you or me may not be obvious to the average user out there who fires up a browser. Our apps development may say, 'Oh, this is easy, anyone can do this!' Which is what one of them said before it was pointed out to them that a significant number of the transactions failed." — *Global media conglomerate, headquartered in the U.S.*

Keeping Development and Operations on the Same Track with DEM

"My group performs the first level triage to assess the application and business impact. We will also reach out to Infrastructure and Application groups for deeper assessments if required. If needed, we can escalate to our governance group to initiate an Incident Management technical bridge call with key infrastructure and application experts. Among other communications, I do daily and monthly trending of the data — and trend it a number of different ways to make it clearer where problems are arising — for instance, outliers where latencies are abnormally high. In other words, I try to make it impossible for Operations and Development to hide from our digital experience management metrics." — *Service delivery overlay team in a large North American financial institution*

WARNING

The reality in many IT organizations, from Operations to Development, is not nearly as progressed as some examples indicate. Just a few unsurprising tidbits to underscore this are

- Development teams most often find out about poor application experiences in the real world through help desk tickets and lodged user complaints.
- Even when pre-release testing is done, many development teams go ahead and send applications into production when there are still signs of DEM failure.
- Critical advanced analytics in support of DEM are often not available, and when they are, all too frequently they aren't shared between Operations and Development.
- In some organizations, Development is so isolated and self-protective that operations engineers have placed stealth monitors in development environments just to get a taste of what might lie ahead before Development sends the application code its way.
- Even when DEM initiatives are in place, without the right technology, basic challenges such as root cause analysis may still remain prevalent.
- The need to communicate at all levels — between Operations and Development, and between IT and the business — is critical for DEM to succeed. But this requires both good technology and cultural and process changes in most IT organizations.

REMEMBER

Understanding the true end-user or customer experience is a human and a technology concern. People aren't generic, and approaching digital experience generically will also lead to failure. Learning what's really needed is a constant dialog that comes more and more not just from business analysts but from analytics-driven DEM data and from dialogs with ITSM teams and social media as end-users and customers have their say.

These issues suggest a need to focus on the following:

- Good technology (see Chapter 3)
- Good organizational models (see Chapter 4)
- Forward-looking trends to facilitate forward-looking needs (see Chapter 5)

IN THIS CHAPTER

- Accelerating DevOps with three core technologies
- Unifying insights with consolidated event management

Chapter 3
Seeing How Technology Can Make a Difference

Technology in itself can't transform Operations, let alone IT as a whole. But without good technology, putting the Ops in DevOps is likely to fail. The good news is that while development teams are enjoying a revolution in how applications can be created, modified, and delivered with a wide variety of new tooling options, operations teams are beginning to enjoy a similar renaissance in technology through several key areas of advancement.

Three Core Technologies For Better DevOps Performance

The investment in automation, service modeling, and analytics is invaluable if Operations is to assume a more proactive and strategic role. The alternatives to not investing in these new areas are bleak because fragmented technologies continue to reinforce fragmented ways of working that only escalate into whirlwinds of failure when new application releases hit the production infrastructure with accelerating frequency.

Automation technology

Of the three unifying technologies for DevOps, the furthest along in most DevOps initiatives today is automation, so we begin here. Both Development and Operations depend on automation tools — for their own teams and for handoffs between the two groups. And both Operations and Development, on average, expect somewhere between 30 and 50 percent of their DevOps tasks to be supported by automation — a trend that will only accelerate in the future, as speed and currency become ever more essential.

TIP

Some of the more prevalent types of automation to look for in DevOps include the following:

- **Advanced levels of workflow automation:** Workflow can be a way of both defining and accelerating processes across IT. With workflow in place, you can help define who owns a problem and how it might relate to anything from an event or alarm to a trouble ticket, to analytic insights about unwanted transactional behavior. In this way, workflow can promote collaboration, which can be essential both in remediation to fix a problem and in managing the changes relevant to the introductions of new releases over a production infrastructure. To do this well, workflow should bridge organizations — from Operations to ITSM teams to Development.
- **Runbook or IT Process automation (ITPA):** Runbook automation is based on wisdom achieved through understanding what to do in critical situations for performance, availability, and change management. Traditionally, runbooks document routine procedures, but when informed by analytics and accelerated with workflow and other types of automation, they become more defined, more precise, and infinitely faster. For instance, runbook or ITPA capabilities can become centers for integrating other automation routines, such as configuration automation or automated diagnostics.
- **Automation for provisioning new application releases:** This is at the heart of effective DevOps and requires a consistent perspective on usage between Development and Operations. The goal here should, however, not just be speed. Application blueprints, and even API-driven amalgamations, may appear to work well in a development environment but

may quickly fall apart when real-world interdependencies — including datacenter, telecommunications, and endpoint interdependencies across geographies — come to bear. This is where everything from DEM to analytics with operational insights across the full application/infrastructure stack to service modeling investments should also come into play.

Service modeling

Just as the advent of advanced topologies helped ignite network management in the early to mid-90s, advances in mapping the full application/infrastructure stack with currency and dynamic relevance are beginning to change the DevOps game. This is true despite the lag between many service modeling systems and the dynamic nature of cloud and API-centric and cloud-related application environments. But don't give up. The service modeling game is changing, too. And it's changing dramatically in some surprising ways — to support both continuous discovery and continuous availability.

In examining where to get service modeling — indeed it comes in different forms and flavors — some of the following options will come forward:

- **Configuration management database or configuration management system (CMDB or CMS):** CMDBs and CMS used to provide a single source of truth. However, that was often a misnomer because data was rarely complete or current. Planning a CMDB should focus on relevance, not just data. And while CMDBs can be a valuable point of context, it's important to seek out, and invest in, more dynamic versions of application/infrastructure interdependencies to keep your CMDB or CMS current — or in some cases to replace a traditional CMDB with a more dynamic modeling system. In all cases, however, the modeling can provide valuable contextual insights in terms of what services reside where and how they may have been impacted by change.
- **Application Discovery and Dependency Mapping (ADDM) for change and performance:** Many of the original ADDM innovators focused primarily on asset and change management, not on performance management. These solutions have evolved to support the more complex and more

dynamic requirements of hybrid cloud. But only a few are capable of aligning with real-time insights on transaction performance. ADDM for change and performance arose largely from a new crop of vendors associated with application performance management (APM), optimized to look at application/infrastructure interdependencies in real time or near real time.

- **ADDM embedded in Advanced IT Analytics (AIA):** Perhaps the biggest source of innovation emerging in the market is a handshake between advanced analytics for IT and more dynamic forms of service modeling. One allows you to see what's there and how it interrelates. The other allows you to know what's going on that's relevant to DevOps performance — with unique insights that traditional monitoring can't achieve. Combining the two dynamically, though often challenging, can become a hallmark for DevOps success for both Development and Operations.

REMEMBER

When investing in service modeling, or any advanced technology, consider your own environment and your own unique DevOps needs. You may be mainframe-centric. Or conversely, your development team may have shifted to focus more on microservices. Or, indeed, you may need support for both. There is no generic answer because no IT environment is generic. Be clear on your needs first and then investigate your options.

Advanced IT Analytics (AIA)

The growing market of analytics in IT is one of the more exciting areas to watch in the technology industry. It's exciting because of the variety and types of vendor innovation and because IT analytics can support data sharing and joint decision making in a way that can be strongly beneficial for Operations, Development, and IT as a whole, as well as business stakeholders. We consider AIA the great unifier.

REAL-WORLD EXAMPLE

A global provider of Internet-based entertainment had this to say about unifying Operations and Development with AIA: "The move to advanced analytics allowed us to unify our applications and operations team with a unified set of insights so that we could share information more effectively. In the past, we caught only 3 percent of our problems proactively. Now that percentage went up to 88 percent. Mean time to repair dropped from hours to as

low as 12 minutes, and we are now able to automate resolutions to known issues."

AIA and DevOps

AIA can be applied to DevOps requirements in multiple ways. These use cases may prove most fruitful in putting the Ops into DevOps with shared data and shared insights. These uses include the following:

- Optimizing application performance by providing rapid feedback to development from production
- Minimizing the time developers spend troubleshooting performance issues
- Supporting a smoother handoff between Development and Operations once a common view of application readiness has been achieved
- Supporting Development directly with shared insights into application/infrastructure issues in pre-deployment scenarios, which also helps Operations better prepare for what's coming
- Providing feedback to Development to help optimize application design

  This data comes from analytics where DEM-related insights have been garnered from either Q/A Test or real-world deployments. Digital experience analytics are relevant to Development, Operations, executive IT, and critical business stakeholders (such as line of business execs or digital marketing leads).

REAL-WORLD EXAMPLE

According to a European financial services company, having a cohesive analytics overlay beyond basic monitoring dramatically helps with being more proactive. With AIA and DevOps, the company has a much more efficient way of managing the event flow, rolling up a spread of events into a single screen to see a clearer picture when new releases are introduced or other changes are made. Now the financial services organization can get a much earlier indication when something's not going well.

Finding the right AIA for you

AIA is a new area of innovation in the industry and not really a market with strict boundaries in the traditional sense. But it does have some common criteria for those seeking to invest wisely.

WARNING

Just pursuing big data for IT, or even investing in some analytic tools, doesn't often set things right. Many IT organizations seek to create large data pots for search and analysis without strong priorities for use case or relevance. This has been disastrous in the past for CMDB deployments, and it will continue to be a token of failure as the quest for AIA moves forward. Big data without big relevance can produce more harm than good.

TIP

To avoid disastrous results, follow these tips:

- **Look for versatile use cases.** This is especially important when it comes to DevOps. These use cases can include performance and availability management, change impact management and change awareness, and potentially even capacity planning insights with an eye to cost and performance across legacy infrastructure, mainframe, cloud, and hybrid environments. The truth is that much of the same data is relevant across all these use cases, so analyzing and optimizing insights only makes sense.

  

  REAL-WORLD EXAMPLE

  Analyzing the different metrics can point to anomalies and can predict an outage. In a real life scenario, a financial services enterprise in Scandinavia was able to dramatically reduce major incidents by 85 percent in the first three months of production by using AIA.

- **Leverage multiple data sources.** Your analytic solution should be able to bring together a wide variety of data types in high volumes across the full application/infrastructure stack. These insights should provide visibility into application/infrastructure interdependencies, often in combination with service modeling. And data types should span events, time series data, logs, configuration insights, and transactional insights, as just a few examples.
- **Assimilate, don't disenfranchise.** Many AIA tools have proven integrations with existing monitoring and other toolsets. The range can be from as low as five or six to as high as 50 or more. The truth is that neither development teams nor operations professionals enjoy being told that the

tools they've been using, and in some cases built their careers on, need to be replaced. Instead, your investment in analytics can provide a common layer of insight and proactive awareness that enables and promotes collaboration and dialog of a new kind without being viewed as an unwanted and disruptive intrusion. Although this isn't to say that once initial integrations take place, you won't be able to stand back and establish common ground for toolset consolidation — often with dramatic savings.

- **More than big data seeks out true analytics.** Just having data to search through is good. But given the dynamic nature of DevOps, agile, and continuous delivery, you'll need a firmer, more analytics-centric approach that not only keeps you current in real time but also anticipates issues before they happen. This can sometimes include if-then analytics to provide you with visibility into what happens in anticipation of new application release changes, as well as prescriptive analytics to recommend specific actions. However, above all, and as a place to start, seek out well-designed approaches to alert you to abnormal (anomalous) behaviors across the full application infrastructure in a way that's meaningful, relevant, and action-inducing.
- **Look for the coverage you need.** Just as with service modeling, you may be centered in mainframes, traditional data centers, or cloud and microservices. Your critical business applications may even reside across all three.

  
  WARNING

  Analytics investments that cover only one piece of your puzzle won't do you much good. Make sure your analytics investment addresses every piece of the puzzle you care about when it comes to DevOps interdependencies. Your analytics need to ingest metrics from many different sources to identify the anomalies.
- **Invest in the right team.** AIA can promote new ways of working. But to optimize that, you need to understand who should be involved and how they should be involved in putting the Ops into DevOps. Check out Chapter 4 for more on this.

Investing in Advanced Levels of Event Management

AIA provides a tremendous set of proactive insights that can literally help transform IT. But IT organizations still need to come together with consolidated events that leverage analytics and service modeling, among other sources as a final, central system of notification and alerts. The ongoing flood of events from many sources can quickly become overwhelming, produce stagnation in IT, and lead not only to finger-pointing but also to perennially unresolved issues. Effective event management has a consistent pattern:

1. **Reduce the noise.**

   With any issue or disruption, large volumes of event noise are generated via faults or anomalies captured by different tools, ranging from application monitoring to infrastructure monitoring to transactional analyses to social media and analytics. These faults and anomalies need to be correlated and grouped together based on their impact on specific incidents.

2. **Dynamically assign the correct application/infrastructure context.**

   This is critical both in helping identify the correct point of remediation and in finding the right subject matter expert (SME) to assume the lead role in remediation.

3. **Alert the right SME.**

   Sometimes teams are spread across the globe to handle issues arising from different time zones. During an outage or problem, finding the right available SME is critical — find one who can hopefully apply automation and analytics as needed to resolve the problem quickly and correctly.

Although in itself it may not be new, event management is critical because without the noise reduction, the teams will be working on too many tickets or too many incidents. Having a dynamically relevant CMDB or CMS can also be invaluable in helping to identify the problem in context with change, service impact, and SME ownership.

IN THIS CHAPTER

- Considering Operations-to-Development collaboration trends
- Understanding leadership needs for effective DevOps
- Looking at processes and best practices for DevOps teams
- Understanding the need for communication

Chapter 4
DevOps Teams — Who Really Leads and Why?

Indeed, DevOps teams are evolving in the industry. Exactly how they're made up, or should be made up, varies based on history, culture, technology, requirements, and IT politics. So rather than trying to provide a fixed answer for a one-size-fits-all approach to DevOps team creation, this chapter examines DevOps teams by looking at four main categories that should factor into your plans for putting the Ops into DevOps. However, exactly how you go about it should depend on where the readiness, leadership, and willingness to go forward is strongest. Starting somewhere and building toward the larger vision is what counts. This chapter also addresses how Operations is changing and provides insight on what information is required by increasingly smaller DevOps overlay teams.

Operations-to-Development Collaborations

Operations-to-Development collaborations as they exist today certainly have a few common trends. But when moving toward a DevOps direction with Operations, start to consider the points in this section.

Multiple versus single teams

While there's strong value in having a consistent core in an integrated Operations-IT service management (ITSM) organization for supporting DevOps needs with shared technology, common processes, and best practices, not all applications are the same, and different tools are often needed for managing microservices and container-driven applications, as well as those strongly API-driven. The growing role of overlay teams (see Chapter 5 for more info) can also help facilitate a multiple-team-based approach.


TIP

While no single best answer exists, we recommend a hub-and-spoke approach with some consistent common investments in people and technology and well-defined overlay teams to support individual, application needs.

Ad-hoc versus dedicated teams

Recent surveys indicate that there's a nearly even divide among IT organizations as to whether DevOps teams are ad-hoc or dedicated with a slight edge to dedicated. And once again, the answer can be both — for instance a dedicated core DevOps team with some ad-hoc overlays. If it's an option, dedicated teams have clear advantages in being able to improve their skills, processes, and tools, and building a consistent set of best practices.

DevOps versus "xyz"

Just what the DevOps team is called can also vary. It's not always "DevOps." Other names reflecting DevOps ownership to fill in that "xyz" placeholder may be application management/support, infrastructure services, IT service management, IT architecture, center of excellence, and digital experience management.

Operations and Development coming together in different ways

In one scenario, the development team is delivering a solution over the cloud. The operations team knows about the dependency of the application components in context with development's deployment strategy. The operations team also receives events from the applications and other sources, and they have runbooks and resolutions for resolving the problems. Collectively, both Development and Operations need to have the visibility into the events and cohesively take actions. But actions on the same event

will be different for the different teams. For example, the development team may want to suppress the event, but the operations team doesn't. Efficiency is improved when tools enable both the operations and the development teams to manage the same event in two different ways.

So what does this mean for putting the Ops into DevOps? Don't try to invent a generic answer for a non-generic situation. All IT organizations carry their own strengths, weaknesses, and cultural flavors. Whether you're hub-and-spoke, dedicated or ad hoc, or both largely depends on your own environment — and that includes how you might want to go about branding your initiative to strengthen Operations' role in DevOps.

Grassroots versus executive leadership

While there's a tendency to think of DevOps as a grassroots and semi-spontaneous effort, top-down executive leadership is far more the norm. DevOps should be overseen by IT and Line of Business (LoB) or enterprise marketing executives because it sits at the heart of both the IT operational efficiencies and the pulse of business opportunity.

TECHNICAL STUFF

On average, more than 60 percent of DevOps initiatives have top-down involvement at the C-level, including CIOs, CTOs, CEOs, and other C-level roles. This structure parallels in many respects digital experience management (DEM) teams, which are also most commonly driven by executive IT. This doesn't mean that all innovation has to come from the executive suite, however. It may well come from middle level management or even versatile, technology-savvy professionals. But the IT executive suite should be involved, aware, and supportive if you're going to make Operations more actively and strategically involved in DevOps.

REAL-WORLD EXAMPLE

A large financial services institution in North America had an executive Vice President who headed up its sales and services organization. She attended one of the quarterly review sessions and came there deliberately to shake things up. She was extremely blunt about the fact that the company wasn't providing the service quality that her staff needed to meet its sales targets. This was the best possible argument for prioritizing user experience for a DevOps initiative. She had statistics on how many sales happened an hour and how much money would be lost due to service degradation. It was probably the most powerful data set ever seen on the impact of IT to a business outcome.

Mr. Fix-It versus a more strategic operations role

Depending on how your operations team is organized, you'll want to align processes such as change and release management as well as availability and performance management with DevOps needs. A short list of how integrated Operations/ITSM teams work with Development in DevOps initiatives should help to shed light on skillsets required:

- Minimizing the time spent by Development in production-level troubleshooting
- Assessing the quality of user experience from pre-production into production and beyond
- Managing infrastructure changes in support of application release needs
- Creating common workflows with Development teams as they move into production
- Scheduling releases for entry into production
- Establishing active feedback loops with Development on application/service quality
- Establishing active feedback loops with Development on application usage and application requirements
- Supporting Development actively in provisioning pre-production environments via effective service modeling and automation technologies
- Providing visibility into the events via the centralized operations team
- Understanding the availability of the SMEs from the development teams when the problem or issues are in the environment
- Ensuring that runbooks are available to development teams to solve simple problems — as appropriate

DevOps Team Leadership

Even with the advent of new more self-sufficient technologies such as containers and microservices, most business applications

will continue to depend on complex infrastructure interdependencies as they traverse geographies, interact with a wide variety of data sources and endpoints, and feed off a growing variety of supporting application capabilities — many of which may come from third parties. While Development needs to be more empowered, more fluid and faster in how it works, it also needs to be free of worrying about all the vagaries of a truly global infrastructure. That's clearly one area where Operations needs to take the lead — including feedback to Development if/when real-world conditions and application designs don't effectively coalesce.

THE LIST OF WHO MAY BE INVOLVED

A list of possible stakeholders for both DEM and effective DevOps teams coincides nicely and should help to round out your plans for putting the Ops into DevOps. The list of who may be involved in your DevOps initiative includes

- Applications support/management
- Network operations
- System/data center operations
- Cloud management (dedicated team)
- ITSM
- Cross-domain service management team
- Q/A Test
- Center of excellence
- Architecture/engineering
- Desktop/mobile support
- LoB stakeholders
- E-commerce, e-business stakeholders, and digital marketing
- Process management and compliance
- Digital or user experience management team
- Development

But Operations can take a lead in another area, and that is helping Development understand the real-world usage and value of its creations. Feedback loops on application quality, usage, and value belong to an engaged and integrated operations team, as well as business stakeholders and application/service consumers — hence the role of DEM in spearheading this more strategic role for Operations.

Looking at the Value of Processes and Best Practices

Putting the Ops into DevOps may also require juxtaposing multiple traditions of best practices in creative ways. Top-of-mind relevant best practices for your consideration should include the following:

- **Agile/Scrum:** Increasingly favored in Development, agile is more of a movement than a pure playset of best practices. Rather than following traditional forms of project management, agile is most effectively directed at modern applications made up of many small components because it features faster, shorter processes and working rhythms designed to promote immediate dialog and feedback on results. This is, for instance, in contrast to the traditional waterfall processes associated with application design. Scrum is a methodology or framework closely associated with agile computing, targeting teams of about ten people with two-week development cycles or sprints. Scrum requires solid levels of communication and isn't always optimized when teams are separated geographically or when complex or legacy applications require a large amount of regression testing.
- **IT Infrastructure Library (ITIL):** ITIL is often viewed as inappropriate for the agile need for speed with its change advisory boards (CABs) and its focus on careful, deliberate review processes. But ITIL can still be a highly useful, and sometimes even a visionary, resource in assessing how broader IT processes need to come together in context with collective decision making and shared ways of working. The most effective ITIL implementations leverage it not as a

standard (which it isn't) but as a high-level assessment of what processes need to be in place for the working of IT as a whole — including Operations and Development. Putting the Ops into DevOps in many environments may require a lane for fast track applications more centered in Scrum and a more ITIL-centric set of processes for more traditional applications.

- **IT4IT:** IT4IT is a reference architecture from the Open Group designed to create a model of IT functions in support of running IT more effectively as a business. It complements ITIL and other best practices and addresses specific use cases, such as agile, DevOps, cloud-sourcing, and service brokering. IT4IT is gaining in popularity as IT organizations increasingly find themselves on the spot to show their relevance to business outcomes, as well as transparency in demonstrating their own business efficiencies.
- **The long list:** The choices for best practices relevant to DevOps are as follows:
  - *IT Balanced Scorecard,* for better aligning team or organizational performance with strategic business or IT outcomes
  - *Six Sigma,* which is directed at improving IT or business processes by gradually reducing the chance of errors or risk
  - *COBIT* (Control Objectives for Information and related Technology), a framework for IT management and governance
  - *TOGAF* (The Open Group Architecture Framework), which is a framework for enterprise architecture, including approaches for designing, planning, implementing, and governing an enterprise information architecture
  - *CMMI* (Capability Maturity Model Integration), a training program to improve the effectiveness of IT processes administered by the CMMI Institute, and first developed at Carnegie Mellon

As annoying as it may seem to have to align with someone else's pre-established recommendations — at least as departure points — best practices consistently show value both for DevOps and for digital transformation initiatives as a whole. Among these, perhaps the most prevalent is improved IT productivity. But other

substantial benefits include improved quality of IT services, including application services; improved IT services in terms of business relevance — of doing what counts most to business outcomes; and perhaps most important here — improved agility and flexibility in provisioning services more responsively. Yet other benefits target reduced IT costs and better collaboration within and among IT groups.

REMEMBER

All this is directly relevant to successful DevOps and a chance to ensure that a solid handshake is achieved between Development and Operations as both groups move forward in a more unified and consistent fashion.

The Need for Dialog

Even best practices can only go so far if both Operations and Development are lost in their own silos of insularity, with professionals determined not to share their tools, their data, and even what they know to be true when it comes to the impacts of changes on critical application delivery. Dialog (communication) is featured in Scrum. But it's critical for virtually all DevOps initiatives. This is where good technology, such as analytics and service modeling, can help provide a common frame of reference, on the one hand, and where executive leadership in breaking through the barriers of bad habits and past convention, on the other hand, may both play a critical role.

Collaboration and communication between the operations and DevOps teams is a *must* when problems arise in the production environment. The rise of overlay teams further underscores the need for dialog, even as it changes the DevOps game. We cover overlay teams in Chapter 5.

IN THIS CHAPTER

- Understanding the growing role of overlay teams
- Optimizing systems of record with systems of engagement
- Measuring DevOps success
- Looking at DevOps in action

Chapter 5
Shifting Left

Shifting Left refers to an approach to software development where testing is performed earlier in the application life cycle. It might also be interpreted metaphorically in terms of shifting from the right-hand/traditional way of working to the left-hand/more agile way of working.

And indeed, the winds are shifting in DevOps and agile in so many directions at once that it's hard to keep track. And of course a lot depends on whom you ask as to where they're likely to blow in the future. Some will settle into technologies like microservices, containers, and API-driven applications as redefining DevOps. Others will focus on Scrum requirements for faster time windows for development. And yet others will decry the lack of an overarching schema for understanding how and where all the processes and workflows can be brought together in a consistent way across all application types and needs. But of course all these are pieces of only parts of the bigger puzzle — one that's assembling and reassembling itself even as this text is being written.

Without trying to predict the whole future of DevOps, this chapter begins with an assessment of the very visible trend toward overlay teams in DevOps and associated areas. You also look at key metrics for DevOps in terms of measuring application quality and relevance and promoting continuous evaluations for improvement. The chapter closes with advice taken from research and interviews relevant to DevOps.

What's Really Going On with Overlay Teams?

Overlay teams help empower smaller groups with more superior levels of communication to do more with less. Overlay teams for DevOps are part of a growing trend for overlay teams in multiple areas within IT — but especially when it comes to in-house developed applications. Most relevant, perhaps, are overlay teams associated with digital experience management (DEM) that often consist of many of the same key players as DevOps teams, including business stakeholders, but with a shift (left or right) toward Operations. DEM teams should provide insight into application requirements but aren't centered — where often 50 percent of application spending goes — in the blueprint stage of application development. Other related overlay teams involve service delivery or cross-functional service management teams.

DevOps overlay teams are more likely to be multiple in nature than their DEM counterparts and more likely to include more Development stakeholders in percentage to operations.

Nevertheless, the move to overlay follows many of the rules outlined for DevOps teams overall. There should still be a strong executive connection, both at the business and the IT level, even if executives aren't formally a part of an individual team. There should still be a strong operations presence, and there should still be ongoing, real-time insights into availability, service quality, and user and customer experience, as well as insights fed back to the team on usage, value, business impact, and relevance. In other words, the optimal approach to DevOps overlays isn't to work in an isolated vacuum but to have clear processes and handshakes with DEM, service delivery, and application portfolio planning-related dialogs.

From an Operations perspective, as well as from the perspective of IT as a whole, this means that overlay teams need to be nested within a larger set of dialogs and processes, even for fast-track applications that may bypass some of the traditional terms and conditions of release management.

Indeed, in the ideal DevOps situation, the world of Jira for issue and project tracking in Development needs to join hands with IT workflows coming through Operations and IT service management

(ITSM) teams. This can be made possible in large part by leveraging advanced technologies such as automation, analytics, and DEM capabilities to help marry consistency, relevance, and quality with speed.

Bi-Modal IT: Systems of Record and Systems of Engagement

One approach to joining hands between (left-handed) overlay teams and (right-handed) core teams, as well as between more traditional ways of working and faster-paced, short-windows for cloud-native and other applications, is the concept of Bi-Modal IT. Bi-Modal IT defines a traditional mode and an agile mode as parallel ways of working that, in the end, should be brought together more cohesively within a larger plan. *Traditional mode* is targeted primarily at on-premises and cloud-enabled applications, often involving longer and more complex DevOps cycles, sometimes measured in years, with a more centralized governance system. The traditional mode prioritizes operational excellence. *Agile mode* is directed primarily at cloud-native applications with two-to-three month windows for development, higher rates of change, managed typically as many small projects with a more decentralized approach to governance. The Agile mode prioritizes transformation and differentiation.

Associated with the traditional mode are systems of record, including classic approaches to change management such as configuration management systems (CMSs), and ITIL best practices. Associated with the agile mode are systems of engagement directed at speed and end-user access through mobile and API-driven applications options, and aligned, not surprisingly, with Scrum and agile methodologies.

Measuring DevOps Success

Speed is always important. But DevOps success also needs to be understood in terms of improved operational efficiencies and application service quality — no matter whether your focus is in a traditional or in an agile mode, or in a hub-and-spoke approach that can accommodate and reconcile both.

Some of the more critical speed and efficiency metrics associated with DevOps are

- Acceleration in the number of releases or software packages (files and associated information) delivered
- Acceleration in the frequency of releases or software packages delivered
- Acceleration in the lines of code delivered
- Reductions in defect counts
- Reductions in the number of Development-related backlogs (software still waiting to be delivered to Operations)
- Reductions in Operations-related backlogs (software with Operations still waiting to be fully delivered into production)
- Reductions in incidents and other adverse effects caused by new code
- Efficiencies in resolving the issues without involving all subject matter experts, achieved in part through defining automations
- Efficiencies in defining the security model for the applications during the development phases — leveraging technologies such as application scans, web scans

Some technical metrics associated with application and hence DevOps quality are

- Availability (continuous availability is the desired answer)
- Response time per transaction
- Response time per multiple transactions to complete a business service
- Response time of third-party components interacting with in-house developed applications
- Dropped transactions
- Navigability — do users really understand how to use their applications?
- Security-related risk minimization
- Any associated service level agreements (SLAs) of a technical nature

- Any associated social media or ITSM-related user feedback on application quality
- Mean time to resolution, understanding how the mean time to repair (MTTR) can be improved
- Mean time to respond, understanding when an issue is owned by a person

Some of the more important business metrics associated with application value, relevance, and costs are

- Usage and relevance — who's using a given application and why?
- User productivity — number of transactions completed
- Revenue-related metrics associated with a given business application
- Business activity metrics — the impacts of a given application on business processes
- Brand impact metrics — the impact of an application on brand value, including conversions from competitive websites
- Overall IT (Operations and Development) operational savings or OpEx efficiencies
- Social media feedback on business-related outcomes associated with an application
- Any business-related SLAs (those reflective of cost or generated business value)

DevOps In Action

IT stakeholders in Operations and overlay teams seek to work more effectively with their development counterparts. In this section, we give you some real-world examples of effective DevOps collaboration.

Healthcare

A healthcare-related company wanted to make sure it benchmarked before production. When it built a trial, it wanted to do load testing and benchmarking during the Q/A Test. This gave the company metrics to measure and a better understanding of what would happen with metrics and performance when it rolled into production.

Financial services

An overlay team in a large financial services enterprise reported degradations and other performance issues to the Vice Presidents of both Operations and Development. It helped that they were familiar with the management structures in both organizations. They had about 75,000 staff members across the world including many silos of technology. By bridging these broader organizations they were able to improve DevOps results in terms of application performance and DevOps efficiencies.

E-commerce

Sharing appropriate data with different stakeholders is key to DevOps success and can help to put Operations in a more strategic role. In an e-commerce company, Operations first shared real-time feeds for success and failure from synthetic transactions to everyone involved in real-time monitoring. Then they shared broader availability and performance reporting via reports to development, customers, the CIO and the data warehouse. Third, the data was used to confirm compliance with contracted service level agreements, which were also shared with the company's finance group and the client for evaluation.

Medical supply

The lead in an overlay team for DevOps in a medical supply company gave the following advice to stakeholders in both Operations and Development: "Implementation is not just about tools — remember to include the details such as communication processes when planning your DevOps activities. You should also involve the user community sooner and get their buy-in on what they need in the way of the applications they use, even if it might slow down your initiative."

IN THIS CHAPTER

- Understanding the changing roles of Operations and Development
- Supporting digital experience management
- Looking at analytics and metrics
- Knowing the importance of communication

Chapter 6
Ten Best Practices for Putting the Ops into DevOps

Putting the Ops into DevOps requires attention to organization, process, technology, and appropriate goal setting and metrics. The best practices in this chapter are meant to help you plan and go forward with your DevOps initiative.

Recognize the Changing Role of Operations

For Operations to become more centrally involved in DevOps, it must be prepared to evolve to run IT more like a business, whether through Line of Business (LoB)-centric teams or through a central IT operations team, or both. With commitments to digital experience management (DEM), integrated IT service management (ITSM) teams, and accelerated use of advanced technologies, Operations can not only provide DevOps teams with superior levels of remediation but also deliver real-world insights on application usage, relevance, value, and cost.

In parallel, Operations should also help development teams understand ongoing concerns with application/infrastructure performance. This can be achieved in part through shared visibility into events, time series data, and shared analytics to support more informed and role-aware decision making. Finally, Operations needs to be flexible, which means empowering overlay DevOps teams to resolve issues on their own — without involving central IT Operations.

Recognize the Changing Role of Development

Development teams are also evolving to become faster, more agile, and in some cases, far more efficient. But this trend isn't monumental. Many development teams today face a mix of cloud-native applications and traditional applications with a vast array of different interdependencies, time windows for development, and end-user requirements. Operations teams must be prepared to diversify in the face of this complex industry shift toward a rainbow of options, rather than one single color. This shift also requires shared insights into toolsets, with consistent data insights in order to help first responders resolve issues more proactively — ideally in advance of an actual service outage.

Support DEM

Perhaps nothing could be more central to putting the Ops into DevOps than a commitment and an investment in DEM — including both customer experience and internal end-user experience. DEM teams help lead a path to successful DevOps initiatives by providing ongoing insight into application quality in pre- and post-production. DEM teams also provide critical feedback loops to Development once applications are in production on real-user experience, application relevance, application design, and application impact.

Reevaluate Your Options for Automation and Service Modeling

TIP

Automation and service modeling are both areas of industry innovation critical to DevOps initiatives (see Chapter 3 for more info). Look for an integrated strategy for automation that can link pre-production and post-production requirements, and seek out innovations in service modeling and discovery that can offer fully dynamic insights to the most changing and changeable application environments.

Don't Forget Analytics

If there is a central technology platform for putting the Ops into DevOps, it's Advanced IT Analytics, or more commonly, Operations Analytics. Metrics from many different tools need to be ingested into the analytics tool to identify the anomalies. Such a platform can become a unifier both in data collection and presentation, as well as into providing proactive and predictive insights into application performance and vulnerabilities, change impacts, and even trends related to business outcomes central to optimizing DevOps.

Seek the Right Organizational Model

No two IT organizations are the same, so there's no generic answer for how Operations should be organized or even how DevOps teams should be organized. Generally though, dedicated teams do better than ad-hoc teams, and overlay teams crossing traditional silos are clearly on the rise for good reason. One answer in larger organizations when there are multiple DevOps directions, such as a separate "Fast Track" for certain applications, is a hub-and-spoke model, in which an ongoing, dedicated center can support multiple "team spokes" to the DevOps wheel.

Invest in Best Practices and Pay Attention to Metrics

Best practices, whether those associated with agile computing per se or those associated with more traditional ways of working, can

provide real value to DevOps teams. The best way to articulate that value is with shared metrics to help you evaluate speed and operational efficiency, service quality, and business impact and relevance.

Don't Neglect the Importance of Dialog and Communication

DevOps teams thrive on sharing critical insights, processes, and values. While automation and analytic insights can minimize time haggling over remediation and other activities, in the end, they should be viewed as a platform to enable more informed and strategic dialog between Operations and Development.

Begin in the Right Place

Find the most relevant and ready place for you to begin putting the Ops into DevOps with renewed vision and vitality. The first step may be a heightened commitment to DEM, making the move toward a hub-dedicated overlay team, or focusing more on a sit-down with Operations and Development to compare notes. You might even target a new technology adoption as a platform for going forward. Make sure you target just a handful of select applications to start with and build from there.

Avoid the Temptation to Do Everything at Once

If you've read this book in its entirety, you know there are a lot of ideas packed into these short pages. And that was the goal — to get you thinking about DevOps requirements from multiple perspectives. But don't go out and try to conquer the world on a long weekend or even an extended summer break. Recognize the need for a multi-dimensional approach to DevOps, but take the time to build toward it and socialize the idea with all relevant stakeholders.

Redefine Operations' role in supporting DevOps

The world of DevOps and agile is changing dramatically. But operations teams are evolving in parallel. This book helps you optimize these two trends in support of improved digital experience with the right attention to process, technology, and organizational leadership.

Inside...

- The changing role of Operations
- DevOps and continuous delivery
- Save the day with new technologies
- Discover how to lead and enable
- Understand how to measure success
- Learn from real-world examples

Dennis Nils Drogseth, a VP with Enterprise Management Associates, is a thought leader in advanced IT analytics, service modeling, digital transformation, and IT governance. **Sudhakar V. Chellam** is a Senior Software Engineer with IBM and a senior architect for ITSM solutions focused on operation insights.

ISBN: 978-1-119-46340-5
Part #: KUM12388USEN-00
Not for resale

WILEY END USER LICENSE AGREEMENT

Go to www.wiley.com/go/eula to access Wiley’s ebook EULA.