



DevOps: The IBM approach

Continuous delivery of software-driven innovation

Contents

- 2 The move to DevOps
 - 3 DevOps for enterprise organizations
 - 4 The DevOps solution: What is needed?
 - 5 The IBM solution to enable DevOps
 - 6 DevOps adoption paths and related practices
 - 7 Benefits of the IBM DevOps solution
 - 7 An IBM case study
 - 7 Client Case studies
 - 7 Conclusion
 - 8 For more information
-

Market shifts are fundamentally changing the way businesses are approaching software. In a [new study](#) from the IBM Institute for Business Value (IBV), more than 400 business executives identified five top software trends that will impact the competitiveness of their enterprises in the next five years: 1) proliferation of mobile devices, 2) explosion of unstructured data, 3) need to collaborate across value chains, 4) cloud platforms, and 5) intelligent/connected devices. Other trends include:

- Growing use of systems of engagement: The transition from current enterprise systems designed around discrete pieces of information (“records”) to systems which are more decentralized and incorporate technologies that encourage peer interactions.
- Exponential increase of empowered users and expectations for higher quality experience due to consumerization of IT: 75 percent of the companies surveyed in the IBM study expect customer experience to be a key differentiator.
- Increase in multi-sourcing: CIOs and application managers will abandon or retire 40 percent of the applications supported with PaaS, SaaS, crowdsourcing, and other sourcing models.
- Increasingly volatile economic and changing regulatory environment: The US Code of Federal Regulations alone has grown more than 7 percent (in the last four years), a pace that is outstripping economic growth.



Of the companies surveyed who are able to leverage software development and delivery for competitive advantage, almost 70 percent outperform their peers on profitability. But while most enterprises feel that software development and delivery is critical, only 25 percent believe that their teams leverage it effectively today. This “execution gap”—the difference between the need to deliver software effectively and the ability to do so—is causing missed business opportunities for the vast majority of companies surveyed.

Businesses are under tremendous pressure to create new value for their customers—with an increasing proportion of business innovation delivered through software—yet they are finding that traditional approaches to software development and delivery are not sufficient. Manual processes are error prone, break down, create waste, and delay response to business need. Technology by itself does not offer competitive advantage; however, delivery of technology-based innovation can be a competitive differentiator and, when sustained over time, a core competency.

Sustained innovation means continuously developing new ideas into innovative software, which in turn continuously improves the value delivered to users.

We believe one significant means for achieving this sustained innovation is **DevOps**—a contraction of “development and operations,” the two teams that form the core of an organization’s technology department. In its conventional sense, DevOps refers to a closer collaboration of these two teams, the integration of associated processes and tooling. But from IBM’s point of view, DevOps is much more. The spirit of DevOps is expanded collaboration among all stakeholders, not just between development and operations, but also among lines of business, suppliers involved in software delivery, and consumers themselves. In this expanded sense, DevOps includes business governance practices around security and compliance, and all aspects of the delivery process such as multi-sourcing.

DevOps is an essential enterprise capability for continuous software delivery that enables organizations to seize market opportunities and reduce time to customer feedback.

A DevOps approach applies agile and lean thinking principles to all stakeholders in an organization who develop, operate, or benefit from the business’s software systems, including customers, suppliers partners. By extending lean principles across the entire software supply chain, DevOps capabilities will improve productivity through accelerated customer feedback cycles, unified measurements and collaboration across an enterprise, and reduced overhead, duplication, and rework. And it offers competitive advantage to a business through three dynamic capabilities:

- Speeding **continuous innovation** of ideas by enabling collaborative development and testing across the value chain
- Enabling **continuous delivery** of these innovations by automating software delivery processes and eliminating waste while still helping to meet regulatory concerns
- Providing a feedback loop for **continuous learning** from customers by monitoring and optimizing the software driven innovation

The move to DevOps

The DevOps approach was pioneered by smaller, lean-thinking, “born on the web” companies seeking just-in-time solutions for rapid and frequent delivery of software, and to obtain and respond to customer feedback. The approach is used extensively for “systems of engagement”—applications typically powered by cloud, mobile, and social media technologies—for expanding customer outreach and enabling an increasingly mobile workforce through frequent update and release. To obtain the benefits of continuous innovation via systems of engagement,

forward-thinking enterprises are moving to a DevOps approach to continuous delivery. But in that move, they are seeing significant sources of friction between their new systems of engagement and their existing systems of record:¹

- Systems of record are largely geared toward providing information to a company's workers, maintaining business stability and integrity, meeting service level agreements, compliance, and security needs. Whether it is ERP, supply chains, partner eco-systems, HR systems, etc., for a system of engagement to function as a cohesive whole, the front office and back office technologies need to be integrated.
- Development teams working on new systems may have adopted agile and iterative software delivery methods that allow them to release new software much faster than operations teams can deploy them, or business users can adjust to them. On the other hand, development teams working on back office applications via traditional waterfall methods may be unable to respond soon enough to emerging business requirements for their output to yield any benefits.
- Development teams are measured by their ability to change rapidly—that is, embrace *agility*. Operations teams are measured by their ability to maintain stability and 100 percent uptime. These two organizations, when managed separately, have competing goals and the collaboration between them can be adversarial.

In order to adopt the DevOps approach and integrate systems of record, as well as offer new systems to support enhanced collaboration across the organization and value chain—which is what systems of engagement is all about—appropriate DevOps foundation technologies and practices will need to be deployed across the application lifecycle. A successful strategy recognizes that DevOps benefits are shared by business and client, and to that end all teams—front office, development,

operations, and customers—must share the commitment to automation, delivery, and feedback on results. When rapid delivery and response to consumer demand is the goal, all teams must optimize and automate; otherwise, the least efficient team determines the effectiveness of the entire chain.

DevOps for enterprise organizations

Quite naturally, the most important capability for an enterprise organization is the transformation of a business idea into a high-quality capability that customers will perceive as attractive, with a mechanism for rapid customer feedback on these new capabilities. At the same time, they must be able to transform their organization to bring agility beyond technology teams to connect to their customers and maintain stability within their systems of record applications.

The value of DevOps can be illustrated as an innovation and delivery lifecycle, with a continuous feedback loop to learn and respond to customer needs. Some of the primary challenges that the business must address in this critical enterprise capability for innovation and delivery of software based value are:

- What a user might find valuable is unknown until the business actually delivers the software in a true production environment
- There is a minimum of 4 - 6 weeks to implement and deliver a change to a customer
- The larger the enterprise, the greater the lack of collaboration among different groups (LOB, Dev, Ops, suppliers, etc.), each with their own processes, tools, and infrastructure
- Manual processes driven by tribal (not well-documented) knowledge add delays to all involved in the process
- Lack of measurements/metrics prevents a business from obtaining feedback on a continuous basis
- Contradictory goals: Development teams motivated by rapid change and Operations teams are motivated to keep the environment stable.

By incorporating lean manufacturing principles and lean thinking (where anything that does not create value or provide benefit for a customer is considered waste) across the enterprise, DevOps can address these challenges. Advances in agile methods—which may be considered the *software* version of lean manufacturing—have contributed greatly in the move toward DevOps, where results are prioritized over processes. “Just in time” thinking reduces the overhead of low-value reporting, traceability, inspections, and other traditionally minded quality assurance activities. False precision in specifications and plans is replaced with lean thinking and agile methods that emphasize incremental iterations/batch sizes, just-in-time and Kanban production, and an acceleration of cycle times, all while leveraging the creativity of knowledge workers. Lean thinking differentiates between value-creating activities and waste, and focuses on resolving the bigger uncertainties, like architecturally significant decisions and integration testing earlier in the process.

DevOps incorporates lean thinking and agile methodology as follows:

- Eliminate any activity that is not necessary for learning what customers want. This emphasizes fast, continuous iterations and customer insight with a feedback loop.
- Eliminate wait times and delays caused by manual processes and reliance on tribal knowledge.
- Enable knowledge workers, business analysts, developers, testers, and other domain experts to focus on creative activities (not procedural activities) that help sustain innovation, and avoid expensive and dangerous organization and technology “resets.”
- Optimize risk management by steering with meaningful delivery analytics that illuminate validated learning by reducing uncertainty in ways that can be measured.

As shown in the recent IBV study, advanced organizations are more likely to apply the above practices, such as agile and continuous delivery. In fact, more than twice as many advanced organizations apply these sophisticated practices consistently compared to intermediate and foundational organizations, as shown in Figure 1.

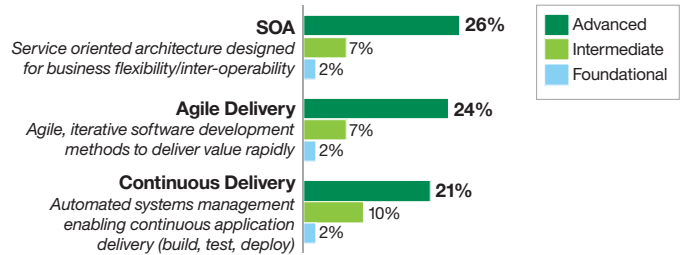


Figure 1. More than twice as many advanced organizations consistently apply SOA, agile, and continuous delivery practices compared to Intermediate and foundational organizations.

The DevOps solution: What is needed?

The following elements are essential to an effective continuous innovation and continuous delivery lifecycle:

- Use a common set of practices based on lean thinking to maximize value through collaborative development and continuous testing (enabling continuous innovation), and eliminate any activity that does not directly benefit the customer.
- Automate manual and overhead activities (enabling continuous delivery) such as change propagation and orchestration, traceability, measurement, progress reporting, etc.
- Create a closed loop feedback mechanism (enabling continuous learning) in a customer facing environment
- Use meaningful measurement and monitoring of progress (enabling continuous optimization) for improved visibility across the organization, including the software value delivery supply chain.

Promote enterprise-wide stakeholder collaboration that brings together and fully leverages existing development and operations teams, applications and platforms across the enterprise.

These requirements and their relationships are illustrated in Figure 2.

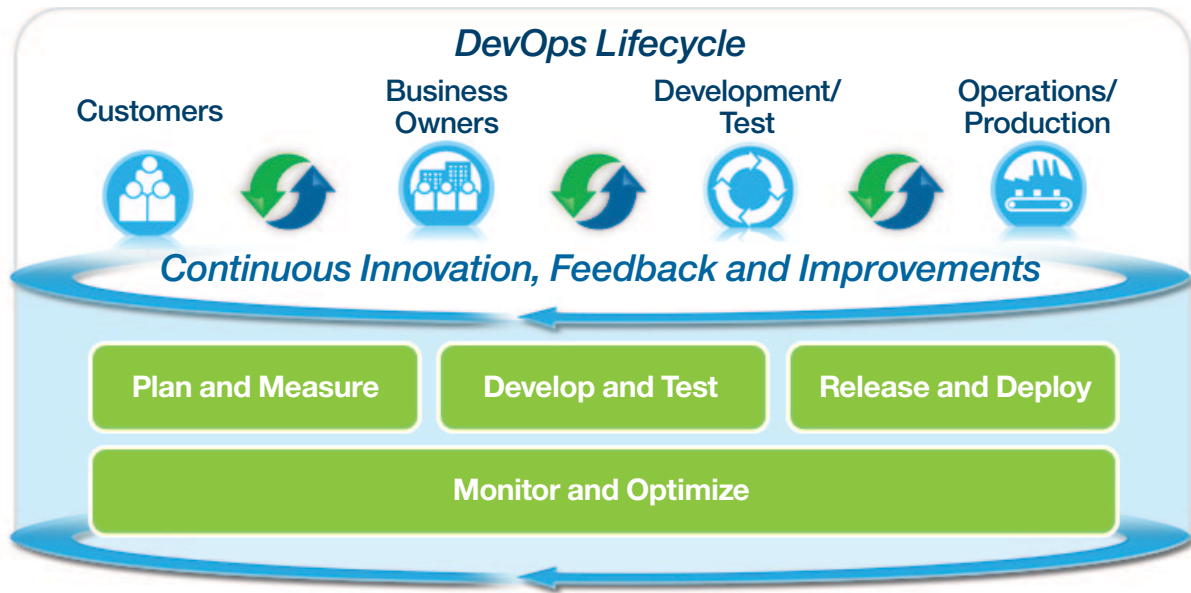


Figure 2. The continuous innovation and continuous delivery lifecycle, enabled by DevOps.

The IBM solution to enable DevOps

IBM provides an open, standards-based DevOps platform that supports a *continuous innovation, feedback and improvement lifecycle*, enabling a business to plan, track, manage, and automate all aspects of continuously delivering business ideas. At the same time, the business is able to manage both existing and new workloads in enterprise-class systems and open the door to innovation with cloud and mobile solutions. This capability includes an iterative set of quality checks and verification phases that each product or piece of application code must pass before release to customers. The IBM solution provides a continuous feedback loop for all aspects of the delivery process (e.g., customer experience and sentiments, quality metrics,

service level agreements, and environment data) and enables continuous testing of ideas and capabilities with end users in a customer facing environment.

IBM's DevOps solution consists of the open standards based platform, DevOps Foundation services, with end to end DevOps lifecycle capabilities. To accommodate varying levels of maturity within an IT team's delivery processes, IBM has grouped the capabilities into four adoption paths as shown in the green rectangles in Figure 2 above.

DevOps adoption paths and related practices

Plan and measure

This adoption path consists of one major practice:

Continuous business planning: Continuous business planning employs lean principles to start small by identifying the outcomes and resources needed to test the business vision/value, to adapt and adjust continually, measure actual progress and learn what customers really want and shift direction with agility and update the plan. ibm.com/software/products/us/en/ratificapoin/

Develop and Test: This adoption path consists of two major practices:

Collaborative development: Collaborative development enables collaboration between business, development, and QA organizations—including contractors and vendors in outsourced projects spread across time zones—to deliver innovative, quality software continuously. This includes support for polyglot programming and support multiplatform development, elaboration of ideas, and creation of user stories complete with cross-team change and lifecycle management. Collaborative development includes the practice of continuous integration, which promotes frequent team integrations and automatic builds. By integrating the system more frequently, integration issues are identified earlier when they are easier to fix, and the overall integration effort is reduced via continuous feedback as the project shows constant and demonstrable progress. ibm.com/software/rational/alm/collaborate/

Continuous testing: Continuous testing reduces the cost of testing while helping development teams balance quality and speed. It eliminates testing bottlenecks through virtualized dependent services, and it simplifies the creation of virtualized test environments that can be easily deployed, shared,

and updated as systems change. These capabilities reduce the cost of provisioning and maintaining test environments and shorten test cycle times by allowing integration testing earlier in life cycle.

Release and deploy: This adoption path consists of one major practice:

Continuous release and deployment: Continuous release and deployment provides a continuous delivery pipeline that automates deployments to test and production environments. It reduces the amount of manual labor, resource wait-time, and rework by means of push-button deployments that allow higher frequency of releases, reduced errors, and end-to-end transparency for compliance. ibm.com/software/products/us/en/continuousdelivery/

Monitor and optimize: This adoption path consists of two major practices:

Continuous monitoring: Continuous monitoring offers enterprise-class, easy-to-use reporting that helps developers and testers understand the performance and availability of their application, even before it is deployed to production. The early feedback provided by continuous monitoring is vital for lowering the cost of errors and change, and for steering projects toward successful completion. ibm.com/software/tivoli/products/smartcloud-monitoring/

Continuous customer feedback and optimization:

Continuous customer feedback provides the visual evidence and full context for analyzing customer behavior and pinpointing customer pain points. Feedback can be applied during both pre- and post-production phases to maximize the value of every customer visit and ensure that more transactions are completed successfully. This allows immediate visibility into the sources of customer struggles that affect their behavior and impact business. <http://www.tealeaf.com/products/customer-behavior-analysis-suite/index.php>

Benefits of the IBM DevOps solution

By adopting this solution to address needs, organizations can unlock new business opportunities:

- Deliver a differentiated and engaging customer experience that builds customer loyalty and increases market share by continuously obtaining and responding to customer feedback
- Obtain fast-mover advantage to capture markets with quicker time to value based on software-based innovation, with improved predictability and success
- Increase capacity to innovate by reducing waste and rework in order to shift resources to higher value activities

An IBM case study

IBM® Rational® Collaborative Lifecycle Management (CLM) is a practice long advocated by the IBM Rational organization, based on a widely adopted suite of IBM products. As the business responsible for these products, IBM is itself faced with many of the challenges discussed in this paper. Until recently, CLM capabilities were released once a year, in what was called a “big-bang release.” However, IBM recognized that such a long delivery milestone is inadequate for meeting the fast changing market needs that characterize the business. For this reason, IBM started to move to a continuous delivery model, where the product will be released every quarter, to address those more rapidly changing market needs.

IBM's initial plans for doing more, and faster, included the addition of more people; initial estimates indicated that this accelerated delivery required additional resources of about 25 percent. Instead, IBM decided to experiment: We asked the team to deliver with the same number of resources, and left them with the daunting task of doing more, with speed!

The team evaluated the entire set of delivery requirements, including business process, development, QA, support, marketing, and sales. They also identified waste due to wait times and delays, given that the current team consisted of over 25 components teams spread over 19 locations worldwide. The team had been not only producing one major release per year,

but also several minor releases to address defects. That process required at least 4 - 6 months lead time to collect requirements, then develop, build, and deliver to Jazz.net and get business approvals.

IBM's early results have been outstanding:

- The team has now achieved the ability to deliver every 12 weeks, with 25 percent fewer resources than originally estimated
- Continuous integration and testing of builds has been compressed to 2 - 3 weeks by automating deployment of test environments with production-like configurations that span over 15 - 20 types of environments with different databases, app servers, and LDAP to build, test, and integrate ~200 builds per day

Client Case studies

Cars.com increases product releases from 40 to 300 per year: Agile methods supported by IBM solutions enable 650 percent increase in releases. http://www-01.ibm.com/software/success/cssdb.nsf/cs/RNAE-92JMYV?OpenDocument&Site=rational&cty=en_us

Multinational company speeds time to market by 25 percent. IBM Rational Jazz technology-based agile platform cuts operating costs 25 percent. http://www-01.ibm.com/software/success/cssdb.nsf/CS/CPAR-933U37?OpenDocument&Site=rational&cty=en_us

Fidelity Worldwide Investment gains predictable release schedules with assured regulatory compliance. http://www.urbancode.com/html/resources/articles/Fidelity_Success_Story.pdf

Conclusion

Traditional approaches to software development and delivery are no longer sufficient. Manual processes are error prone, break down, and they create waste and delayed response. Businesses can't afford to focus on cost while neglecting speed of delivery, or choose speed over managing risk. A DevOps approach offers a powerful solution to these challenges.

DevOps reduces time to customer feedback, increases quality, reduces risk and cost, and unifies process, culture, and tools across the end to end lifecycle—which includes adoption paths to plan and measure, develop and test, release and deploy, and monitor and optimize.

The IBM DevOps approach helps with you incrementally adopt the solution, enabling you to speed innovation while balancing cost, quality, and risk, without tradeoffs. The IBM solution is based on open standards to allow enterprises to leverage their existing investments and allow open source lifecycle tools to coexist and interoperate. The IBM solution is also based on best practices and helps enterprises in their transition to systems of engagement, yet be able to connect to integrate existing systems of record. The solution provides a platform for integrating an organization's value chain and extended lifecycle to collaborate across a broad group of stakeholders that includes not only development, but also line of business, clients, and operations teams. What's more, the IBM approach to DevOps will enable you to continuously integrate feedback from customers and optimize processes by reducing the time to obtain feedback.

To get started on establishing DevOps as a enterprise capability for continuous software-driven innovation visit:
<http://www-01.ibm.com/software/rational/devops/>

For more information

To learn more about the IBM solution for DevOps, please contact your IBM representative or IBM Business Partner, or visit the following website: ibm.com/software/rational/devops/



© Copyright IBM Corporation 2013

IBM Corporation
Software Group
Route 100
Somers, NY 10589

Produced in the United States of America
May 2013

IBM, the IBM logo, ibm.com, and Rational are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at ibm.com/legal/copytrade.shtml

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

The client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

¹"Systems of engagement" and "systems of record" are terms recently coined by Geoffrey Moore to describe a qualitative difference in user expectation and interaction regarding modes of enterprise IT systems. His seminal paper can be found here: <http://www.aiim.org/~media/Files/AIIM%20White%20Papers/Systems-of-Engagement-Future-of-Enterprise-IT.ashx>



Please Recycle
