

DevOps for hybrid cloud: an IBM point of view

*How DevOps for hybrid cloud can help organizations
succeed with digital reinvention*



Introduction

DevOps started as a culture and set of practices to support collaboration and communication across development and operations, and to apply automation to key phases of the software delivery process. It has been popularized by successful new companies developing business models and related applications empowered by the cloud (cloud-native applications). More recently, large, established enterprises have recognized the need to deliver innovation faster to stay relevant and capitalize on industry disruption, while also improving operational metrics for application quality and cost. DevOps and cloud have emerged as essential parts of their IT strategy as they improve core competency in continuous delivery of software-driven innovation.

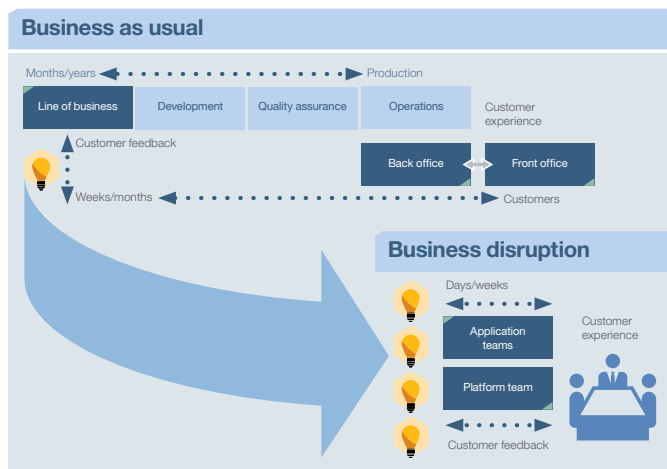


Figure 1. The ingredients for business disruption: continuous innovation, fast time-to-market, exceptional user experience, and fast time-to-feedback—pointing to the need for Continuous Delivery.

The IBM point of view on DevOps makes the following assumptions:

- DevOps covers the end-to-end software delivery lifecycle including an expanded set of stakeholders such as business owners and end users, and practices such as design thinking and user analytics.
- DevOps adoption is expanding in large organizations as they enable existing IT applications for cloud (cloud-enabled applications). New methods enable organizations to successfully implement DevOps as they move to cloud.
- Hybrid cloud architecture is becoming the norm for both cloud-enabled and cloud-native applications. Hybrid cloud provides flexibility in deployment, enabling organizations to choose the right platform to run their workloads.
- DevOps solutions can vary as teams across large organizations have different goals, processes, culture and tools.
- In cases where disparate teams work together on common business objectives, DevOps helps organizations respond to the challenges of multi-speed IT in combination with methods such as the Scaled Agile Framework environment (SAFe) to facilitate collaboration.
- Application security is essential and needs to be integrated early and throughout the DevOps lifecycle to keep pace with frequent releases and not be a bottleneck.
- Continuous delivery is the desired future-state achieved with the application of a DevOps approach.

DevOps and cloud help enterprises succeed in digital disruption

The business change driven by cloud, analytics, mobile, and social technologies is unprecedented in its speed and scope. New business models are emerging while others are becoming obsolete; born-on-the-web enterprises such as Uber, Airbnb and Spotify are among the highest-profile examples of how traditional businesses are being disrupted by new market entrants. This disruption is driven primarily by changes in customer behavior, facilitated by new technologies. As customers seek a direct, digital link with the businesses they deal with—increasingly, from their mobile devices—application developers are taking on major responsibility for the customer experience.

No industry today is untouched by this dramatic transformation; more than ever before, innovation and adaptability are critical to the survival and development of both existing businesses and startups.

In the present business environment, product and service delivery processes must be optimized for innovation and time-to-market. Methodologies including Agile, Lean Startup and Design Thinking are well-suited to supporting the design and build of truly innovative offerings, and cloud is increasingly seen as the best platform for software delivery.

Organizations are embracing approaches to software development that put the customer front and center. By increasing the frequency of software delivery and reducing the time-to-feedback from customers, they can respond faster to shifts in the market and keep customers happy.

Naturally, increased release frequency demands tighter alignment and collaboration than are seen traditionally between line of business, development, and IT operations, which drives the requirement for enhanced collaboration, automation and information transparency among these groups. To achieve this seamless internal cooperation and promote sustained innovation across the enterprise, IBM recommends the adoption of DevOps.

Defining DevOps

As an approach that promotes closer collaboration among line of business, development and IT operations, DevOps has an impact across the business. For line of business executives and CIOs, a key concern is the capability of DevOps to enable business transformation through faster development of innovative

software that meets emerging business needs—or even creates those needs in the market. For senior application development executives, the primary concern is about improving operational metrics around cost, risk, quality, productivity and speed in the development cycle. DevOps practitioners want to work on great applications, focus on delivering value, eliminating waste, automating grunt work, and making release party weekends a thing of the past.

IBM defines DevOps as an “essential enterprise capability for the continuous delivery of software-driven innovation that enables organizations to seize market opportunities and reduce time to customer feedback”. DevOps has three main objectives around business transformation which map to three objectives around IT efficiency, as shown in Table 1:

Transformation and organizational objectives	
Business-transformation objectives	IT-efficiency objectives
<ul style="list-style-type: none"> To speed continuous innovation of ideas by enabling collaborative development and testing across the value chain. To enable continuous delivery of these innovations by automating software delivery processes and eliminating waste—while still helping to address regulatory concerns. To provide a feedback loop for continuous learning from customers by monitoring and optimizing the software-driven innovation. 	<ul style="list-style-type: none"> To correct the present misalignment of people and goals by fostering closer links between developers, operations and the business. To accelerate and remove error from the delivery of changes by introducing automation throughout the development cycle. To improve insight into the real value of applications by using customer feedback to drive optimization.

Table 1. DevOps objectives

Practicing DevOps

Organizations that practice DevOps successfully tend to adopt the following processes and technologies:

Design thinking: for a focus on delivering exceptional user experience and for increasing user conversion

Lean startup: for validating ideas and testing possible solutions before committing significant manpower, helping organizations to stay focused on solving the problems that matter

Agile: as the development methodology for fast feedback cycles through early customer involvement

Cloud operations: for fast and flexible management of development and production environments

Continuous Security: for eliminating security vulnerabilities from applications before they reach production

Delivery automation: for removing the silos between development and IT operations, and enabling the continuous delivery of changes

Application Monitoring: for quickly detecting and addressing software application issues in test and production environments

Application and user analytics: for continuous learning used to improve the quality and value of applications.

Many successful startups naturally apply all of the above practices from the outset; particularly in the case of smaller organizations, it is easier for a new company to adopt a whole new approach than for an existing company to change its practices. Nevertheless, at the enterprise level, traditional IT organizations are transforming the way they work to incrementally adopt these processes and technologies.

Digital disruption is also driving associated cultural changes in the way software is delivered. The DevOps processes and technologies outlined above require a shift in mindset away from isolated development teams and rigid silos toward collaborative best practices. Typically, we see successful DevOps organizations embracing the following:

- A **work environment** that welcomes innovation, productivity, satisfaction and continuous improvement.
- Software delivery based on **autonomous co-located squads**—small, cross-functional, self-organizing teams that own end-to-end responsibility.
- A process that delivers **transparency with metrics** and embraces free and open collaboration to enable visibility, **continuous learning**, and **continuous improvements** for all stakeholders.
- An environment where developers can **fail fast** in a **penalty-free environment** where individuals are empowered to try something new and different without risk of doing damage or looking foolish and where they can turn small-scale mistakes into productive sources of future creativity.

IBM provides process, software and service offerings to enable the implementation of these practices.

IBM DevOps approach

IBM has identified six phases in the DevOps lifecycle, plus necessary cultural considerations, for successful implementation of a DevOps approach. This is based on our experience with thousands of clients as well as IBM's internal DevOps adoption as part of our own transformation journey.

IBM® Cloud Garage Method describes these phases as:

Think: Conceptualization, refinement, and prioritization of capabilities.

Code: Generation, enhancement, optimization and testing of features.

Deliver: Automated production and delivery of offerings.

Run: Services, options and capabilities required to run.

Manage: Ongoing monitoring, support, and recovery of offerings.

Learn: Continuous learning based on outcomes from experiments.

These six phases are built around a shared culture of foundational values and principles. For more information on the IBM Cloud Garage method, visit ibm.com/devops/method

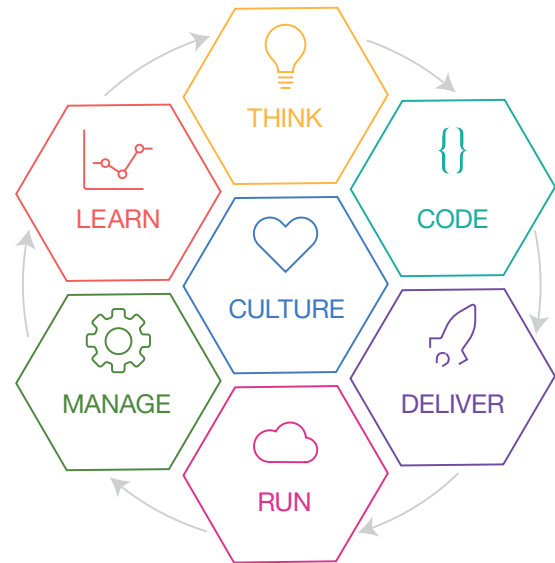


Figure 2. The six phases of the DevOps lifecycle

IBM gets it right with the Cloud Garage Method

“...IBM has done the IT industry a great service by describing the attributes and process that will help any enterprise be more successful leveraging the cloud.”

[Read the full brief from The Enterprise Strategy Group](#)

[Explore the IBM Cloud Garage Method](#)

Architectural changes with cloud

Cloud-native and cloud-enabled applications

Cloud and DevOps increase the speed at which new applications can be released and existing applications modified. Software is evolving from complex, monolithic applications, whose many dependencies are resolved at build-time, toward a more distributed, service-centric architecture whose dependencies are resolved at runtime. Whether they are cloud-enabled or cloud-native applications, these service-centric creations take advantage of the elasticity of the cloud to enable more discrete updates.

Cloud-native applications are created specifically for the cloud. They focus on engaging users across multiple touch points, including mobile platforms and social media. Cloud-native applications often utilize a microservices architecture to enable agility in change and deployment, and reuse existing web services (data management, analytics, cognitive processing, Internet of Things, and so on) to speed their development time. Typically associated with born-on-the-web organizations, cloud-native applications leverage cloud capabilities such as multi-tenancy and the automatic, elastic scaling of resources.

Cloud-enabled applications, on the other hand, are typically a combination of existing applications originally designed for a pre-cloud environment (also called “systems of record”) and new “systems of engagement” applications developed in the cloud. Their architectures tend to be complex due to their many dependencies, and they use APIs to bridge between the existing systems of record and the new systems of engagement. They leverage API management and cloud integration technologies to enable integration while addressing the organization security requirements. Their workloads may run across multiple environments: on-premises, private cloud, and public cloud—an architecture also referred to as hybrid cloud. Their architectures often dictate specific platform requirements—for example, a particular operating system or a supporting code library—and changes to these elements must be tightly controlled.

Application-level multi-tenancy and elasticity are not typically present, and these cloud-enabled applications are often associated with traditional IT organizations.

Most enterprises will need to support both cloud-native and cloud-enabled applications for the foreseeable future. The mix will gradually tilt toward cloud-native, but it will be some years before cloud-native applications become the dominant application model within most large enterprises.

Successful DevOps transformation with IBM services

IBM provides a broad scope of services including:

- [Strategic consulting with IBM Digital Reinvention services](#)
- [DevOps implementation and execution with IBM DevOps Innovation services](#)
- [Startup DNA with IBM Cloud Garage services](#)

The importance of DevSecOps

With data breaches from application-related attacks rising rapidly and representing the largest source of data loss, short-cutting or bypassing application security testing for the sake of delivery velocity is an unwise strategy. Vulnerabilities inadvertently introduced during development can give hackers the ability to destabilize applications and obtain unfettered access to confidential company information or private customer data—leading to potentially significant financial loss.

DevSecOps is the concept of integrating application security testing within a DevOps environment—this is a big process and cultural challenge because application delivery speed and release frequency are primary DevOps goals. Many organizations have overburdened security teams and DevOps can further increase their pressure if not addressed appropriately. This makes a risk-based application security management strategy, together with integrated, automated security testing early and throughout the software delivery lifecycle, essential for success.

The use of application security gates—Development, QA, and Security—enables organizations to inject automated security testing throughout the software delivery lifecycle. Static analysis is integrated into the coding and build/continuous integration phases and passes through to the Development gate. Dynamic analysis is integrated into the testing phase and passes through to the QA gate. Interactive analysis, hybrid analysis and penetration testing are integrated into the final phase, passing through to the Security gate and being released into production. By integrating tooling and security testing throughout the software delivery lifecycle (SDLC) in this manner, organizations can minimize application vulnerabilities and risk, without sacrificing delivery speed.

Ensuring Monitoring plays its part in DevOps

One of the primary goals of a DevOps approach is to speed up the release timeline and minimize slowdowns in the workload. Architectures are becoming more complex with the use of microservices and of both legacy and newly created APIs. The resulting web of interdependencies makes it more difficult to quickly determine the root cause of a problem. In addition, cloud-native companies have set customer expectations very high in terms of uptime and fast response times. Therefore, the goal that both cloud-native and cloud-enabled organizations need to work towards is determining how to resolve issues as quickly as possible and ideally to prevent them even occurring.

One way to move towards this goal is to integrate application performance monitoring into the development and test phases, as well as into the production phase. This strategy employs capabilities such as synthetic tests to determine where problems might lie before any code is actually put into production. Instead of waiting for the Ops team to surface issues when an app is in production, the Dev team can perform synthetic tests and fix the issues before anything is deployed. This new way of thinking can help move teams toward a more agile, DevOps approach and improve their ability to meet end-user expectations.

IBM Cloud Platform: The largest Cloud Foundry implementation

IBM Cloud Platform® is a hybrid cloud platform as a service (PaaS) based on Cloud Foundry, and has a flexible deployment model in public cloud, dedicated cloud and locally on customer premises. Cloud Platform is among the first offerings in the industry to be designated Cloud Foundry Certified.



It supports several programming languages and services as well as integrated DevOps to build, run, deploy and manage applications on the cloud. Cloud Platform supports instant runtimes, IBM containers and virtual machines.

More on IBM Cloud Platform can be found at ibm.com/cloud/platform

Two DevOps profiles

IBM believes that the requirement for cloud-native and cloud-enabled applications reflects the existence of two general profiles among adopters of DevOps in the cloud. Teams that fall into the cloud-native profile will have different needs and expectations from teams that fall into the cloud-enabled profile.

Cloud-native profile

While the cloud-native profile has often been associated with born-on-the-web organizations, it is increasingly found in Innovation or Line of Business groups within larger organizations. This profile is characterized by small teams working to short delivery cycles who are focused on effectiveness and user (or business) outcomes. See Figure 3.

Critical challenges for this profile of DevOps adopters are shown on the left-hand side of Table 2, with the corresponding solutions they are seeking on the right-hand side:

Cloud-native challenges and solutions	
Critical challenges	Solutions sought
<ul style="list-style-type: none"> Determining if their offerings are resonating with customers. Eliminating random defects in production that could lead to customer satisfaction issues. Focusing resources on delivering only those new features that will actually engage customers. 	<ul style="list-style-type: none"> Quickly understand what customers are doing and experiencing by implementing analytics and metrics tied to customer scenarios. Keep the application in a constant production-ready state by implementing delivery processes which fully test every check-in within minutes. Experiment with disruptive innovations, continuously assessing customer feedback to align these innovations—and the resources required to implement them—with changing customer demand.

Table 2. Critical challenges for cloud-native adopters

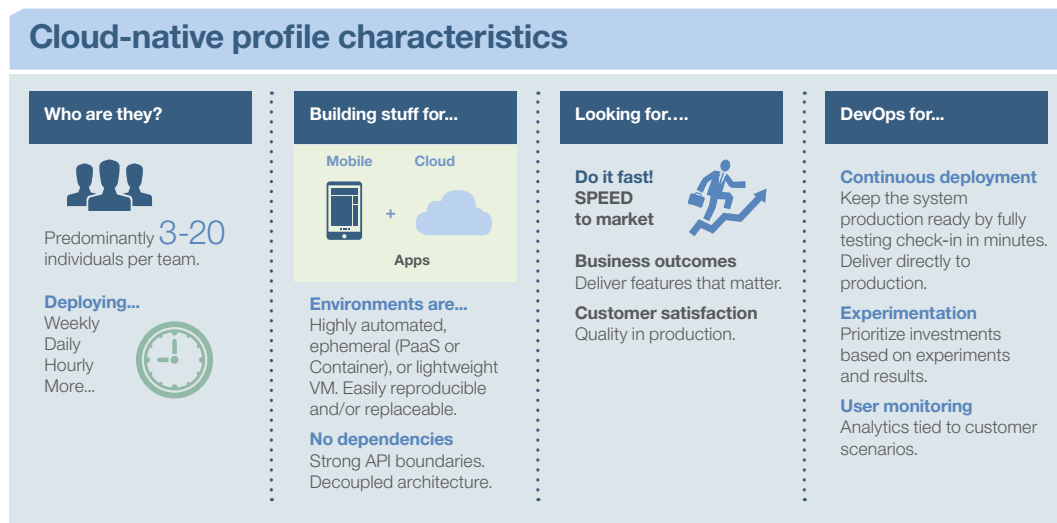


Figure 3. Cloud-native profile characteristics

Cloud-enabled profile

The cloud-enabled profile is traditionally found in IT organizations that maintain and evolve complex systems. This profile is characterized by teams of teams working to longer delivery cycles who are focused on quality improvements, faster time-to-market and balancing cost and value. Organized into teams of teams, they manage complex systems with multiple interdependent components, using a variety of technology platforms across teams. Typically, there will be not only different platforms (mainframe, mid-range, distributed, and so on) but also different generations of technology to manage—for example, certain components that are dependent on code written for a legacy platform. See Figure 4.

The cloud-enabled profile faces not only the same challenges as the cloud-native profile, but also a significantly tougher set of additional critical challenges. These are shown on the left-hand side of Table 3, with the corresponding solutions on the right-hand side:

Cloud-enabled challenges and solutions	
Critical challenges	Solutions sought
<ul style="list-style-type: none"> Accelerating and removing errors from the processes of evaluating and delivering changes. Eliminating miscommunication between teams, in order to reduce high levels of rework and waste. Reducing the excessive time spent on firefighting production incidents and downtime. 	<ul style="list-style-type: none"> Reliable, repeatable delivery to production through the implementation of highly automated and orchestrated release processes. Shorter release cycles through the implementation of small-batch release planning. Upward trends in release quality through the implementation of direct feedback loops from production monitoring systems back to the teams tasked with addressing critical issues.

Table 3. Critical challenges for cloud-enabled adopters

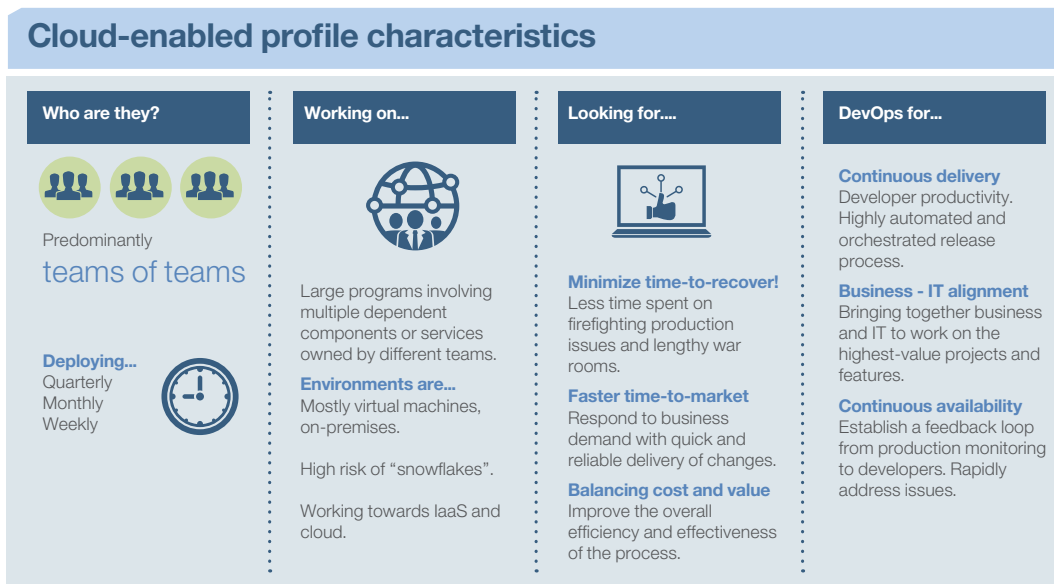


Figure 4. Cloud-enabled profile characteristics

IBM solutions for DevOps

There are three main scenarios for the adoption of a DevOps solution:

- Innovation/startup scenario: new DevOps solution is selected to address the need of cloud-native audience
- Evolutionary scenario: existing enterprise DevOps solution evolves to support both cloud-enabled and cloud-native audiences
- Blended scenario: where cloud-enabled and cloud-native teams with different cultures, processes and DevOps solutions need to collaborate together in order to deliver a complete solution to the business.

Innovation/startup scenario

This scenario can happen either:

- In a startup/line of business where teams have no dependencies on legacy enterprise systems;
- In large organizations with an “Innovation/startup mode” initiative where a set of identified “Innovation” teams are going through a disruptive transformation, with a complete new set of processes/practices and a new DevOps toolchain.

The notable feature of this scenario is that the DevOps solution is primarily targeted at addressing the needs of cloud-native teams, and there is a greater focus on adopting simpler capabilities with large adoption in the market. However it can be challenging to decide which tools to choose, and integrating and managing them can be time-consuming.

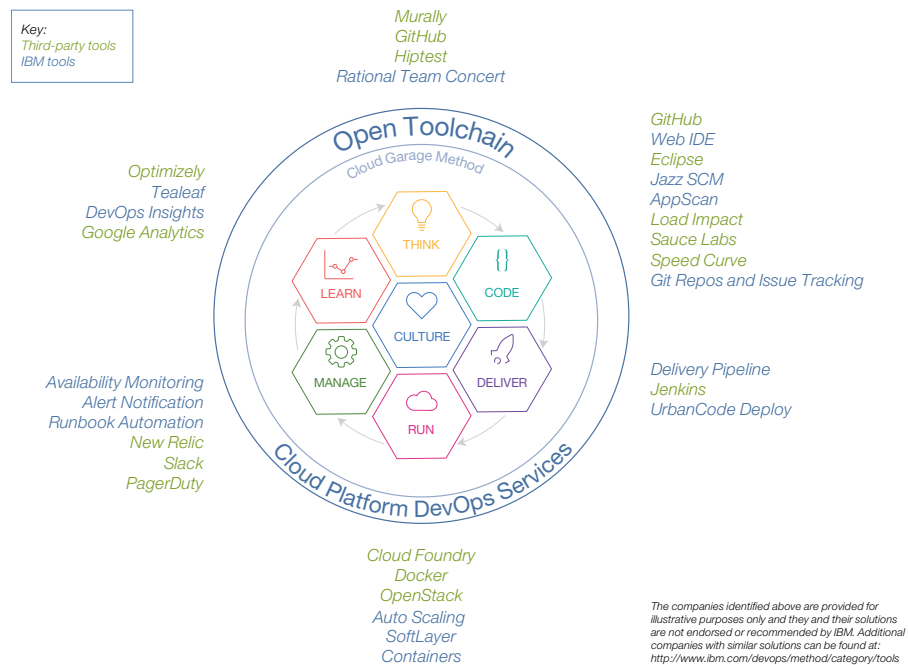


Figure 5. IBM open toolchain for innovation/startup scenario

The IBM Cloud Garage Method is IBM's approach to enable business, development and operations to continuously design, deliver, and validate new functionality. The practices, architectures, and toolchains cover the entire product lifecycle from inception through capturing and responding to customer feedback and market changes. The Open Toolchain architecture makes it easy to combine Cloud Platform services, such as Continuous Delivery, with open source and leading third-party tools into an integrated toolchain aligned with DevOps practices. These patterns can be shared between teams as templates to promote successful adoption of DevOps across an organization.

Evolutionary scenario

This scenario is typical of an evolutionary DevOps transformations as IT organizations evolve their workloads to become cloud-enabled and include cloud-native applications.

In this context the DevOps solution needs to support complex scenarios for coordinating work across multiple dependent teams, orchestrating complex deployments across hybrid cloud (IBM and third-party), and providing simple usage for the less complex cloud-native applications.

For cloud-native teams, IBM provides IBM Cloud Platform available on premises, in dedicated clouds and in the public cloud. IBM offers both on-premises and managed DevOps solutions to support such a scenario.

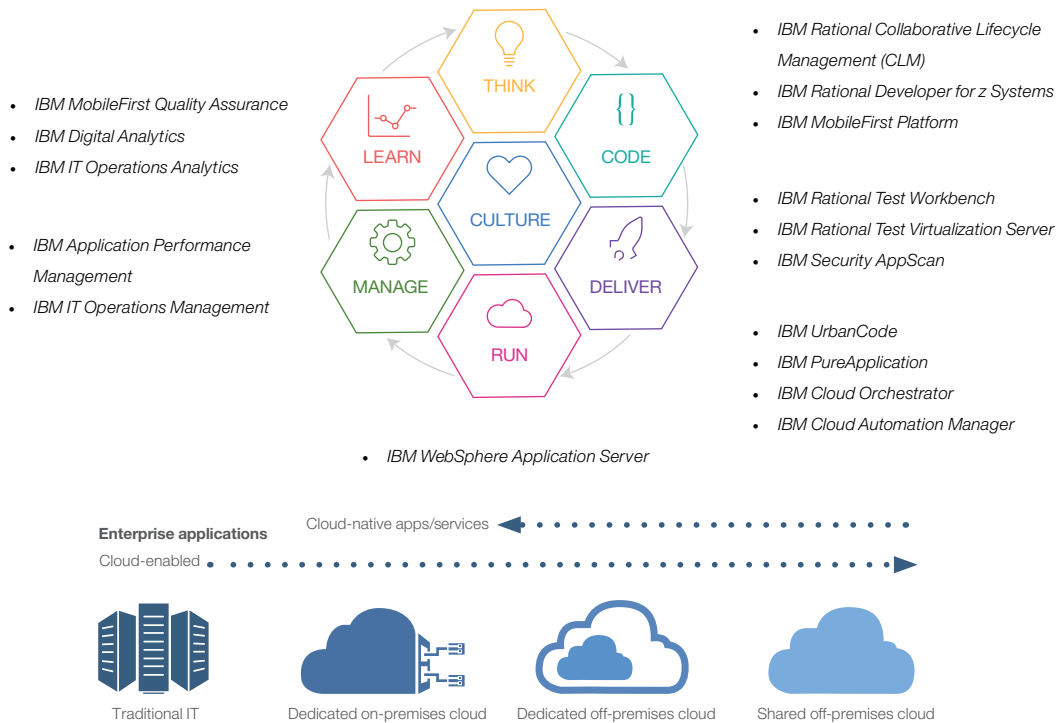


Figure 6. IBM DevOps solution for evolutionary scenario

Blended scenario for multi-speed IT

In many scenarios, cloud-native and cloud-enabled teams need to work together; this introduces “multi-speed IT”. To ensure effective outcomes, an additional layer of management is required to align activities between cloud-enabled teams and

their cloud-native counterparts. These teams may have different cultures, processes, DevOps tooling and may have to deliver at different schedules as required by the business.

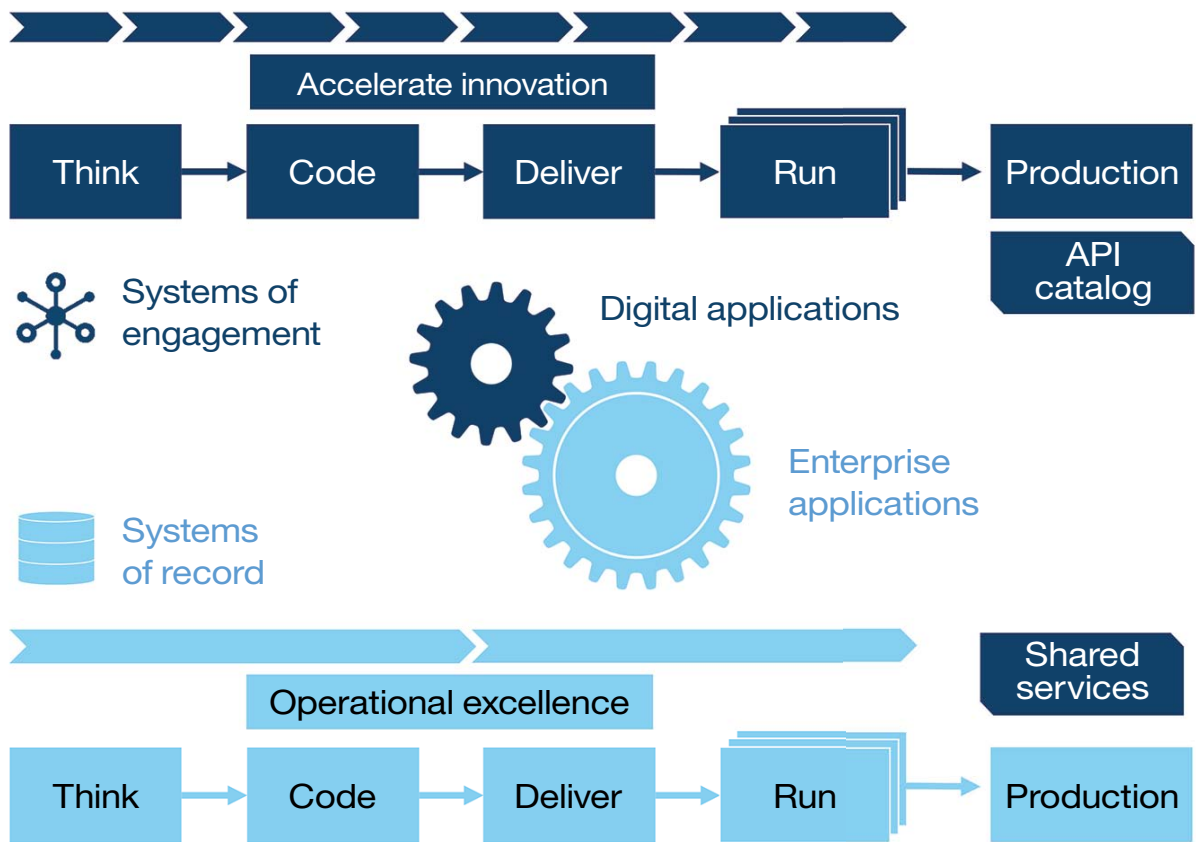


Figure 7. Multi-speed IT with potential needs for a tight collaboration

Based on many client engagements, IBM has observed that potential collaboration problems can be avoided through the use of APIs with clear boundaries. APIs provide a built-in mechanism for cross-group collaboration and alignment, reducing the need for the development processes to be closely aligned. Using APIs in this manner is more common when a cloud-enabled team needs to coordinate a release with a cloud-native team. In such a scenario, the cloud-native team leverages APIs provided by the cloud-enabled team. Unless a change is required to the API, the cloud-native team can then work independently.

For more information on API management see:
ibm.com/software/products/en/api-management-family;

For more information on cloud integration see:
ibm.com/software/products/en/category/cloud-integration

If APIs cannot be used, IBM observes three potential difficulties: lack of alignment between teams, inability to test applications where back-end services are not yet available, and difficulty in coordinating the deployment of the entire system.

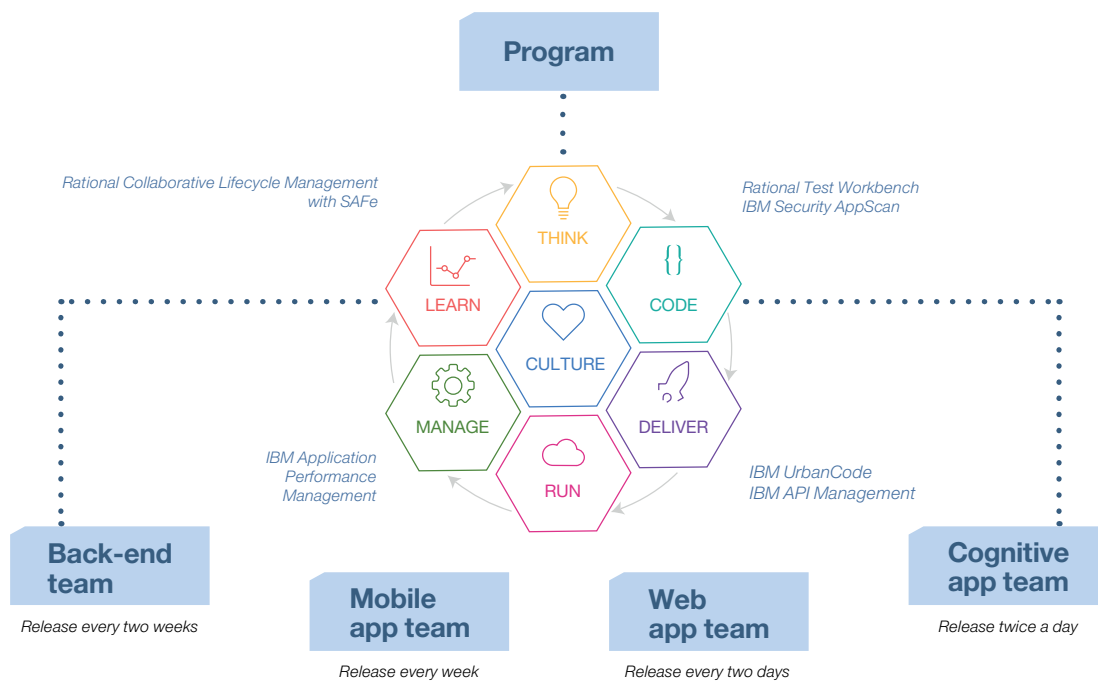


Figure 8: Example of flexible DevOps tool standardization for multi-speed IT. Each team uses the toolchain and processes best-suited to their needs. Standardization at the program level using SAFe ensures collaboration, portfolio management and governance.

IBM recommends the adoption of SAFe (Scaled Agile Framework environment) as a way to coordinate the overall process among teams using different processes and DevOps solutions. SAFe requires teams to synchronize at specific milestones (called Program Increment), and teams can progress on their own cadence in between those milestones. For more information on IBM's support for SAFe, see: <https://bit.ly/ibmsafesupport>.

IBM recommends the use of test virtualization with IBM Rational® Test Virtualization Server, to allow decoupling the environment dependencies for integration testing. This enables teams to quickly test their changes, regardless of the implementation status of the components they depend on, and to mitigate the risks associated with late integration testing. For more information on Rational Test Virtualization Server, see <https://developer.ibm.com/rational/products/testvirtualizationserver>.

IBM recommends the use of application monitoring with IBM Application Performance Management to monitor and manage application performance, availability and security throughout the application lifecycle by measuring the response times of users, application components and specific transactions. For more information on IBM Application Performance Management, see ibm.com/cloud-computing/learn-more/it-service-management/application-performance-management. For proactive operations that empower your team to identify, isolate and resolve problems before they impact your business services, IBM recommends IT Operations Management (ITOA). For more information on ITOA, see ibm.com/cloud-computing/products/hybrid-it-management/it-operations-management.

IBM recommends release management/deployment automation with IBM UrbanCode™ for this multi-speed IT deployment coordination. For more information on UrbanCode, see <https://developer.ibm.com/urbancode/>

Conclusion

In a world driven by emerging cloud, analytics, mobile and social technologies, customers are increasingly seeking a direct, digital relationship with the businesses they choose to transact with. In many industries, born-on-the-web businesses are seizing market share by predicting or even steering customer preference—and delivering exceptional user experience that keeps customers coming back for more. Recognizing that customer preferences change continuously, and that new competitors face minimal barriers to entry, these businesses have designed their application development lifecycles to incorporate frequent customer feedback that drives equally frequent releases and updates.

As they face disruptive competition from increasingly agile market entrants, established enterprises have recognized that traditional approaches to software development and delivery are insufficient. Manual development processes are error-prone and tend to create waste and delayed responses. They also focus too heavily on internal issues, and not enough on customers and their responses to applications. In a competitive landscape where continuous innovation and exceptional user experience are key to winning and retaining customers, enterprises must embrace new approaches that put the customer front and center.

To enable faster and better response to changing customer needs, enterprises are increasing their release frequency for applications. This in turn demands tighter alignment and collaboration than traditionally seen between line of business, development, and IT operations, driving the need for enhanced

collaboration, automation and information transparency among these groups. To achieve seamless internal collaboration, IBM recommends the adoption of a DevOps approach. Naturally, enterprises want to cut costs while improving delivery speed, but they must also manage risk and compliance. The IBM DevOps approach offers a powerful solution to these challenges. DevOps for hybrid cloud can reduce time to customer feedback, increase quality, reduce risk and cost, and unify processes, cultures and tools across the end-to-end lifecycle.

IBM has broad and deep experience helping enterprise organizations transform application delivery, and provides a broad set of offerings in software and services covering the end-to-end lifecycle. IBM solutions support complex enterprise environments (for web, mobile, distributed, mainframe, hybrid cloud), and support multi-speed IT to align systems of engagement and systems of record to teams, skills and requirements. IBM has proven and rapid business outcomes – with best-in-class offerings. For example, IBM UrbanCode is essential to DevOps for hybrid cloud and has demonstrated 482 percent ROI, 97 percent reduction in cost per application release, and 75 percent faster app deployment times (The Total Economic Impact of IBM UrbanCode, Forrester Consulting, August 2015).

The IBM DevOps approach helps organizations incrementally adopt DevOps practices, enabling them to accelerate innovation without tradeoffs in terms of cost, quality or risk. Based on open standards, the IBM solution enables enterprises to leverage existing investments and build an environment in which open source and proprietary lifecycle tools coexist and interoperate. The IBM DevOps solution is also based on best practices designed to help enterprises as they build new systems of engagement that are seamlessly integrated with tried-and-trusted systems of record. The solution provides a platform for integrating an organization's value chain and extended lifecycle across a broad group of stakeholders that includes not only development, but also line of business, IT operations teams and customers.

Whether an organization includes cloud-native teams, cloud-enabled teams or a combination, IBM can provide a DevOps solution from off-the-shelf components to meet the precise requirements. What's more, the IBM approach to DevOps enables organizations to reduce time to obtain and continuously integrate customer feedback. This enables organizations to outperform their peers in a world where managing the customer experience is a key competitive differentiator.

For more information

For more information on how an IBM DevOps solution—on-premises, in the cloud, or delivered as a fully managed service—can help you achieve fast development cycles, lower costs, improved quality and exceptional customer satisfaction, please contact your IBM representative or visit: ibm.com/devops

Additionally, IBM Global Financing provides numerous payment options to help you acquire the technology you need to grow your business. We provide full lifecycle management of IT products and services, from acquisition to disposition.

For more information, visit: ibm.com/financing



© Copyright IBM Corporation 2017

IBM Systems
Route 100
Somers, NY 10589

Produced in the United States of America
July 2017

IBM, the IBM logo, ibm.com, Cloud Platform, UrbanCode, Rational, and Rational Team Concert 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

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

This document is current as of the initial date of publication and may be changed by IBM at any time. IBM Business Partners set their own prices, which may vary. Not all offerings are available in every country in which IBM operates.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary. It is the user's responsibility to evaluate and verify the operation of any other products or programs with IBM products and programs. 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.

Statement of Good Security Practices: IT system security involves protecting systems and information through prevention, detection and response to improper access from within and outside your enterprise. Improper access can result in information being altered, destroyed, misappropriated or misused or can result in damage to or misuse of your systems, including for use in attacks on others. No IT system or product should be considered completely secure and no single product, service or security measure can be completely effective in preventing improper use or access. IBM systems, products and services are designed to be part of a lawful, comprehensive security approach, which will necessarily involve additional operational procedures, and may require other systems, products or services to be most effective. IBM DOES NOT WARRANT THAT ANY SYSTEMS, PRODUCTS OR SERVICES ARE IMMUNE FROM, OR WILL MAKE YOUR ENTERPRISE IMMUNE FROM, THE MALICIOUS OR ILLEGAL CONDUCT OF ANY PARTY.



Please Recycle