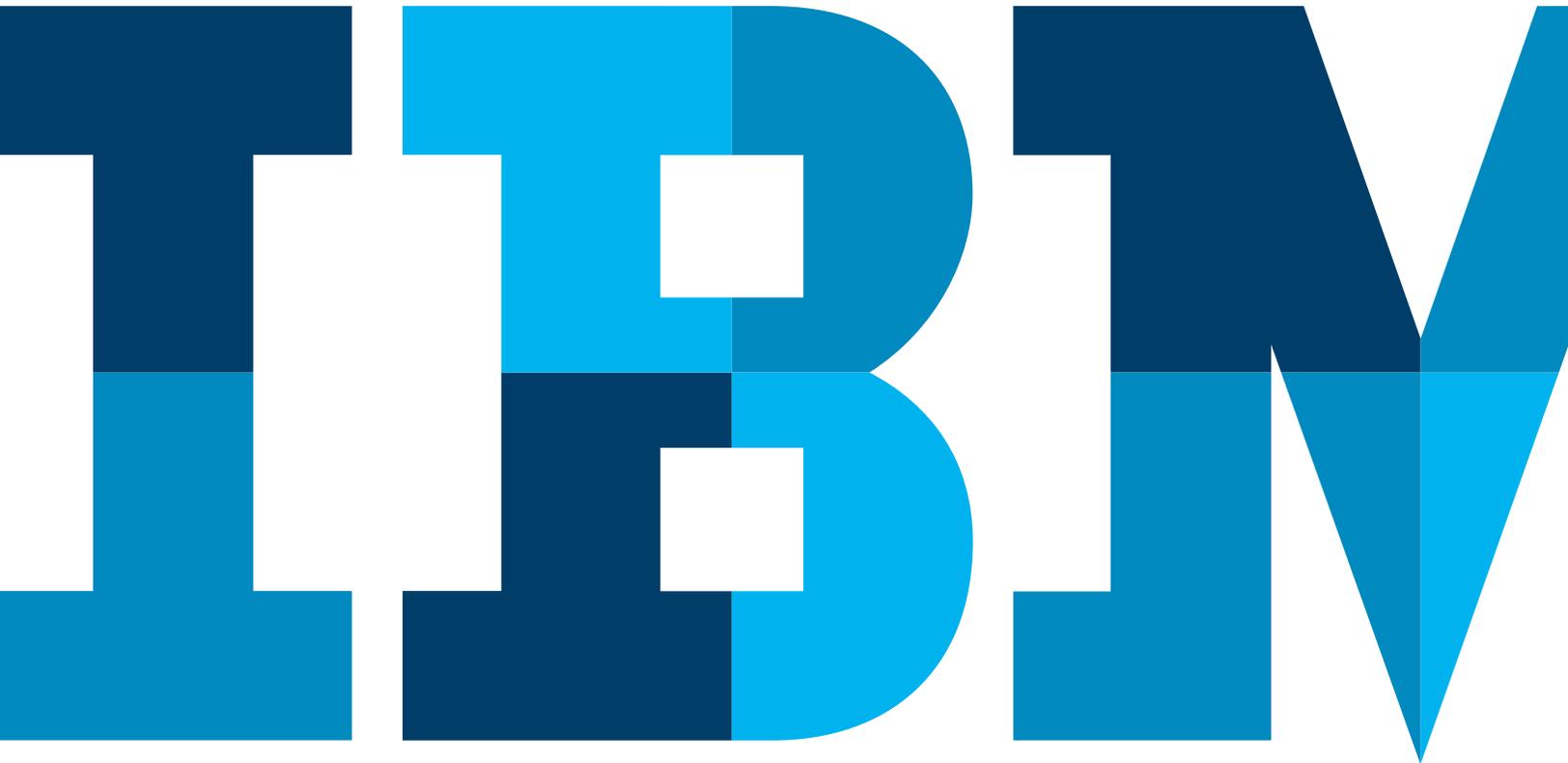


IBM Services Building DevOps on IBM Cloud Private (ICP)



Abstract

At its core, DevOps makes delivery of applications more efficient. IBM Cloud Private has the platform and services to recognize a code change and automate delivery of that change from development, through the support environments, to production. However, delivery of code is just one aspect of DevOps.

IBM extends the DevOps definition, making it an enterprise capability that enables organizations to seize market opportunities and reduce time to customer feedback. IBM's main objectives are speeding continuous innovation of ideas, enabling continuous delivery of those innovations, and providing meaningful feedback for continuous learning, thereby putting all the emphasis on deciding what code to change.

IBM extends DevOps to include all stakeholders in an organization who develop, operate or benefit from businesses systems. DevOps enables design thinking, which focus on user outcomes, restless reinvention, and empowering teams to act. Also, DevOps enables lean and agile methodologies, which guide teams to deliver in smaller increments and get early feedback. These approaches improve the content and quality of the changes in the application delivery lifecycle.

IBM provides an engineering approach to implementing DevOps on IBM Cloud Private on an existing portfolio of applications. Through a discovery workshop, we'll analyze your application delivery lifecycle, identify areas for improvement and then execute proof points on preselected applications. Based on those proof points, we'll help you learn and move forward to onboard new applications, while also monitoring, metering and measuring impact.

The IBM point of view

This paper will provide IBM's point of view on DevOps and how deploying on cloud can help you make the most of it. It will discuss practical approaches while focusing on IBM Cloud Private offerings. It is the latest in our series of papers to help our customers achieve cloud success.

IBM defines DevOps as an enterprise capability that enables organizations to seize market opportunities and reduce time to customer feedback, and has three main business objectives:

- 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 also helping to meet regulatory concerns.
- Providing a feedback loop for **continuous learning** from customers by monitoring and optimizing software-driven innovation.

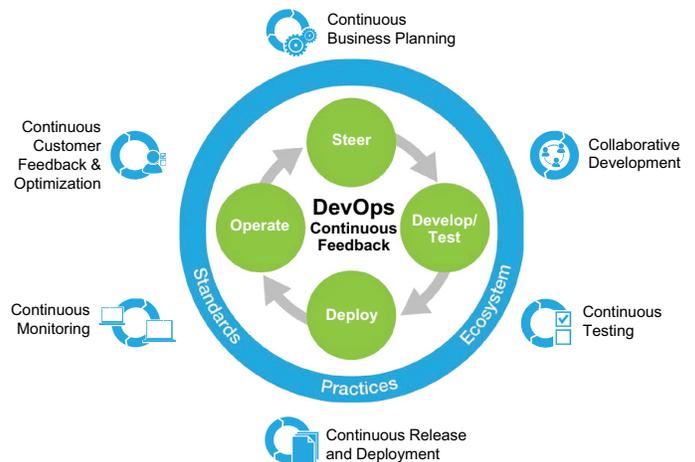


Figure 1: The DevOps continuous feedback loop

DevOps enables process and technology

Process

DevOps works with agile, lean, and design thinking to drive the loop of continuous delivery, feedback and innovation. As we'll see when discussing technology, IBM Cloud's rapid deployment and data collection feed and improve this cycle.

	Speeding continuous innovation of ideas	Enabling continuous delivery of these innovations	Providing a feedback loop for continuous learning
Agile • Fast feedback cycles through early customer involvement	✓	✓	
Lean • Value stream mapping • Eliminate waste	✓		✓
Design thinking • Focus on delivering a delightful user experience	✓		✓
Cloud operation • Quick and flexible management of development, test and production environments • Resilient and scalable		✓	
Automation • Removing the silos between development and IT operations: "treat infrastructure as code" • Infrastructure as Code • Continuous delivery of changes		✓	
Application Analytics • Real time insight on problems in production • Insight on application usage			✓

“Lean and agile thinking guides teams to deliver in smaller increments and get early feedback. As a result, teams reduce cycle time by focusing only on those activities that maximize value based on feedback. Wasted effort is identified and eliminated, enabling teams to spend time on value add activities, such as innovation and quality improvements.”

—Agile for Dummies, 3rd IBM Limited Edition

IBM's DevOps approach is based on the [IBM Cloud Garage Method](#), which applies these thinking principles to all stakeholders in an organization who develop, operate or benefit from the business' software systems, including customers, suppliers and partners. By extending lean principles across the entire software supply chain, DevOps capabilities can improve productivity through accelerated customer feedback cycles, unified measurements and collaboration across an enterprise, and reduced overhead, duplication and rework.

Design Thinking principles include:

- Focus on user outcomes, and drive business by helping customers achieve their goals.
- Restless reinvention: stay essential by treating everything as a prototype.
- Move faster by empowering diverse teams to be proactive.

Table 1: The role of process and technology in DevOps

Design Thinking provides a complementary set of principles and practices that fit very well with a DevOps approach. In the traditional model, developers are often the furthest removed from the customers. Design Thinking reverses this by allowing developers to respond directly to customer feedback. A DevOps team that applies Design Thinking will focus on achieving their customers' goals, delivering a quickly expanding minimum viable product based on customer feedback, and empowering team members to fail until they succeed.

Automation

Continuous integration is a DevOps practice where developers continuously commit their code changes into a source repository. Then, at regular intervals, the system will conduct an automated build, deploy, and test. **Continuous delivery** expands on continuous integration by automatically deploying code changes through the support environments, pausing for approval before going to production. **Continuous deployment** does not wait for approval; rather, it goes to production automatically after successfully passing through automated testing in the support environments.

You can use automation technology to build a target support environment that includes application, database and test servers, install and configure middleware and applications, and then execute automated testing. When the testing is completed, the environment can be released. IBM DevOps-as-a-Service enables this level of automation through its toolsets and pay-as-you-go pricing model.

DevOps-as-a-Service can either be consumed through a shared services model, or you can opt to have your own dedicated instance of the toolchain. The shared service platform is setup on IBM Cloud, with all the security standards and checks. If there is a need to customize the toolchain to meet the client demands, this platform can be used as an Open Toolchain Platform that allows plug-and-play. Some of the highlights of this shared service model are:

- Supports Hybrid solution scenarios across custom / analytics components & platforms
- Shared pool of cross skilled DevOps SMEs to support various projects / scenarios
- Re-usable DevOps patterns for key offerings
- Platform supports plug-and-play of client specific and/or other open source and 3rd party tools
- Low cost of ownership and very low CapEx and OpEx

Below is the diagram with the standard toolchain that is used in IBM DevOps. One thing to notice is the blend of Open Source tools, Industry Standard & IBM tools. This blend helps us implement an end-to-end DevOps solution for most of the technologies for Hybrid and Multi Cloud deployments.

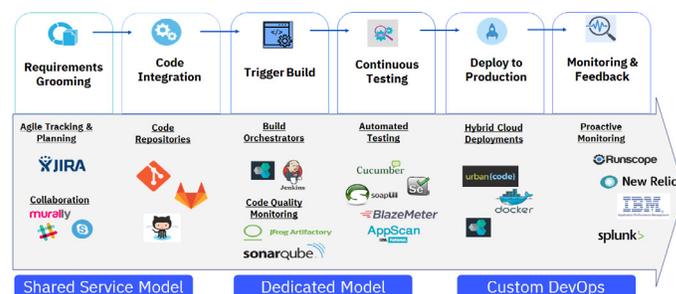


Figure 2: The DevOps day in the life

Application analytics

Understanding how an application is being used is valuable to both the business and technology teams. Telemetry services of DevOps Continuous Delivery Pipeline is measured by a set of Industry standard monitoring tools. The data from the monitoring tools are fed into Cognitive tools, with today's analytics capabilities one can derive insights in various facets of the delivery life cycle. The service can also be used to gain system-wide visibility into resource utilization, application performance and operational health.

For instance, data can be captured about how customers interact with applications. The section on continuous learning later in the paper will touch on some of the tools available to help you gain insight into how your customers are interacting with your applications.

Continuous Innovation

Businesses are under tremendous pressure to create new value for their customers through innovation. However, they are finding that traditional approaches to software development and delivery are not sufficient to deliver the business innovation their customers expect. Manual development processes are error prone, wasteful, and known to cause significant delays. Through proper application of new technology and the principles of continuous innovation, businesses can eliminate these manual tasks, and start delivering value like never before.

Continuous innovation means continuously developing new ideas into innovative software, which in turn can continuously improve the value delivered to customers. IBM believes that DevOps is one of the primary means for achieving this sustained innovation.

In its conventional sense, DevOps refers to a closer collaboration between development and operations teams, and the integration of associated processes and tooling. In IBM's point of view, DevOps is much more than that. We believe that DevOps should encompass collaboration among all stakeholders—not just between development and operations, but also among lines of business, suppliers involved in software delivery and customers themselves. In this expanded definition, DevOps includes business governance practices around security and compliance and all aspects of the delivery process, such as multi-sourcing.

Continuous Delivery

The main goal of DevOps is to make delivery more efficient. Support environment availability and configuration is a roadblock that often interferes with achieving this goal. It's important to ensure that the support environment matches the production environment, as a mismatch can introduce significant quality issues. Also, changes to complex systems—even when componentized—can have unexpected results. Requirements, written or verbal, can be misinterpreted. Automating functional and non-functional testing, along with early feedback by stakeholders, is critical to maintaining quality. Deploying DevOps-as-a-Service can help address these problems. The operational process of DevOps-as-a-Service is as follows:

A typical “day in the life” of the development team starts with Product Backlog (Requirements) grooming, and all the Product Backlog Items are captured in JIRA tool. The team also uses these tools for all their planning and tracking purposes. All the code and other development artifacts are stored in GitLab. All the commits to the repository is linked to the Work Item in the planning tool to create a traceability view.

When a developer checks-in his/her code, a build is triggered. This build includes code compilation, code quality check, static code analysis and sanity tests. This is all done as part of the Continuous Integration (CI) pipeline. Jenkins is used as the orchestrator that is plugged in to all the tools in the toolchain. SonarQube is used for Static Code Analysis, with AppScan for Security checks. All the Unit/Integration Tests are also run as a part of the CI process. Once the code has been built successfully, the code will be committed to the repository, and linked to the Work Item (Story, Task, Defect, etc.) that triggered the build. The built artifacts are then stored and version controlled in JFrog Artifactory. Artifactory is also used to store the Container images.

The code then gets promoted to the QA environment. In the QA environment, all the end-to-end tests are run before deploying the application to the production environment. Selenium is used for Web based functional testing. For all Mobile based functional testing we use Appium. The other automated test tools include SoapUI for API and end point testing, Blazemeter for Performance and Cucumber for Behavior Driven Development.

Application deployment is done using UrbanCode Deploy (UCD). Docker Containers are used for Containerization, and the toolchain is also capable of automatically deploying containers using Chef and UrbanCode Deploy. The deployed application is then Continuously Monitored for any anomalies. The deployed application code can be traced back to its work item as this process has created an end-to-end traceability

IBM shared service DevOps platform supports multi-cloud solutions, enabling deployments across multiple environments. Any other tool/s that is not a part of the standard IBM shared service DevOps platform toolchain can be plugged into the platform and the toolchain can be created.

An accelerator for automated creation of Continuous Delivery Pipeline in any cloud provider like IBM private cloud or AWS can be used. The DevOps toolchain creation has been automated so the setup time required is considerably reduced. The DevOps toolchain creation has quite a few options, like you can select the required DevOps tools, select the particular type of cloud provider and create the Continuous Delivery Pipeline in the selected cloud provider. Once the selection of tools are done the DevOps toolchain creation is just a two-step process.

See the glossary at the end of this paper for a more detailed description of these services.

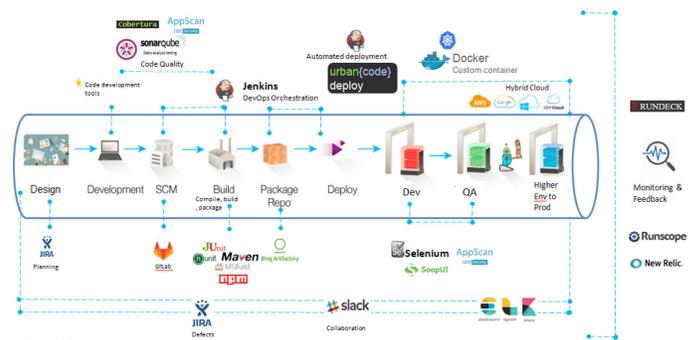


Figure 3: Deploying a code change in Shared Service DevOps Model

Continuous Learning

As mentioned previously, understanding how an application is being used is valuable to both business and technology teams. In addition to Industry standard monitoring tools, there are a variety of analytics solutions available to help you better understand how customers use your applications.

IBM Digital Analytics, formerly Core metrics

Web Analytics, is a platform for near real-time digital analytics, data monitoring and comparative benchmarking. The solution allows you to track and analyze visitor behavior over time,

across multiple touchpoints and channels, and deliver more personalized, relevant and effective information. It also allows you to optimize your web, mobile and social channels by monitoring critical data and key performance indicators in near real-time. With IBM Digital Analytics, you can uncover growth opportunities and areas for improvement.

IBM Tealeaf is a family of products to improve visitor interactions with customer experience management solutions. For example, IBM Tealeaf CX provides visibility into web and mobile browsers, capturing data down to the individual session level, and analyzes the data to uncover trends and valuable insights. The solution can help you discover unexpected customer pathways through your applications, and areas where customers tend to struggle the most.

IBM's approach to DevOps on IBM Cloud Private

IBM adapted its methodology to provide an engineered approach to implementing DevOps on IBM Cloud Private.

Discovery workshop

During a discovery workshop, IBM will help your team perform an assessment of your application delivery lifecycle management (ADLM) through value stream mapping. This is a lean strategy for analyzing the current “as is” state and designing the future “to be” state of ADLM.

The mapping looks at the full lifecycle documenting each step, milestone, and gate. Data about the effort in man hours and duration, as well as value is considered. In the end, the discovery workshop looks to address bottlenecks in your pipeline. Some common examples include:

- Replacing ticket-based environment provisioning with non-cloud or cloud-hosted self-service
- Replacing weekly hands-on deployments to an integrated environment with more automated daily or on need basis deployments
- “Shift left” integration testing to match more frequent deployments, while focusing on increasing automation testing

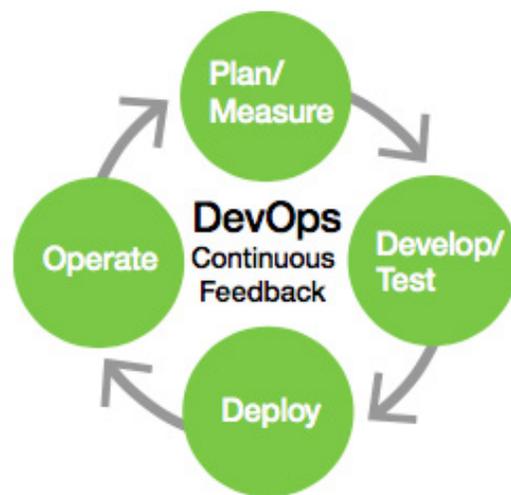


Figure 4: DevOps continuous feedback loop

In the discovery workshop, you will review your existing application portfolio for automation readiness, and then select applications that can provide quick wins.

Proof point

Following the discovery workshop, IBM will help you gather baseline as-is metrics for environment management and application code delivery. Then, we’ll execute proof points on select applications by testing proposed remediation of bottlenecks such as automation of environment creation and code delivery. Finally, we’ll help you capture DevOps-enabled metrics, and then analyze.

Optimize and expand

Just as we apply DevOps principles to application development, we'll be applying them here as well. Our goal at this stage is to help you learn from the proof points, onboard the next set of applications, and continue to monitor and measure impact.

For more information

To learn more about IBM Cloud Migration Services, visit us at ibm.com/services/cloud/migration, or contact your IBM representative.

Glossary**Jira**

Jira is a leading Agile project management and Issue tracking tool developed by Atlassian.

Jira forms an important tool of DevOps to olchain as a part of Continuous Business Planning (CBP).

Jira comes with various agile reporting functionalities. Software's capabilities can be greatly extended with the help of over 1000 Plugins available in the Atlassian marketplace. Some of the important plugins include Zephyr Test management plugin, Portfolio Management, Dataplane reporting etc. All the activities are assigned to developers and tracked to closure in Jira. All the team members have access to their respective project area to update their activities. Jira is configured to give real time status of the activities via advanced Scrum or Kanban Boards. Agile teams can stay focused on delivering iterative and incremental value, as fast as possible. The project status can be measured via metrics like velocity, burndown charts and others as reports.
www.atlassian.com

GitLab

Gitlab is an Open Source tool or repository management tool from Git. GitLab forms an important ingredient in the DevOps toolchain as a part of Continuous Integration (CI) process. GitLab has extensive user management capabilities and supports multiple project repositories isolated from each other with extensive scalable features. Gitlab will be integrated with tools like developer IDE, Jira, Jenkins and others as part of the Continuous Integration platform. Branching strategy aligned to Agile way of working is well managed in GitLab. The traditional merging and reviews is governed by the branching strategy and process defined in GitLab. The webhook between the tools- GitLab, Developer IDE and Jira gives activity status and traceability.
about.gitlab.com

UrbanCode Deploy

IBM UrbanCode Deploy is an application release automation solution that combines robust visibility, traceability and auditing capabilities into one optimized package. UrbanCode Deploy orchestrates and automates the deployment of applications, middleware configurations, server virtualization and database changes into development, test and production environments. UrbanCode Deploy is designed to facilitate rapid feedback and continuous delivery in agile development while providing audit trails, versioning and approvals needed in production. Continuous delivery breakers and Rollback scenarios in the DevOps pipeline can be created based on UrbanCode Deploy workflow results. In UrbanCode Deploy patterns can be created for a technology area which can be reused thereby reducing significant rework.

The reports from UrbanCode Deploy are analyzed for deployments, number of virtualized VMs and deviations from release cycles.

developer.ibm.com

Jenkins

Jenkins is a self-contained, open source automation server which can be used to orchestrate tasks such as build management, package management, testing, deploying, server virtualization and monitoring. Jenkins has a vast set of plugins and integrates with most of the industry standard DevOps tools. Jenkins can be installed through native system packages, Docker, or even run standalone by any machine with the Java Runtime Environment. Jenkins is used to build Continuous Delivery pipeline which is unique for a particular technology and framework. Once the Continuous Delivery pipelines are done, the team gets a ready to use automated build and deployment process. The development teams can use them for their daily consumption.

jenkins.io

Artifactory

Artifactory is a Universal artifact repository management tool. Artifactory forms an important ingredient in the DevOps toolchain as a part of Continuous Integration (CI) process. Artifactory supports software packages created by any language or technology. Integrating with all major CI/CD and DevOps tools, Artifactory provides an end-to-end, automated solution for tracking artifacts from development to production. Artifactory is used to store all the binaries of the pipeline runs. Once the pipelines are done, Artifactory is configured to store all the binaries. The development team might need to check if the binaries are stored in the designated folders.

jfrog.com

SonarQube

SonarQube software (previously called Sonar) is an open source quality management platform, dedicated to continuously analyze and measure code quality, from project portfolio to

methods. It is a static code analyzer. SonarQube forms an important ingredient in the DevOps tool chain in the Continuous Integration (CI) process. The reports from SonarQube are analyzed for code quality which gives quality indications to the development team. Build breakers in the Continuous Delivery pipeline can be created based on SonarQube results. The threshold limits of the SonarQube results can be set, deviations of the threshold limits will lead build breaker and notifications. Threshold limits can be configured according to the needs of the project operations.

www.sonarqube.org

Cucumber

Cucumber is a testing framework which supports Behavior Driven Development (BDD). Cucumber is an open-source for executable specifications. Cucumber defines application behavior in plain meaningful English text using a simple grammar defined by a language called Gherkin. Cucumber merges specification and test documentation into one cohesive whole. Cucumber-BDD is at the heart of development practices, in a broad way. Not only developers are involved but also testers, business analysts, etc. The test specification can be written much earlier than the actual development starts. Cucumber is also a key element in the agile transformation process and is what made it possible for us to truly do continuous delivery.

cucumber.io

Selenium

Selenium is an open source automation testing tool for Web applications. Selenium has a client-server architecture. Selenium supports multiple programming languages like HTML, Java, PHP, Perl, Python, Ruby and C#. Automated test scripts can run on multiple browsers like IE, Mozilla Firefox,

Safari, Google Chrome and Opera. Selenium supports multiple operating systems like Windows, Linux and Mac. Automated scripts written in Selenium can be version controlled for reuse. The testing scenarios can be tailored according to the need of the domain or application. The automated test scripts can be provisioned to execute a particular test suite on need basis in the Continuous Delivery Pipeline. This helps in creating quality product and fail fast fail early -Shift Left a reality.

www.seleniumhq.org

AppScan

AppScan enhance web and mobile application security, improve application security program management and strengthen regulatory compliance. Testing web and mobile applications prior to deployment can help you identify security risks, generate reports and fix recommendations.

Reduce risk exposure by identifying vulnerabilities early in the software development lifecycle.

Test applications prior to deployment and for ongoing risk assessment in production environments.

Classify and prioritize application assets based on business impact and identify high-risk areas. AppScan can be instantiated on need basis during the process of Continuous Delivery Pipeline.

ibm.com/security/application-security/appscan

Slack

Slack is used as a collaboration tool. Real time collaboration using multiple channels (web, mobile, etc.) enabling all interested stakeholders to tune in for real time status on high severity issues being worked via Slack. This enables a faster resolution of notifications.

slack.com

Rundeck

Rundeck is cross-platform open source platform used as runbook automation enabler. In Runbook one can create and execute runbooks, event-triggered guidance and automated tasks. One can also enable runbook execution for manual runbooks, semi-automated runbooks and fully automated runbooks. Runbook are set of events which can trigger multiple set of activities based on triggers. The triggers can vary depending on the context of the continuous delivery pipeline process.

www.rundeck.com

Runscope

Runscope is used to monitor and test APIs, Global API Uptime Monitoring. Third party SaaS product which execute synthetic transactions to simulate user behavior by invoking APIs/URLs. Runscope integrates with a wide variety of popular notification and messaging apps, including Slack, PagerDuty, email and Command center ops. Runscope can be triggered by Continuous Delivery pipeline on need basis.

www.runscope.com

ELK

The ELK Stack is an open source log management platform. ELK persists the logs in a centralized fashion, the logs can be from application, process and infrastructure. The logs from ELK can be viewed and monitored from a single location. The downstream from ELK can fed to a cognitive engine to predict the failure of systems involved in Continuous Delivery.

www.elastic.co

Resources

IBM Cloud Garage method

ibm.com/cloud/garage

IBM Cloud products and services

ibm.com/cloud

IBM Cloud Innovate

ibm.com/services/campaigns/cloud-innovate-infographic/

IBM Cloud DevOps

ibm.com/cloud/devops

Authors

Raghava Venkat

Venkat is Offering Leader for DevOps and IBM Cloud Private at IBM Services. He is focussed on Cloud Advisory and Consulting Services, with expertise in DevOps and Agile Transformations across IBM's Enterprise clients.

Mahesh Chandranna

Mahesh is the Global Lead DevOps Architect at IBM Global Business Services. He is focused on architecture and delivering DevOps transformations in Hybrid Cloud areas for some of IBM's largest enterprise customers, where he brings extensive experience and expertise in DevOps and Cloud.



© Copyright IBM Corporation 2018

IBM Corporation
New Orchard Road
Armonk, NY 10504

Produced in the United States of America
July 2018

IBM, the IBM logo and ibm.com 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 www.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 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.



Please Recycle