Continuous Integration using DevOps
The future of Information Technology

Akshata S Mahashabde
PG Student (Advanded Information Technology)
REVA UNIVERSITY, Bangalore, India.
Email: smakshata@gmail.com

Dr. Kiran Kumari Patil
Director,
REVA UNIVERSITY, Bangalore, India.
Email: Kiran.kumari@reva.edu.in

Abstract—**This paper describes the implementation of DevOps technology on the traditional IT system. The software industry talks about CD - Continuous Deployment, CI – Continuous Integration and CT – Continuous Testing, these jargons are being used worldwide. All the above mentioned jargons describe the end-to-end process of the software development life cycle. Continuous delivery helps in deploying the changes made in the code immediately into production at a fast pace possible. To have the continuous delivery successful, all other areas of continuous integration and continuous testing should be well defined and should be in place. All these capabilities constitute the goal of DevOps. Hence, I would choose to work on the vast coverage of continuous integration which forms the major part of the entire DevOps cycle for any software development.**

DevOps is an approach follows the lean principles and also the agile principles in which business owners and the development teams, operation teams, and quality assurance teams collaborate to deliver software in a continuous manner that enables the business to more quickly seize market opportunities and reduce the time to include customer feedback. Indeed, enterprise applications are so diverse and composed of multiple technologies, databases, end-user devices, and so on, that only a DevOps approach will be successful when dealing with these complexities. DevOps aims to produce the software rapidly to reach the competitive deliverable to the consumers.

The software industry takes a broad overview and recognizes DevOps as a strongly business-driven software delivery approach that takes a new or enhanced business capability from an idea all the way to production, providing business value to customers in an efficient manner and capturing feedback as customers engage with the capability. To perform this, you require participation from the customers or the stakeholders beyond just development teams and operations teams. An effective DevOps approach includes different lines of business. There could be practitioners, business executives, business and technology partners, suppliers, and so on.

DevOps is a business capability in terms of capacity, material and expertise that spans all stakeholders in an organization, including business owners, architecture, design, development, quality assurance, operations, security, partners and suppliers. To get started with DevOps it includes creation of the right culture. The onus of identifying the business challenges and also the bottlenecks that needs to be eliminated in the delivery pipeline.

I. INTRODUCTION

DevOps started as a culture and set of practices and a cultural change that utilize the automation tools and processes 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

The customers want to increase the release cycles of the applications to achieve the best possible customer experience. DevOps helps create an automated development, automated testing and automated delivery. The industry 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 (a.k.a. two-speed IT, or bimodal IT) in combination with methods such as Scaled Agile Framework environment (SAFe) to facilitate collaboration.

●● Continuous delivery is the desired future-state achieved with the application of a DevOps approach.

The goal of a software company is to deliver the software in demand from the customers to be reached at a faster pace. The software should reach the customer on an iterative basis with a dynamic feedback for improvements. There are lots of works and write ups on DevOps. Here is a futuristic understanding of DevOps as a culture, a process and as an architecture.
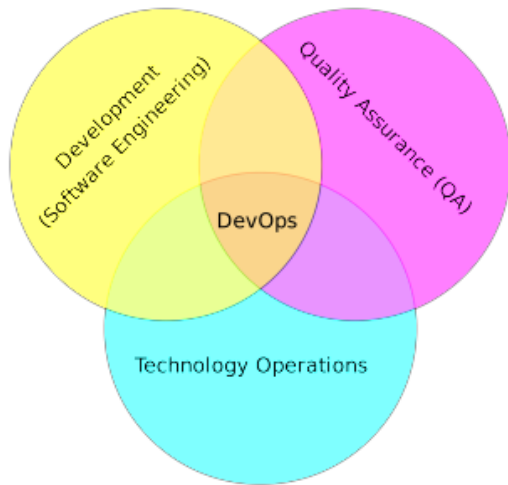
Featured DevOps Solutions

DevOps solutions make a marked change in software delivery processes using the below practices:

Application Lifecycle Management: Collaborative development for agile delivery of high-quality solutions.

Software Project Management: Continuous business planning to deliver optimal customer value with speed.

Application Monitoring and Feedback: Continuous software application monitoring and feedback to optimize user experiences

Test Automation: Continuous testing for agile delivery teams.
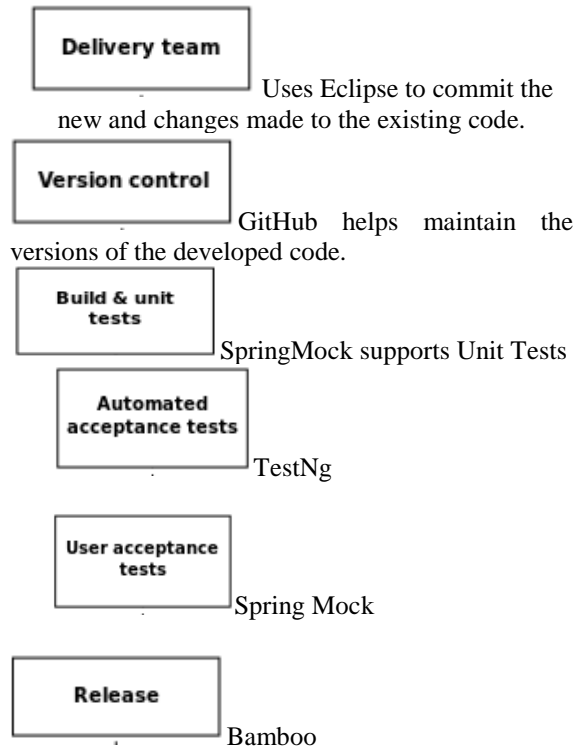
Adoption of DevOps

DevOps is adopted by all major organizations like Amazon, Flipkart, Snapdeal and many more which rank in the e-commerce business. The motto behind adoption of DevOps by all these organizations is to achieve higher quality, quicker delivery and speed up to the current digital speed. Modern service delivery concentrates and experiments on the different applications of DevOps to define an ideal benchmark. Competency models are built to compare with the traditional system and the newly evolving DevOps to conclude the best methodology that would help stand high in the industry.

The transition from the traditional waterfall model to DevOps is a very long journey and does not happen overnight. The efforts behind implementing DevOps culture is a challenge as every team in the organization should be convinced on the journey. The major change an organization would see is a fully automated process. Manual effort would reduce and hence the learning curve in the organization is the key challenge.
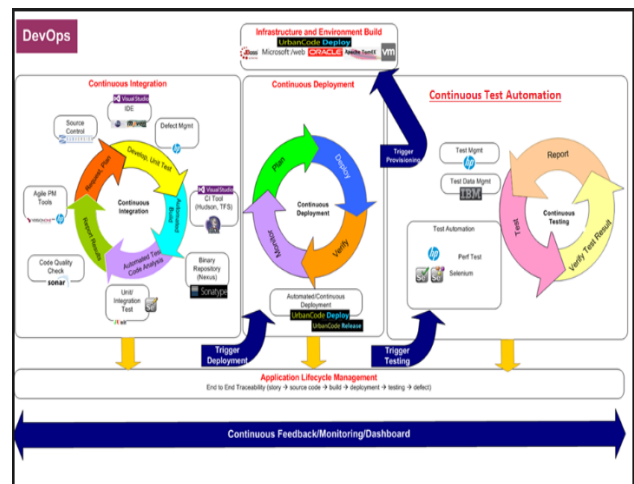
The above image demonstrates the DevOps architecture which is a combination of Continuous Integration, Continuous Deployment and Continuous Test Automation. The Infrastructure stands above all the three where the actual deployment happens. The deployment could happen virtually on the VMs or it could be on cloud or it could be a bulk deployment on hundreds and thousands of servers. These servers are further accessed by the customers. Every customer can view the required customization of the architecture relevant to their business. The Continuous feedback and monitoring happens in the backend by the operations team to help understand the views of the customers. This monitoring is dynamic and happens every time the new deployment happens. The internal feedback monitoring happens every time a new line of code is committed onto the development branch.

Architecture of continuous delivery with the below representation with some open source tools aligned.

Delivery team — Uses Eclipse to commit the new and changes made to the existing code.

Version control — GitHub helps maintain the versions of the developed code.

Build & unit tests — SpringMock supports Unit Tests

Automated acceptance tests — TestNg

User acceptance tests — Spring Mock

Release — Bamboo

Literature Survey:

DevOps can be applied to many industries that include Manufacturing, Automotive, Healthcare, Medical, Finance, Mobile and Networking industries. Rackspace 2014, gives an evidence that by end of 2015 about 94% of the industries would adopt DevOps. Hence, being in the year 2017 we can assume the adoption percentage is reached and would have in-fact crossed.

I have referenced about 10 papers on DevOps implementation and have come with a gist on what DevOps is and how it can be implemented on any software.

Any software application shall start with a process of requirement gathering and prioritizing them to identify the requirement that has to be implemented first and the ones that can be ignored for the moment. The manual effort involved in prioritizing the requirements can be reduced by using certain tools available which help in identifying the high priority requirements based on a particular criteria selected.

The next phase would be the design phase that helps the development and the test team to understand how the software would look like and what would be the behavior after it is developed. There are many tools in the market that help in the design phase which auto-generate the designs in the form of flowcharts and UML diagrams.

Eclipse is the most common tool used to develop a software and hence, an ideal tool for developers to write code. It helps in auto completion which is one of the best features offered that also reduces the cost and effort. Debugging of the code is easy and helps navigate to the erroneous line easily by giving a highlighted underline. It provides a feature of importing the packages required to support the behavior being developed.

Here, I have used JIRA as the project management tool that helps me in managing the Plans and tracking them through the release. The issues, plan sprints and the tasks can be created and distributed within the team. Prioritization and discussion of the teams work with visibility can be done using JIRA software. JIRA also supports releasing and shipping the build to the intended customers. Reporting is also a major feature provided by Atlassian with JIRA.

Bamboo is used for continuous integration of all the tools, product deployment and delivery. The build generation is triggered automatically after a code commit. This can also be configured that helps auto generate a build ready for testing. No manual intervention is required to generate a build. Whatever the time of commit of code could be, it triggers a build to be generated that also gives insight on the new lines of code added or modified by which team member. Bamboo is quite a famous and promising tool that easily gets integrated with JIRA.

I shall be using Spring Boot which is a java library framework meant to reduce the efforts for writing and helps reduce the time taken. It helps to run, build, deploy and test the unit tests in a single click from the eclipse IDE. Spring Mock similarly is a spring framework that is used to perform the unit tests as part of the framework.

Gradle is another open source tool which I have used for build automation which builds upon the concepts of Maven. It also helps identify which parts of the build tree are latest to ensure any task that has dependency need not be run.

I shall use GitHub to do the version control as it is a web based service and supports ease of integration with JIRA. Distributed version control and source code management is the core feature job of the tool and is an open source.

Sonarqube would be yet another tool integrated in the entire flow that is meant for code coverage. It is an open source platform that helps check the quality of the code and also checks if the code meet the global standards defined.

At the end, I would like to mention that all the tools above are integrated to create an end to end flow that demonstrates DevOps. This helps the IT performers who incorporate it to be more agile with high reliability. Continuous software delivery is the key factor that is emphasized upon that involves lesser complexities and faster resolutions.

Culturally DevOps creates a productive team with high development opportunities. The business benefits align to the technical benefits which include faster delivery of features, stable environments, improved collaboration and communication within teams.

Reference:

[1] DevOps through Continuous Testing: https://www.spirent.com/-/media/White-Papers/Test_Automation/Accelerating_devops_through_continuous_testing_whitepaper.pdf

[2]DevOps literature review: https://www.researchgate.net/profile/Chintan_Amrit/publication/267330992_Report_DevOps_Literature_Review/links/544ba33f0cf2bcc9b1d6bd8a.pdf

[3] Challenges in Continuous Delivery: https://helda.helsinki.fi/handle/10138/169089
Continuous Integration for Cross Platform: http://ebc.ie.nthu.edu.tw/StudentProject/eifinal/2015_eiProject/word/15.pdf

[4] Principles and Practices of Continuous Delivery and Deployment: https://peerj.com/preprints/1889/

[5] Changing scenario of Testing Paradigms using DevOps: http://computerresearch.org/index.php/computer/article/view/1143/1130

[6] Role of Continuous Delivery in IT and Organizational performance: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2681909

[7] Book: Continuous Delivery and DevOps – A Quickstart Guide (Second Edition) , Author – Paul Swartout https://books.google.co.in/books?hl=en&lr=&id=AtMGBgAAQBAJ&oi=fnd&pg=PT7&dq=devops+through+continuous+delivery&ots=CkvZg-uIEZ&sig=oYu-f3vfttm99cIao5z7OfnGtOI#v=onepage&q=devops%20through%20continuous%20delifvery&f=false

[8] Forrester.com: Six Trends that will shape DevOps adoption https://www.suse.com/promo/devops/?campaign_description=FY17%20Nov%20Global%20DM%20DevOps%20on%20DZone&campaign_id=GSDGNdzon23455&utm_campaign=Devops&utm_medium=banner&utm_source=DZone.