Fundamental Challenges in Identifying Tools for DevOps Pipeline

R. Vaasanthi
Research Scholar
SCSVMV University
Kanchipuram

V. Prasanna Kumari, PhD
Professor, IT
Rajalakshmi Engineering
College, Chennai

ABSTRACT

In the world of automation, confronted with a new problem that is selecting the right tool for implementing Continuous Integration and Continuous Deployment. While organizations understand the importance and benefits of DevOps, the richness of the tools and process available to us now have led to an "agony of abundance" when it comes to selecting the tools for building a DevOps tool pipeline. There are enormous amount of tools are present in DevOps space [1] and many challenges associated are with building a DevOps pipeline such as analyzing the existing tools, understanding the developer's pain point and process, existing manual tasks, choosing the right tools and integrating the tools with the existing tools. Although there are quite some DevOps maturity models [2] available in the literature, they are neither specific to large-scale financial organizations nor include the agile aspects within their scope. There must be a clear understanding of what kind of tools can be suggested to the customers in each space. For instance, to suggest a build tool alone in the pipeline they need to compare seven build tools. [3]This works aims to understand the challenges faced before suggesting a tool to the customer.

Keywords

Devops, Tools pipeline, Challenges, How to select the tool?

1. INTRODUCTION

The way organizations deliver software is going through a wave of change as there is more pressure to deliver the products quickly and with quality to sustain their business among their competitors. Enterprises can no longer afford to make their customers keep waiting for months or years for a release. To achieve this it's mandatory for the enterprises to adopt DevOps and Agile. DevOps and Agile play a vital role in implementing Continuous Integration, Continuous Deployment, and Continuous Delivery both in Cloud and On-premises. This paper is organized as follows. Section II overviews DevOps and Agile. Section III details the current real-time scenario and Section IV describes the process DevOps architects have to undergo before forming a DevOps pipeline and Section V provides an example of the current state and target state changes with the conclusion.

2. WHY DEVOPS?

DevOps, the combination of Development and Operation is a set of practices that are trying to bridge the developer-operations gap in the software development Lifecycle. The main focus of the DevOps is Continuous Integration, Continuous Testing in a production-like environment, implementing security and customizing the application to be in a shippable state, Continuous feedback, Continuously Monitoring the software and Notification of the application/server status to the stakeholders. Due to these enormous advantages of DevOps, enterprises are forced to adopt DevOps to sustain in the market.

3. WHY AGILE?

Software development is plan-driven which is always associated with changing requirements, change in a tool, late delivery, minimal or no coordination, compromised in quality. Agile Methodology provides flexibility to handle changing requirements, improved communication, and coordination between developers, testers and business analysts, improved quality and time to market. The Agile principles also provide Enterprises with the flexibility to adopt a selected subset of principles and practices based on their needs. Thus, Agile plays an important role in Enterprises dynamic software development.

4. REAL TIME SCENARIO

ABC is a multinational telecommunications company that believes the proliferation of networks allows people across the globe to enrich their lives through easy interaction and has revenue of more than 10$ million in 2019 is headquartered in America. The company processing 5 million transactions every day decides to standardize its software application tools and streamline the process gaps due to manual intervention, process gaps and lack of end-to-end orchestration. ABC organization decides to outsource this requirement to a xyz service company located in India. The DevOps Architects of the xyz service company have to suggest the tools which are repeatable, predictable and consistent to the ABC Company. As a first step, they need to analyze the existing legacy tools used in the ABC Company. To analyze ABC Company's full architecture they are following three methods of communication.

4.1 Methods of Communication

1. Email Questionnaire
2. Face to Face
3. Telephone

Post discussion with ABC team, xyz identifies the following data.

1. Current process
2. Current Tools list
3. Capabilities evaluation
4. Customer pain point

Once the tools and process are identified from the ABC Company, DevOps architects has to understand if customer is ready to spend money, undergo cultural and process changes else the success of DevOps will be compromised.

4.2 Before proposing a Pipeline

To form a complete successful DevOps tool pipeline from check in to deployment, DevOps architects have to undergo the following steps so that they have complete knowledge on each tool.

1. Market Analysis
2. Tool feasibility/desirability study
3. Cost Analysis
4. Case study

4.2.1 Market Analysis

A survey suggests that DevOps market size is expected to grow USD 10.31 Billion by 2023, at a Compound Annual Growth Rate (CAGR) of 24.7% during the forecast period. [5] DevOps Market analysis plays a vital role in DevOps as day by day there are new tools and as a DevOps engineer it's mandatory to get the insights of each new tool. Approximately, they have to learn a new tool each month to be a successful DevOps practitioner.

4.2.2 Tool feasibility/desirability study:

There are myriad tools available in the market. Nonetheless, without tools, it's hard to automate. Without automation DevOps is imaginary. Thus, tools are a critical part of making the transition. Whenever there is a new tool it is the responsibility of the DevOps engineer to understand the advantages and disadvantages, do a proof of concept, test integration with other tools and become a Subject Matter Expert on the tool. As DevOps Practitioners Knowledge on one tool is not enough. He or she should integrate the newly analyzed tool with the other existing tools in the pipeline. For instance, if there is a new build tool in the market, the tool has to be integrated with Version control Management, Artifactory management and with orchestration tool. Taking a deep dive [4] into all these areas is important as DevOps SME's cannot suggest a tool by just reading a list of features of the tool. They need to figure out how the build tool will work for six months from now or longer for multiple projects in various scenarios. The main take away from the tool feasibility are the answers for the questions in the following figure which will take a considerable amount of time for the DevOps practitioners.
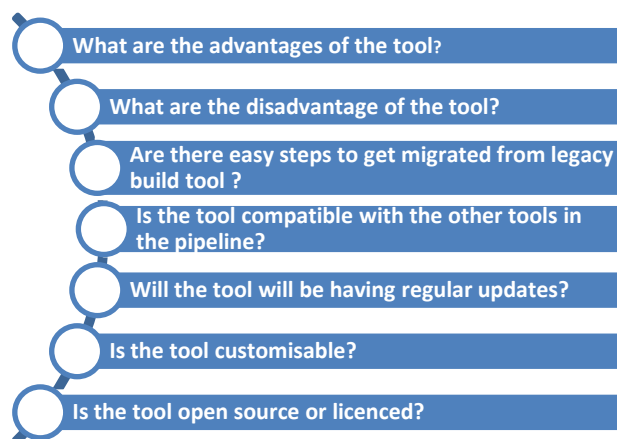

Fig 1: Tools analysis

4.2.3 Cost analysis

DevOps Practitioners are concerned with determining the cost incurred in installing, maintaining the tool and how well these can be re-arranged to increase productivity. As a DevOps engineer, they can subscribe and use a free tier for 30 or 60 days to understand the basic features of the tools but it's not beneficial to get a deep knowledge of the tool. For Example, Some pro version tools may provide only the basic feature and all other complex features may be commercial or it can support via plugins whereas some plugins are readily available in the market and some are still to be developed by the engineers

4.2.4 Case study

A case study is a brief story of enterprises explaining the state before the implementation of the tool, an explanation of what happened post implementation, discussions on conflict and their success stories. As DevOps practitioners have the privilege of installing and testing on a small scale, exploring the case studies of other organizations will benefit in understanding the tool in a more efficient way to provide references to the new customer.

The DevOps practitioner from XYZ Company has been top of all the steps and have identified the gaps of ABC telecommunications and recommended changes to form a better DevOps tools pipeline.

4.3 As-IS state gaps

1. Centralized source code management tool which is not available offline for the developers
2. Manual build with maven which relies on conventions and provides predefined commands
3. No automated test cases to check code coverage and code quality
4. No automatic monitoring, notification and dashboard view to the stakeholders
5. No CI-CD [continuous Integration and continuous delivery]

5. PROPOSED CHANGES

1. Migrate from SVN version management tool to GIT tool for better branching strategy[6]
2. Migrate from Maven tool to Gradle tool for much faster builds
3. Integrate Git, Gradle with Orchestration tool-Jenkins for continuous integration
4. Automate test cases and integrate with Jenkins tool
5. Integrate Sonarqube tool for continuous inspection of code and fortify tool for security
6. Integrate Splunk for monitoring and Hygiea dashboard for a graphical view of all the status
7. Email notification for all the stakeholders
8. Implement Peer review of the code and pull request based deployment
9. Ensure whenever a commit is pushed to the branch, all the continuous integration tests are executed against the code base and automatic mail notification on the build status
10. Automatic promotion of the packages to higher environments.

6. CONCLUSION

This paper has highlighted that each DevOps practitioner has to do Market Analysis, Tool feasibility/desirability study, Cost Analysis and case study manually to provide support to the enterprises and hence lifelong learning is mandatory. Future research is to identify and automate the process to make DevOps practitioner's life easy and fast.

7. REFERENCES

[1] R. Vaasanthi, V. Prasanna Kumari & S. Philip Kingston, "Analysis of DevOps Tools using the Traditional Data Mining Techniques" In International Journal of Computer Applications ISBN : 973-93-80895-78-9 (0975 – 8887) March 2017 DOI:10.5120/ijca2017913319

[2] Abubakerm Osmanan, Majdi, Hu Xiong and Zhiguang, "Toward Unified DevOps Model"

[3] R. Vaasanthi, V. Prasanna Kumari & S. Philip Kingston, "Comparative study of DevOps Build Automation tools" In International Journal of Computer Applications ISBN : 973-93-80895-78-9 (0975 – 8887) July 2017 DOI : 10.5120/ijca2017914908

[4] https://associationsnow.com/2018/03/challenges-choosing-new-software/

[5] https://www.marketsandmarkets.com/Market-Reports/devops-824.html

[6] Migration[Online]http://searchcio.techtarget.com/definiti on/migration