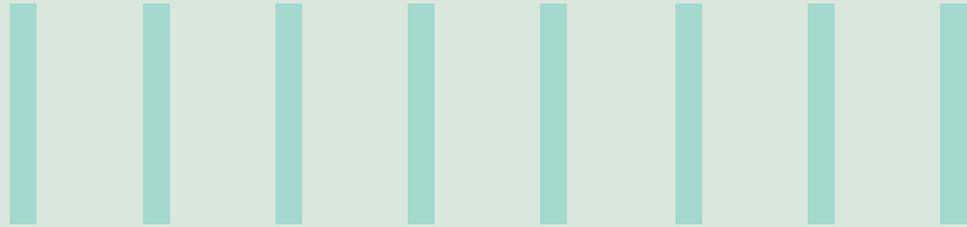




DEVOPS FOR MAINFRAMES – DREAM OR REALITY?

DevOps for IBM Z: Part-1 of 3 part series
An Infosys and IBM Joint Paper



What is DevOps?

DevOps is a term obtained by combining the two terms Development and Operations. As the name suggests it is meant to cover all practices that can help bring both these business units closer. It aims to improve the communication and collaboration between these two units, which were operating as individual units in silos for a long time. DevOps is not just connecting Dev and Ops team, it's a culture to align all participants in the software delivery lifecycle to work towards a shared goal to deliver high-quality and stable software/product more frequently to meet the customer needs. Thus, DevOps is more of a cultural change than processes or products. As per the 2021 Puppet State of DevOps Report¹. The #1 insight is that the majority of organisations are “**stuck in the middle**” level of DevOps performance. The percentage of middle tier DevOps performance organizations has remained remarkably constant over 2018-2021 at around 78%. So why are they stuck? Is it slow cloud adoption? Is it legacy technology? No. The main blockers are cultural. A culture that discourages risk. Lack of clear responsibilities. Most of all, a failure of leadership at the top to promote DevOps, remove barriers and build momentum behind the DevOps implementation efforts.

As DevOps is becoming more and more popular people are either confused or have less clarity on what is the Dev and what is the Ops in DevOps. Dev primarily focuses on software development, testing and application support also share additional responsibility to select application stack, deploying application, monitoring, and resolving application problems. While Ops focus more on manage install, upgrade & configure infrastructure, maintain policies, compliance, and change control, assist application monitoring & deployment and

support other non-production application tasks. DevOps enables operation team own infrastructure and IT policies while empowering the developers to take the ownership and focus more on application development, maintenance, and monitoring.

Today, DevOps term has become an umbrella term for the products (tools), processes, methodologies used to enable faster releases using short & faster feedback loops powered by collaboration and automation.

The major phases in a typical DevOps powered development and release cycles are:

1. Continuous Planning
2. Continuous Development
3. Continuous Integration
4. Continuous Testing
5. Continuous Deployment
6. Continuous Monitoring
7. Continuous Feedback

In this part (**Part-1**), we will cover the introduction to DevOps for Mainframe, the Developer workflow to bring in more agility and reference architecture to DevOps for Mainframe applications.

In **Part-2**, we will cover how to build Continuous Integration and Continuous Deployment pipelines for Mainframe applications, basic details/technical aspects to be considered to build CICD pipelines.

And in the last part (**Part-3**), we will cover the Challenges faced during implementation of DevOps process, best practices to be followed to adopt DevOps for Mainframe applications, Infosys offerings along with DevOps for IBM Z Hybrid Cloud offerings.



DevOps Evolution – A Look Back

DevOps has entered its second decade of existence, standing tall as a key IT phenomenon focused by ~80% of the large enterprises to accelerate deliveries and prepare for the digital savvy future. Back in 2007-2008, Patrick Debois, a Belgian project manager and agile practitioner got frustrated of the huge wall that separated application teams and infrastructure teams. Patrick Debois is known as the father of DevOps as he turned his frustration into fruitful action by joining hands with Andrew Schafer to bring forth the concept of “agile systems administration”. But, the central concept of DevOps as we know it today was highlighted when two Flickr employees – John Allspaw and Paul Hammond in their famous presentation “10+ Deploys per Day: Dev and Ops Cooperation at Flickr.” This historical presentation inspired many to put their thinking caps on and devise unique and customized methods to tackle the problem of disintegrated teams working with different objectives. Since then, there has been no looking back in this journey of DevOps and each year it expands its horizons, seeks better collaboration, better performance and better efficiency of the software development and release life cycles.

The promising results citing the benefits of DevOps published by Google, Amazon, Netflix, General Motors etc. have fueled the momentum of the DevOps adoption across the industry. Thus, gone are the days when organizations asked WHY or IF we should adopt DevOps. Instead the big question now to be answered is WHAT and HOW for many enterprises in their nascent stages of the DevOps adoption. Initially, the idea was just to break the silo between Dev and Ops, but today it encompasses each stage in the Development life cycle, from a novel idea to the product on the end users table. Thus, we will see patterns like Opsfication of Dev, Devification of Ops, Devsecops, Devsecbizops, Dataops, Archops to finally NoOps and we need to wait and watch to know what the next is.



DevOps for Mainframes – Introduction

IBM Mainframe continues to drive the world's economy as the core infrastructure for business-critical applications for more than six decades. Mainframe system drives credit card transactions worth \$7.7 trillion every year², caters to more than 30 billion transactions daily and hosts close to 70% of worldwide production workloads.

Below mentioned are few additional data points highlighting the popularity of IBM Mainframes³:

- 80% of Fortune-500 companies use Mainframes
- 92% of the top 100 banks use Mainframes
- 100% of the top 10 insurance companies use Mainframes
- 92% of the top 25 US retailers use Mainframes
- 90% of the credit card transactions are processed by Mainframes
- 70% of world's business data is stored in Mainframes

Additionally, Mainframe usage is not going down as perceived, as per "2020 BMC The largest survey⁴" of its kind, this bellwether study queried over 1000 executives and practitioners on their priorities, challenges, and growth opportunities for the platform. High-level insights include:

- 90% of respondents see the mainframe as a platform for growth and long-term applications.
- 68% expect MIPS, the mainframe's measure of computing performance, to grow.
- 63% of respondents say security and compliance were their top mainframe priorities.
- More than half of respondents (54 percent) reported an increase in transaction volume and 47 percent reported an increase in data volumes.

"The Mainframe Survey validates that businesses see the mainframe as a

critical component of the modern digital enterprise and an emerging hub for innovation," says Stephen Elliot, Program Vice President, Management Software and DevOps, IDC. "They're putting it to work more and more to support digital business demands as they strive to achieve greater agility and success across the enterprise."

Top Mainframe Priorities: with mainframe enterprises competing to bring new, digital experiences to market to delight customers, the survey's themes are resoundingly strong: **Adapt, Automate, and Secure.**

Adapt: New processes to keep up with digital demand. Technology demands such as application development/DevOps across the mainframe; 78% of respondents want to be able to update mainframe applications more frequently than currently possible.

Automate: Mainframe modernization continues to play a key role in priorities among respondents with the need to implement AI and machine learning strategies jumping by 8% year over year.

Secure: While the mainframe has a reputation of being a naturally secure platform, respondents are seeing the growing need to fortify its "walls." Security trumped cost optimization as the leading mainframe priority among respondents for the first time in the 15-year history of the survey.

Mainframe with its virtually limitless scale, superior efficiency, trusted computing with high level of security is at the center of most of the large financial enterprises. But, there are challenges that makes developing and maintaining applications on Mainframe difficult. As the Mainframe code base has been built and re-built over decades, monolithic and spaghetti code base is a prevalent phenomenon. Thus, impact analysis and end-to-end testing of these complex applications takes enormous amount of time. The tools used for development, build, test and deploy are outdated and cumbersome. In addition, most of these tasks are manual and thus people dependent, which cannot

assure the quality of the code delivered. Adding to the woes is the fact that over the years, the gap between development teams and operations has widened so much that each works as a separate entity without any common goal. All this leads to high cycle times and fewer releases. Thus, agility and lack of the niche skill set required to maintain mainframe applications is the key challenge in front of these Mainframe heavy enterprises. There is an assumption that modernizing Mainframe platform is not a viable option and modern development methodologies like Agile, Spiral etc. are not feasible for Mainframe based applications.

However, the fact is that modern day application development methodologies that are foundational to the distributed world is possible largely in the Mainframe world also. These modern-day tools can seamlessly integrate with the Mainframe system to build a strong, coherent DevOps CI-CD pipeline. Bringing DevOps to Mainframe system will create a ripple effect and a paradigm shift in the thinking process of large Mainframe enterprises. Changes are now delivered faster via a synchronous, high quality, controlled and monitored DevOps pipeline.

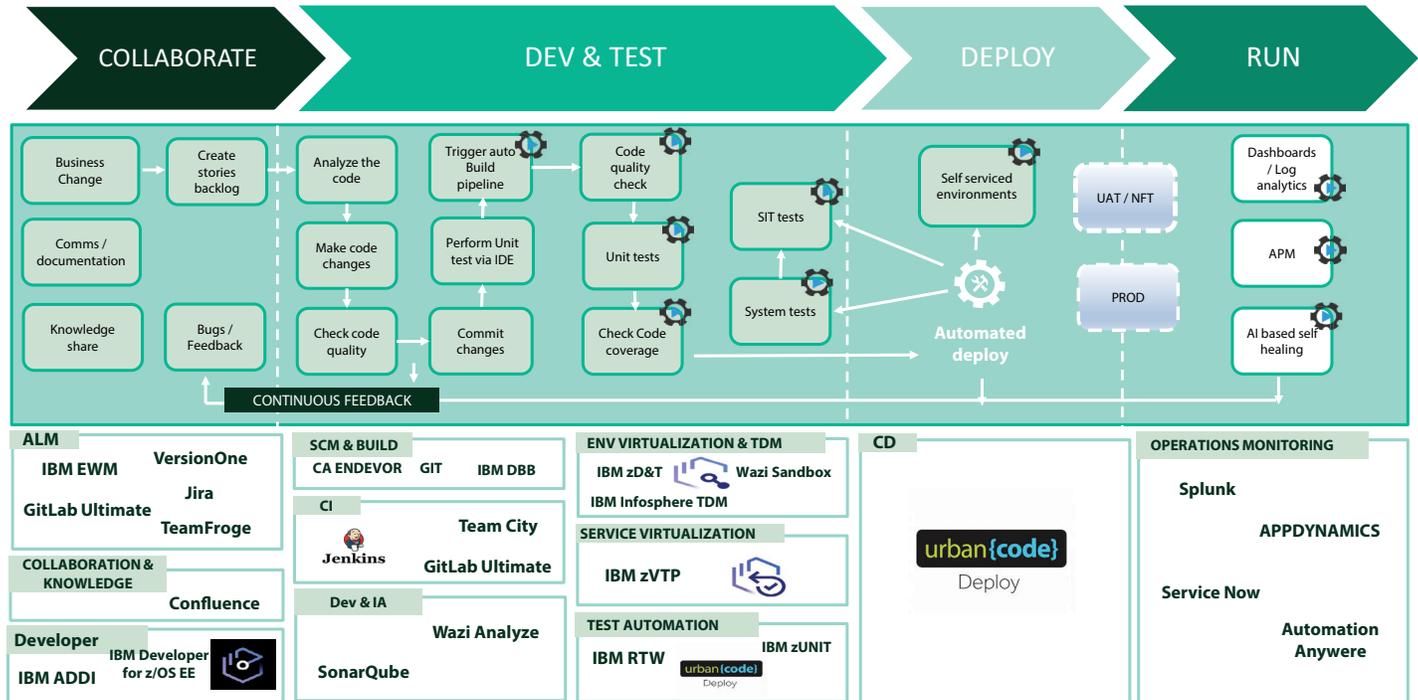
Yes, DevOps on Mainframes is necessary for large Mainframe enterprise. However, we should also be aware that DevOps is not a magic, a DevOps transformation will not happen overnight. One doesn't have to wait for a large-scale management initiative to kick-off to start with DevOps. By making small, incremental changes to embark on the journey and reap more and more benefits as the journey goes along. There is plethora of options available today, it is advisable to collaborate with an experienced system integrator in planning and making judicious decisions.

It's important that to do a thorough due diligence to strategize the DevOps transformation journey, finalize the tools, roadmap the path ahead, support in adoption and work towards continuous improvement.

DevOps for Mainframes – Reference Architecture

DevOps is all about continuity and thus it cannot be tied to any one stage of development lifecycle. This is no different for the Mainframe platform. Thus, true DevOps is only when all the stages are automated, and the component changes can go from development to production in a continuous pipeline without any hiccup or manual intervention. However, this might seem a far-fetched dream for Mainframe platform today, but we can start in small steps and eventually reach there.

Below is a sample architecture diagram of a Mainframe based DevOps pipeline with its typical tools:



In this paper, we try to detail each of these stages from Mainframe's preview and share our learnings gathered in each of these stages.

Developer Workflow

As organizations on-board next gen developers, it's important that they are provided with tools that they are familiar with.

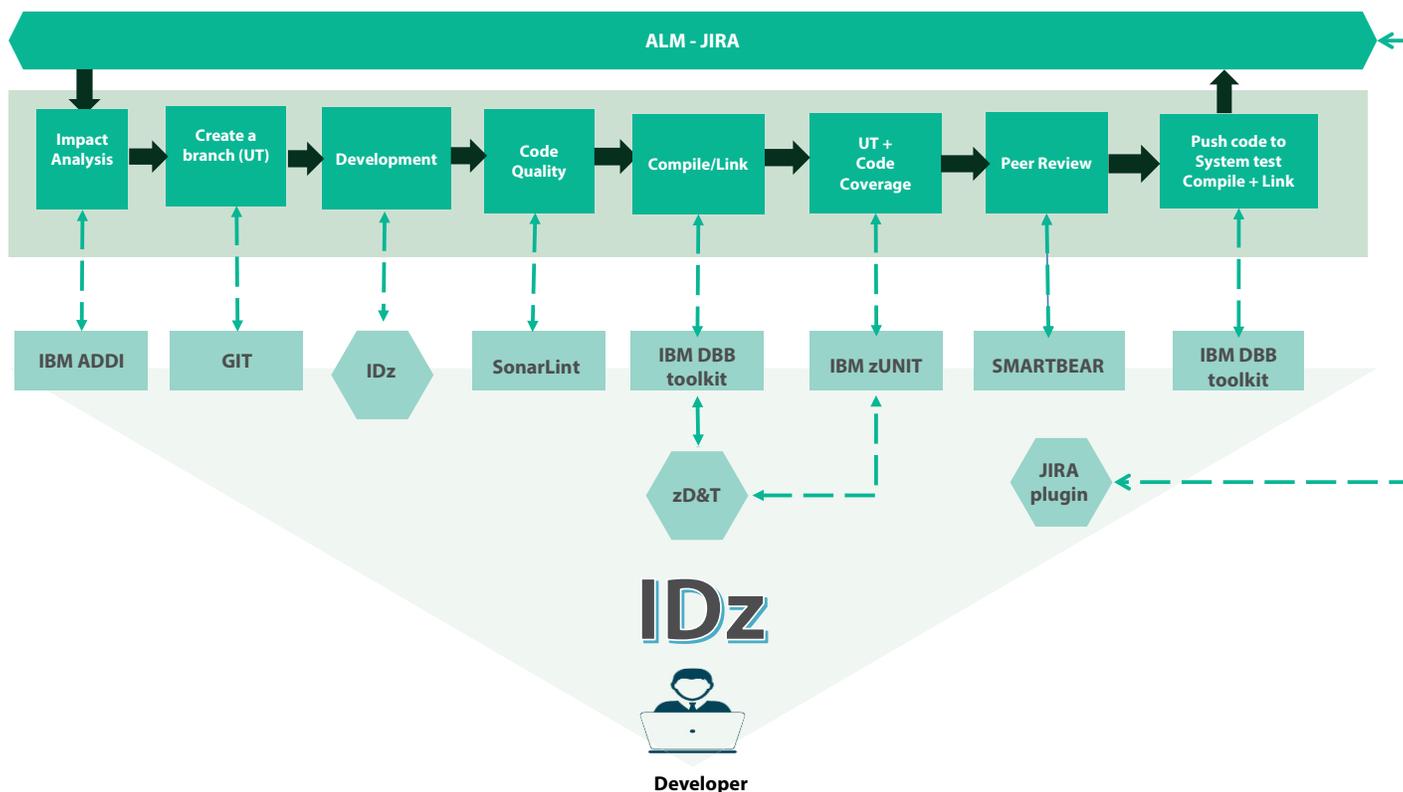
Developers use for their day-to-day development (coding and unit testing) activities constitute the Developer Workflow. It is important that the young developers are given an opportunity to use these modern tools for their development activities.

Typical tools used in this space are for:

- *Integrated Development Environment (IDE):* These are modern development

toolkits that enable developers to get away from the old school of using ISPF and the green screens for mainframe application development. IBM provides a choice of IDEs for modern day developers to develop and maintain z/OS applications. BringYourOwnIDE (BYOIDE) is available with IBM IDz Enterprise Edition (IDzEE) and supports VSCode and Codeready Workspaces. IBM IDz is the most popular among IDEs for z/OS applications developments. In this part we will focus on IDz IBM Developer for z/OS (IDz) is an Eclipse-based IDE. Along with several plugins, it can expedite the application development and enhancement through features like automated syntax and standard violation checks, modern debugging and integration with ALM, SCM and Build tools.

- *Automated Unit Testing:* Testing constitutes of the major effort in any software development lifecycle. It is a difficult activity in the mainframe landscape. Tools like ZUnit, an automated unit testing framework, which can automate unit testing for batch and CICS based online modules.
- *Tool assisted Impact Analysis:* Mainframe applications are decades old and the applications are built with high inter-dependencies, this makes it difficult for a manual impact analysis. In addition, lack of accurate documentation increases the complexity for performing a bottom up analysis of the system. IBM's Application Discovery & Delivery Intelligence (ADDI) can be integrated with IDz and is used for business rule discovery and visual analysis/ understanding of the system.



This concludes the **Part-1** of this series.

Part-2 covers on how to build Continuous Integration and Continuous Deployment pipelines for Mainframe applications, basics details/ technical aspects to be considered to build CICD pipelines.

Part-3 covers about the Challenges faced during implementing DevOps process, best practices to be followed to adopt DevOps for Mainframe application, Infosys offerings along with IBM for Mainframe DevOps.

References

1. State of DevOps Report 2021 | Puppet
2. Company - Cloud data management for Mainframe (model9.io)
3. 9 Mainframe Statistics That May Surprise You - Precisely
4. 2020 BMC Mainframe Survey Results | BMC Software, Inc.



About the Authors

Elango Paulraj

Technology Architect – Legacy Modernization

Elango has vast experience in Mainframe Modernization, MIPS Optimization & Performance tuning, DevOps for Mainframe and Tools & Product evaluation. Currently working across various customers for providing solutions, options in Mainframe Modernization & building CI/CD pipeline in mainframe.

Hardik Dave

Technology Architect – Legacy Modernization

Hardik is Legacy Modernization Professional with experience in Mainframe Integration, DevOps and APIs. He is a certified TOGAF Enterprise Architect and has worked on several Legacy Integration Projects, Core Banking Transformation Programs and API Enablement Projects.

Madesh Ramamoorthy

Technology Architect – Legacy Modernization

Madesh is a Technology Architect in Infosys Ltd having expertise in mainframe environment virtualization using zD&T & zVMs and implementing end-to-end CI-CD pipeline for Mainframe applications. He is instrumental in bringing DevOps on Mainframes for one of the leading banks in UK.

Manas Pani

Technology Architect – Legacy Modernization

Manas is a member of Legacy Modernization Centre of Excellence and is currently leading the Mainframe DevOps adoption in a leading Banking and Financial services holding company in US.

Nirmalskandan Natarajan

Technology Architect – Legacy Modernization

Nirmalskandan has over 3 years of experience in implementing DevOps On Mainframes. This includes various tools evaluation for IDE, Unit and Integration testing framework, CI tool and Static code analyzer. He has also been trainer and instrumental in various tool adoption and training in the organization on these tools.

Special thanks to all the reviewers for their guidance!!

Srinivas J

AVP and Unit Technology Officer
(Infosys Ltd)

Vijayeendra S N

Product Management Leader, DevOps for
IBM Z Hybrid Cloud (IBM)

Sivaramkrishna K

Senior Principal Technology Architect
(Infosys Ltd)

Bivas Banerjee

Principal Technology Architect
(Infosys Ltd)

Anjali Abraham

(Z DevOps Solution Architect)
(IBM)

Chetan Ramamurthy

Principal Consultant
(Infosys Ltd)

For more information, contact askus@infosys.com



© 2021 Infosys Limited, Bengaluru, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/ or any named intellectual property rights holders under this document.