- **ASSIGNMENT**: 1.2 - THE TECHNOLOGY VALUE STREAM OF (MODULE 1: THE HISTORY OF DEVOPS)
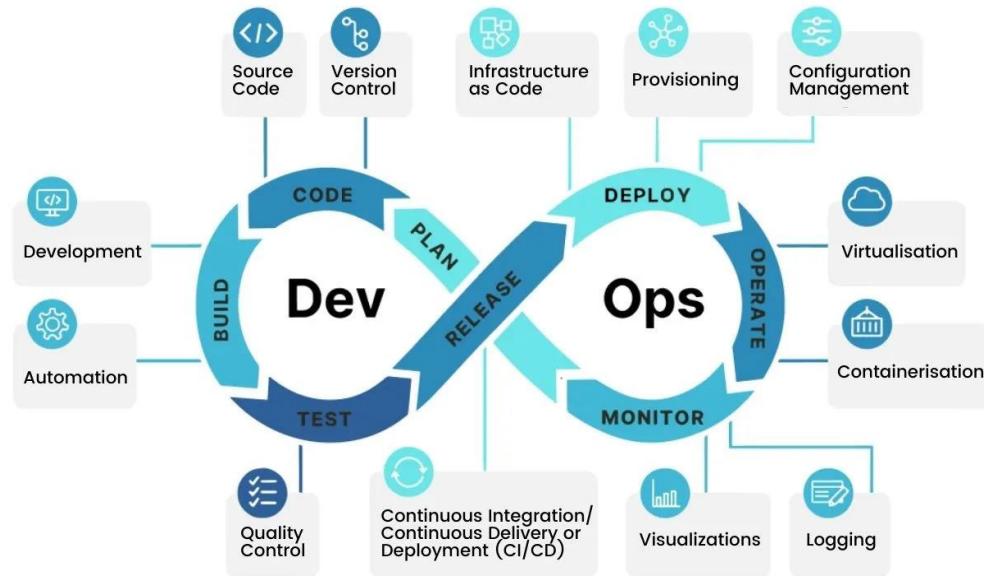- **NAME**: ARUN SHARMA
- **COURSE**: CSD380-H326 DEVOPS
- **DATE**: 01/12/2025

OVERVIEW

The Technology Value Stream is a central concept in DevOps, aiming to optimize how value is delivered to customers through technology. This presentation explores three key aspects of the value stream: understanding lead time versus processing time, analyzing the common scenario of long deployment lead times, and envisioning the DevOps ideal of reducing deployment lead times to minutes. By addressing these areas, we can uncover inefficiencies in traditional workflows and identify opportunities for improvement. Lead time and processing time form the foundation for analyzing workflow bottlenecks, highlighting areas where automation and collaboration can make a difference. Deployment lead times requiring months demonstrate the challenges faced by organizations relying on manual processes and siloed teams. Finally, the DevOps ideal emphasizes the benefits of automation, breaking down silos, and achieving rapid, reliable deployments. This presentation aims to provide actionable insights into optimizing the value stream, enhancing both organizational efficiency and customer satisfaction.

DEFINING LEAD TIME VS. PROCESSING TIME

Lead time and processing time are critical metrics for understanding and optimizing workflows within the technology value stream. Lead time is the total duration from when a customer request is made to when the product or feature is delivered, encompassing all waiting, development, and testing periods. In contrast, processing time measures only the active work done to fulfill the request, such as coding, testing, or deploying the feature. For example, if a customer requests a new feature, lead time includes the days the feature spends in a task queue awaiting assignment, while processing time accounts for the actual development and testing efforts. The distinction between these two metrics is vital because lead time often reveals bottlenecks caused by inefficiencies like task backlogs or manual approvals. DevOps practices aim to reduce these inefficiencies by streamlining workflows, automating repetitive tasks, and fostering collaboration. This focus on reducing lead time while maintaining or improving processing time is essential for delivering value to customers quickly and reliably.
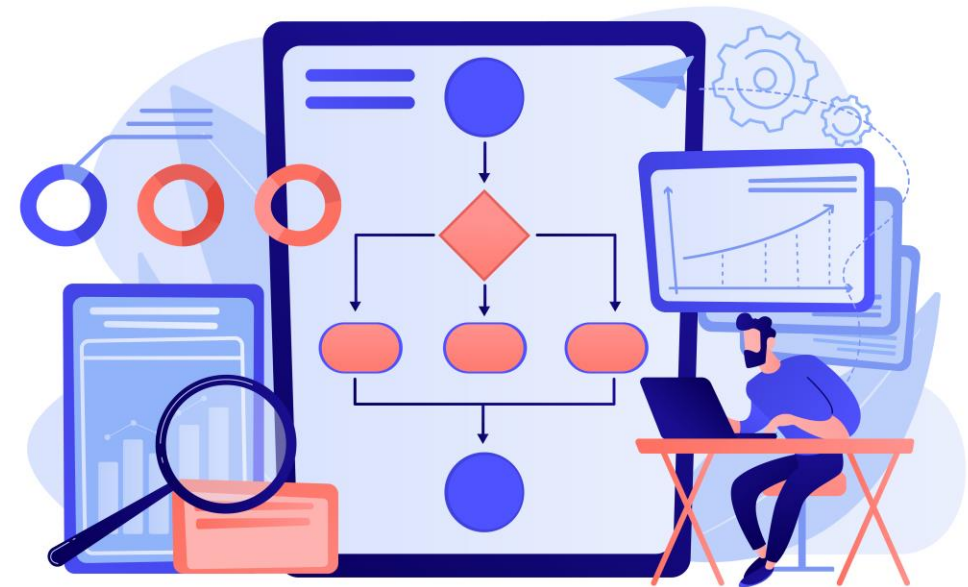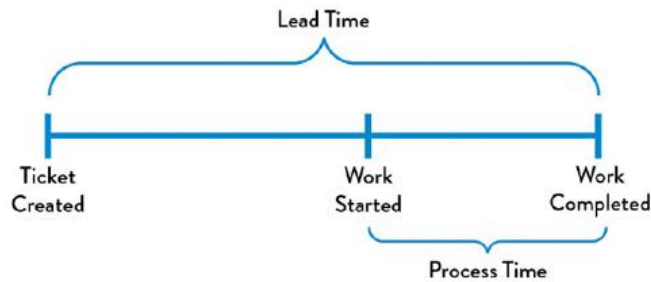
LEAD TIME VS. PROCESSING TIME

Figure 2. Lead time vs. process time of a deployment operation

The relationship between lead time and processing time is best understood through a visual representation of a typical workflow. Imagine a task progressing through stages such as development, testing, and deployment. Lead time spans the entire journey from request to delivery, including periods of waiting or inactivity, while processing time captures only the periods when active work is performed. For instance, a feature request may have a lead time of seven days, including five days waiting in a backlog and two days of active development and testing. A timeline or flowchart depicting these stages helps highlight where delays occur and how DevOps practices can minimize them. By focusing on reducing lead time without compromising processing time, organizations can deliver features faster while maintaining quality. This visual comparison underscores the importance of identifying and eliminating bottlenecks, such as manual approvals or communication silos, which significantly extend lead time.
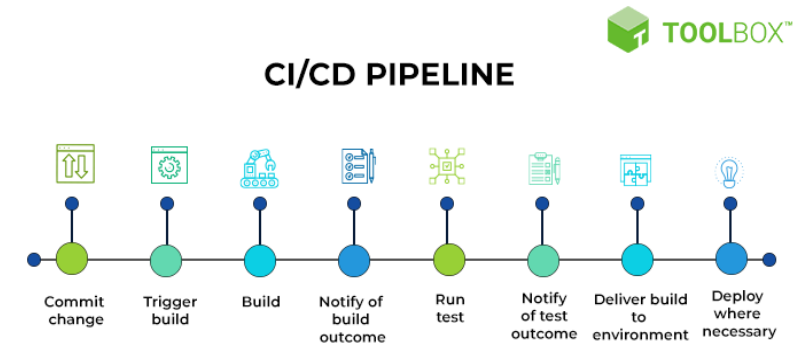
THE COMMON SCENARIO: DEPLOYMENT LEAD TIMES REQUIRING MONTHS

- Traditional workflows often struggle with deployment lead times that extend into months, a challenge many organizations face. These long lead times arise from a combination of inefficiencies, including siloed teams, manual processes, and limited automation. For example, when development, testing, and operations teams work in isolation, handoffs between them cause significant delays. Similarly, manual testing and deployment processes introduce bottlenecks, further extending the time needed to deliver new features. As a result, organizations experience delayed product launches, reduced responsiveness to customer needs, and lower overall productivity. This scenario highlights the limitations of traditional workflows, where the absence of streamlined processes impedes innovation and efficiency. Addressing these issues requires adopting practices that reduce friction between teams, automate repetitive tasks, and create a culture of collaboration and continuous improvement. Understanding these common challenges sets the stage for exploring how DevOps can transform workflows and achieve shorter deployment lead times.

1980

1990

2000

2010

2020

es g ec by freepik
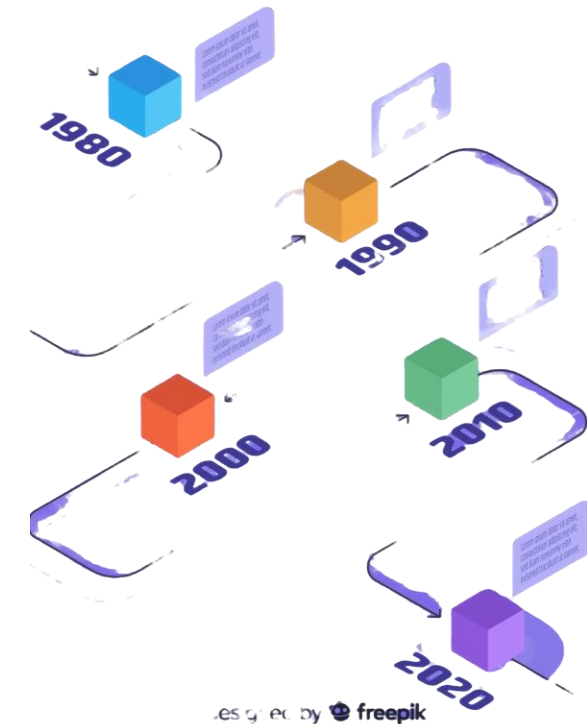
OUR DEVOPS IDEAL: DEPLOYMENT LEAD TIMES OF MINUTES

The DevOps ideal transforms traditional workflows by reducing deployment lead times from months to minutes. This shift is achieved through practices like automation, continuous integration, continuous delivery (CI/CD), and fostering collaboration between development and operations teams. By automating tasks such as testing and deployment, DevOps eliminates manual bottlenecks that delay progress. Small, frequent deployments replace large, infrequent ones, reducing the risk of errors and enabling rapid feedback. For example, organizations using DevOps can deploy multiple updates daily, ensuring quicker response times to customer needs and market changes. Collaboration is another critical component, as breaking down silos between teams promotes shared responsibility for delivering value. The result is a workflow that prioritizes speed, quality, and reliability. Achieving the DevOps ideal requires a cultural shift toward embracing change, continuous learning, and leveraging tools that support automation and integration. This ideal not only enhances efficiency but also delivers greater satisfaction for customers and team members alike.

TOOLBOX™

CI/CD PIPELINE

| Commit change | Trigger build | Build | Notify of build outcome | Run test | Notify of test outcome | Deliver build to environment | Deploy where necessary |

TRADITIONAL VS. DEVOPS WORKFLOW

The comparison between traditional workflows and DevOps workflows highlights the transformative potential of DevOps practices. In traditional workflows, tasks move sequentially through stages such as development, testing, and deployment, with long waiting periods at each stage. Manual approvals, lack of automation, and siloed teams create bottlenecks that significantly extend lead times. In contrast, DevOps workflows are characterized by automation, collaboration, and real-time feedback. Continuous integration and delivery pipelines automate testing and deployment, ensuring faster and more reliable releases. Teams work together across development, operations, and testing, reducing delays caused by miscommunication or handoffs. A visual flowchart comparing these workflows illustrates how DevOps streamlines processes by minimizing waiting times and eliminating redundant steps. This streamlined approach enables organizations to respond quickly to customer demands, improve productivity, and maintain a competitive edge in the market.
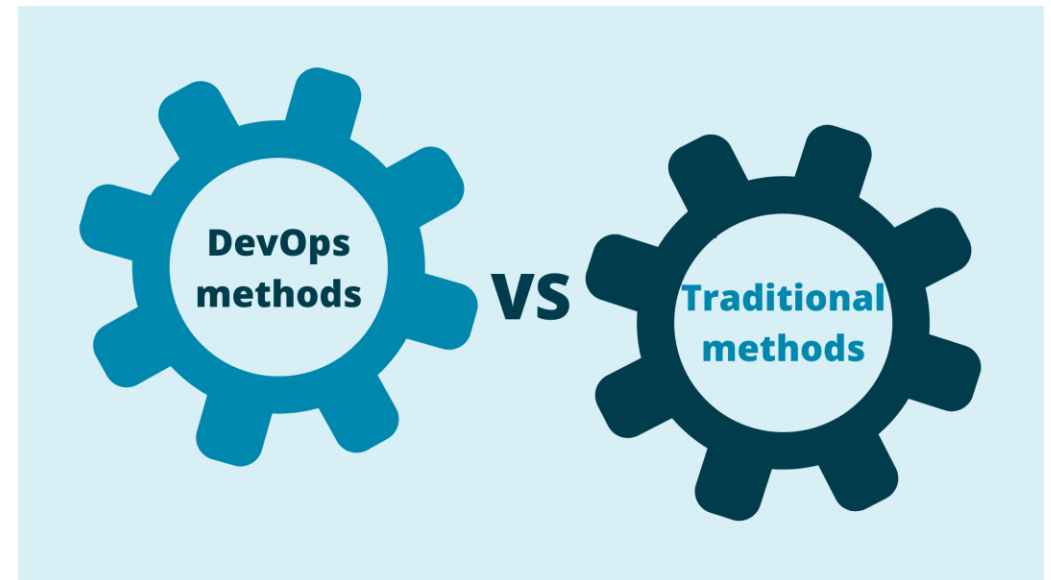
BENEFITS OF OPTIMIZING THE VALUE STREAM

Optimizing the technology value stream brings numerous benefits that enhance organizational performance and customer satisfaction. First, reducing lead times allows for faster delivery of features and updates, ensuring that organizations can respond quickly to market demands. Second, fostering collaboration between teams breaks down silos, enabling smoother workflows and improved communication. Third, automation of repetitive tasks reduces human error and enhances the quality of deployments. For example, automated testing ensures that code changes are thoroughly validated before release, resulting in fewer defects. Finally, optimizing the value stream improves customer satisfaction by delivering high-quality products faster and more reliably. These benefits not only improve operational efficiency but also help organizations maintain a competitive edge. By adopting DevOps practices that prioritize continuous improvement and collaboration, organizations can achieve significant gains in both productivity and customer experience.

CHALLENGES IN ACHIEVING DEVOPS IDEAL

While the DevOps ideal offers numerous benefits, achieving it is not without challenges. Resistance to change is a common obstacle, as teams may be hesitant to adopt new tools and workflows. For example, moving from manual processes to automation requires a shift in mindset and significant training. Another challenge is the lack of necessary skills or resources, as implementing DevOps practices often demands expertise in areas like CI/CD, cloud technologies, and monitoring tools. Organizational silos further complicate the transition, as teams must learn to collaborate and share responsibility for delivery. Overcoming these challenges requires a strategic approach, starting with pilot projects that demonstrate the value of DevOps. Providing training and resources helps build the required skills, while fostering a culture of collaboration and continuous learning encourages teams to embrace change. By addressing these challenges systematically, organizations can successfully transition to the DevOps ideal and reap its full benefits.
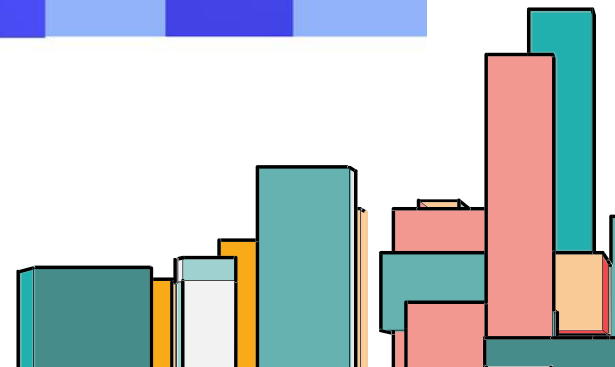
CONCLUSION

The technology value stream is a critical framework for understanding and optimizing how organizations deliver value to their customers. through technology. By examining lead time versus processing time, we can identify inefficiencies that hinder workflow efficiency. Traditional deployment lead times, often measured in months, demonstrate the challenges posed by siloed teams, manual processes, and a lack of automation. In contrast, the DevOps ideal of achieving deployment lead times measured in minutes highlights the transformative power of automation, continuous integration, and team collaboration.

DevOps not only improves deployment speed but also enhances quality and customer satisfaction. By breaking down silos and fostering a culture of shared responsibility, organizations can streamline workflows and ensure faster, more reliable deliveries. The benefits of this optimization are clear: improved productivity, higher customer satisfaction, and a competitive edge in the market.

However, achieving the DevOps ideal requires addressing significant challenges, including resistance to change, skill gaps, and organizational silos. With a strategic approach that includes training, pilot projects, and cultural shifts, these challenges can be overcome.

Ultimately, the adoption of DevOps practices empowers organizations to align their workflows with business goals, ensuring that technology truly delivers value. As we continue to embrace these principles, the potential for innovation and growth in the technology value stream is limitless.

REFERENCES

Agile Alliance. (2023, October 18). Agile Manifesto for Software Development | Agile Alliance. Agile Alliance |. https://www.agilealliance.org/agile101/the-agile-manifesto/

History: The Agile Manifesto. (n.d.). https://agilemanifesto.org/history.html?

Lean Enterprise Institute. (2023, March 31). A brief history of lean - lean Enterprise Institute. https://www.lean.org/explore-lean/a-brief-history-of-lean/

Pais, M. (2014, July 21). The birth of continuous delivery and DevOps. InfoQ. https://www.infoq.com/news/2014/07/birth-cd-devops/

What is Lean | History and early development. (n.d.). The Lean Way. https://theleanway.net/what-is-lean?

Wikipedia contributors. (2024, December 4). Continuous delivery. Wikipedia. https://en.wikipedia.org/wiki/Continuous_delivery

THANK YOU

Arun Sharma