

## Introduction to DevOps

*DevOps is basically a combination of two words- Development and Operations. DevOps is a culture that implements the technology to promote collaboration between the developer team and the operations team to deploy code to production faster in an automated and repeatable way.*

### Why DevOps?

The goal of DevOps is to increase an organization's speed when it comes to delivering applications and services. Many companies have successfully implemented DevOps to enhance their user experience including Amazon, Netflix, etc.

Facebook's mobile app which is updated every two weeks effectively tells users you can have what you want and you can have it. Now ever wondered how Facebook was able to do social smoothing? It's the DevOps philosophy that helps Facebook ensure that apps aren't outdated and that users get the best experience on Facebook. Facebook accomplishes this true code ownership model that makes its developers responsible that includes testing and supporting through production and delivery for each kernel of code. They write and update their true policies like this but Facebook has developed a DevOps culture and has successfully accelerated its development lifecycle.

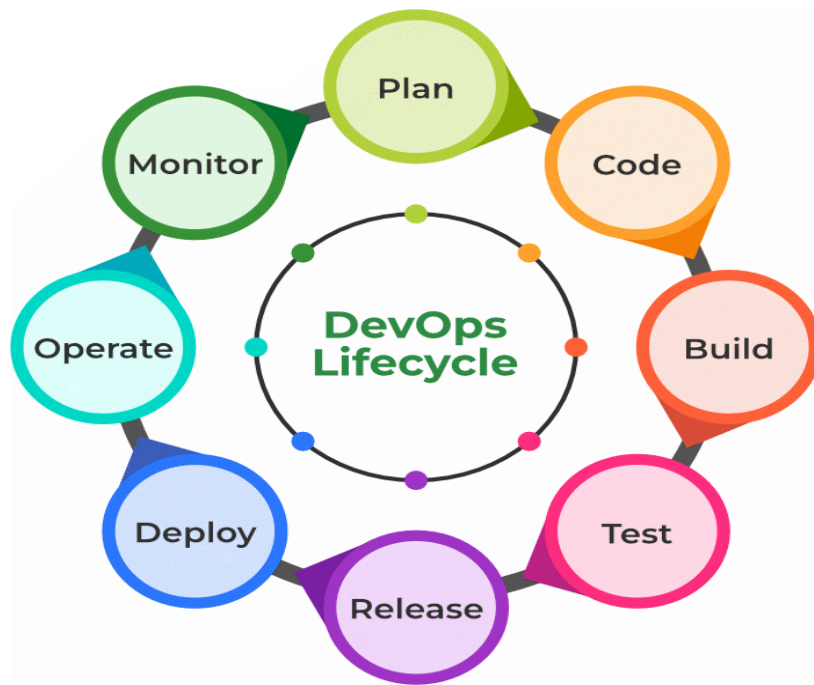
Industries have started to gear up for digital transformation by shifting their means to weeks and months instead of years while maintaining high quality as a result. The solution to all this is- DevOps.

### How DevOps is different from Traditional IT?

Traditional IT has 1000s lines of code and is created by different teams with different standards whereas DevOps is created by one team with intimate knowledge of the product. Traditional IT is complex to understand and DevOps is easily understandable.

### DevOps Lifecycle

DevOps lifecycle is the methodology where professional development teams come together to bring products to market more efficiently and quickly. The structure of the DevOps lifecycle consists of Plan, Code, Building, Test, Releasing, Deploying, Operating, and Monitoring.



**Plan:** Determining the commercial needs and gathering the opinions of end-user by professionals in this level of the DevOps lifecycle.

**Code:** At this level, the code for the same is developed and to simplify the design, the team of developers uses tools and extensions that take care of security problems.

**Build:** After the coding part, programmers use various tools for the submission of the code to the common code source.

**Test:** This level is very important to assure software integrity. Various sorts of tests are done such as user acceptability testing, safety testing, speed testing, and many more.

**Release:** At this level, everything is ready to be deployed in the operational environment.

**Deploy:** In this level, Infrastructure-as-Code assists in creating the operational infrastructure and subsequently publishes the build using various DevOps lifecycle tools.

**Operate:** At this level, the available version is ready for users to use. Here, the department looks after the server configuration and deployment.

**Monitor:** The observation is done at this level that depends on the data which is gathered from consumer behavior, the efficiency of applications, and from various other sources.

**Best Practices to follow:**

- Implement automated dashboard
- Keep the entire team together
- Allow DevOps to be a cultural change
- Be patient with the developers
- Maintain a centralized unit
- Build a flexible infrastructure

**Advantages:**

**Faster Delivery:** DevOps enables organizations to release new products and updates faster and more frequently, which can lead to a competitive advantage.

**Improved Collaboration:** DevOps promotes collaboration between development and operations teams, resulting in better communication, increased efficiency, and reduced friction.

**Improved Quality:** DevOps emphasizes automated testing and continuous integration, which helps to catch bugs early in the development process and improve the overall quality of software.

**Increased Automation:** DevOps enables organizations to automate many manual processes, freeing up time for more strategic work and reducing the risk of human error.

**Better Scalability:** DevOps enables organizations to quickly and efficiently scale their infrastructure to meet changing demands, improving the ability to respond to business needs.

**Increased Customer Satisfaction:** DevOps helps organizations to deliver new features and updates more quickly, which can result in increased customer satisfaction and loyalty.

**Improved Security:** DevOps promotes security best practices, such as continuous testing and monitoring, which can help to reduce the risk of security breaches and improve the overall security of an organization's systems.

**Better Resource Utilization:** DevOps enables organizations to optimize their use of resources, including hardware, software, and personnel, which can result in cost savings and improved efficiency.

Disadvantages:

**High Initial Investment:** Implementing DevOps can be a complex and costly process, requiring significant investment in technology, infrastructure, and personnel.

**Skills Shortage:** Finding qualified DevOps professionals can be a challenge, and organizations may need to invest in training and development programs to build the necessary skills within their teams.

**Resistance to Change:** Some employees may resist the cultural and organizational changes required for successful DevOps adoption, which can result in resistance, resistance to collaboration, and reduced efficiency.

**Lack of Standardization:** DevOps is still a relatively new field, and there is a lack of standardization in terms of methodologies, tools, and processes. This can make it difficult for organizations to determine the best approach for their specific needs.

**Increased Complexity:** DevOps can increase the complexity of software delivery, requiring organizations to manage a larger number of moving parts and integrate multiple systems and tools.

**Dependency on Technology:** DevOps relies heavily on technology, and organizations may need to invest in a variety of tools and platforms to support the DevOps process.

**Need for Continuous Improvement:** DevOps requires ongoing improvement and adaptation, as new technologies and best practices emerge. Organizations must be prepared to continuously adapt and evolve their DevOps practices to remain competitive.