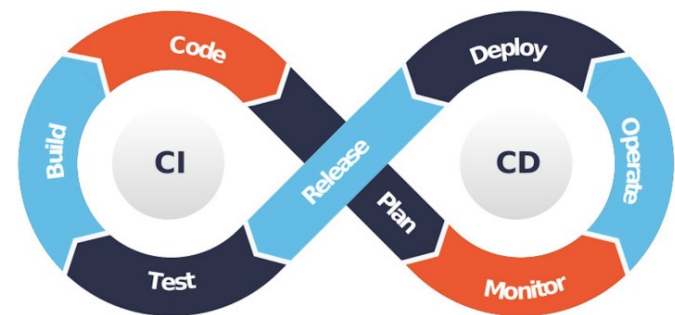


Fundamentals and Benefits of CI/CD



Overview

- CI/CD Introduction
- CI/CD Pipelines
- Benefits of CI/CD
- Challenges
- CI/CD Solution

Introduction

- Continuous Integration (CI) and Continuous Deployment (CD) are fundamental practices in modern software development that streamline the process of building, testing, and deploying software.
- These practices are designed to overcome the challenges of traditional software delivery methods, where manual interventions, slow releases, and lack of collaboration often hinder the development process.

CI/CD

- **Continuous Integration**

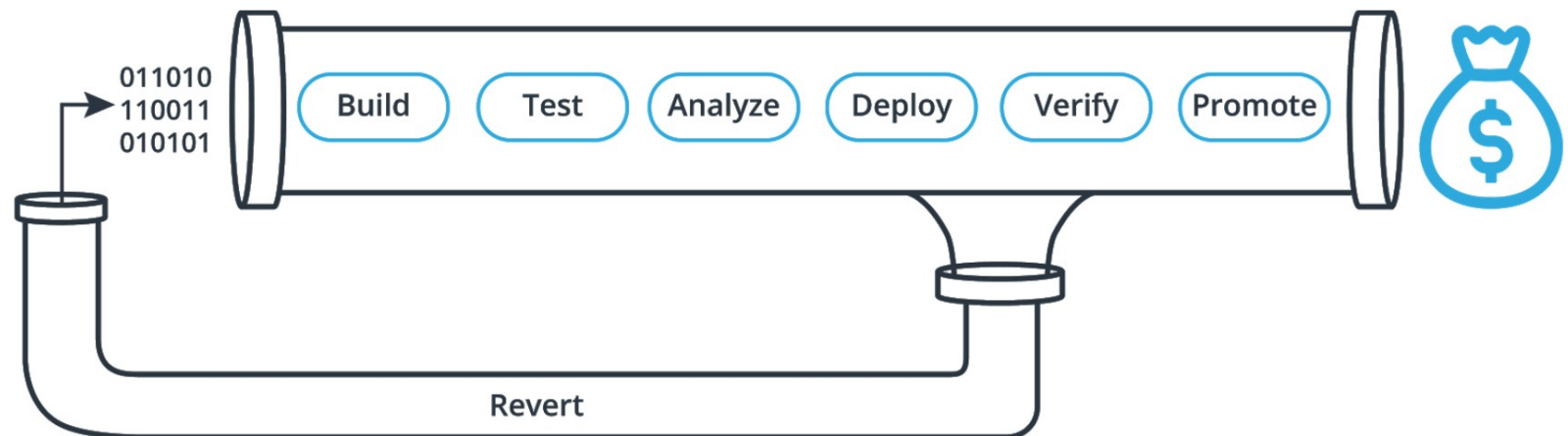
- Focuses on frequent and automated integration of code changes into a shared repository.
- CI fosters collaboration, reduces integration headaches, and accelerates the development cycle.

- **Continuous Deployment**

- CD extends the CI concept by automating the deployment of successfully tested code changes to production.
- It ensures that software is always deployable, reducing the time and effort required to release new features or fixes.
- It aims to achieve a reliable and efficient process for delivering software to end-users.

CI/CD pipeline

- CI/CD pipeline is combination of steps to be completed to deliver a new release.



Benefits CI/CD

- **Faster Releases**
 - CI/CD enables frequent and predictable releases, speeding up the delivery of new features and improvements.
- **Reduced Risks**
 - Automated testing catches bugs early, minimizing the chances of defects reaching production.
- **Enhanced Collaboration**
 - CI/CD fosters collaboration between development and operations teams, breaking down silos.
- **Flexibility**
 - Automated deployment and rollbacks facilitate quick responses to changing requirements.
- **Continuous Feedback**
 - Regular feedback loops improve software quality and user satisfaction.

Challenges

- Delays in production deployments
 - Due to manual and error-prone release process, affects release time lines.
- No code consistency
- No proper Rollback mechanisms
- Manual Infrastructure setup and clean up
- Inconsistent environment configurations
- Conflicting goals between development and operation teams
- Complex Deployments

CI/CD Solution

- CI/CD code changes are simpler/smaller and deployments are automated.
- Fault isolation is easy and quick
- Integrated common repository
 - CI/CD pipeline enables developers to integrate their code into a common repository in smaller batches
- Using CI/CD can improve test reliability to a great extent.
- CI/CD improves overall communication and accountability between team members.
- CICD is able to automate infrastructure creation
 - leading to no human error or Inconsistent environment configurations
 - faster deployments.