



The Problem

Challenges often encountered in our software Development process can be summarised below;

- ❖ We invest more time in a release cycle than delivering value
- ❖ We go through integration hell every time we finish a feature
- ❖ We face a lot of merge issues
- ❖ We ignore writing units tests because we inherit a lot of code with failing tests cases
- ❖ Deployments usually contribute to missed deadlines
- ❖ Too much friction between development and operations department
- ❖ Only few people really know how to deploy a system.
- ❖ Deployments are not usually a call for celebration because no client satisfaction.



The Solution - CI/CD

Best practice of achieving synchronization between development and operations departments is through Continuous Delivery. Continuous Delivery is a paradigm where engineering teams produce and release values in short circles.

Continues Delivery can be practiced through Continuous Integration and Continuous Deployment (CI/CD).

What is Continuous Integration (CI)?

CI is the practice of merging all developers' **working copies** in a shared mainline several times a day. This is the process of “**Making**” the software. The ultimate goal of CI is a ***high quality, deployable artifact!***

Phases in the “Making” of the software may include

- Compiling
- Unit Testing
- Static Software Analyses
- Security Vulnerability Testing
- Storing generated artifact.



The Solution - CI/CD

What is Continuous Deployment (CD)

CD is a software engineering approach in which the value is delivered frequently through automated deployments. Everything related to deploying the artifact fits here! It is the process of “**Moving**” the artifact from shelf to spotlight.

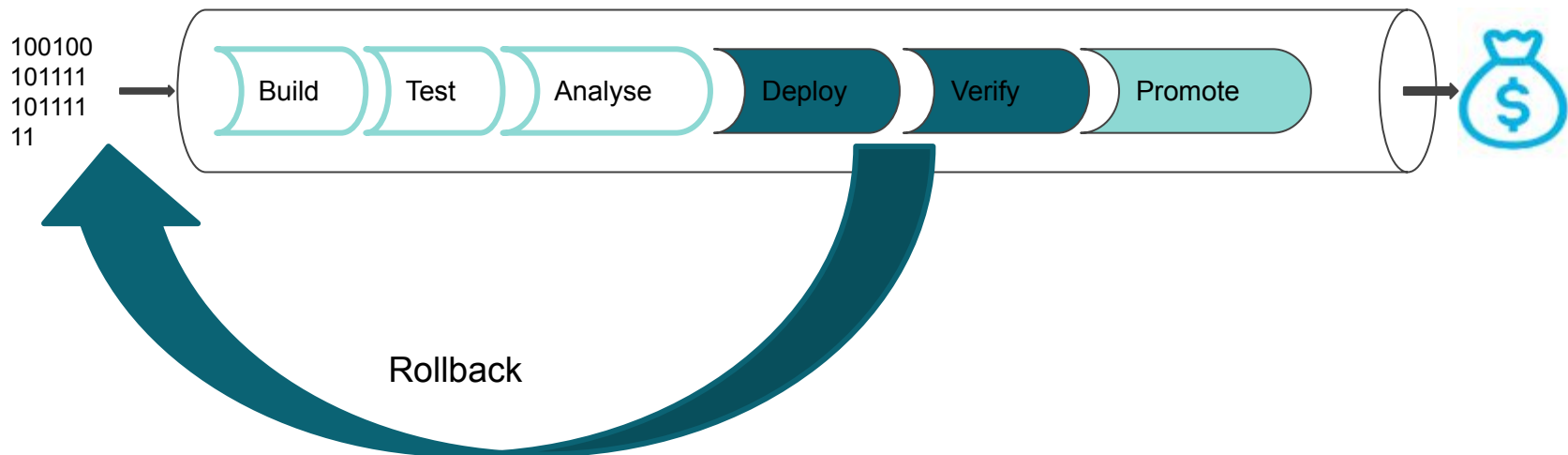
Common CD-related phases may include;

- Creating Infrastructure
- Provisioning Servers
- Copying files
- Promoting to production
- Smoke testing (Verifying, load testing, etc)
- Rollbacks



CI/CD pipeline

Combining CI and CD together can help us achieve a pipeline which transports high quality code from developers to clients.



Benefits of CI/CD to the Company

CI/CD feature	Value	Why it helps
Catch compile time errors	Reduce Cost	Less developer time on issues from new developer code
Catch Unit Test Failures	Avoid Cost	Less bugs in production and less time in testing
Detect Security Vulnerabilities	Avoid Cost	Prevent embarrassing or costly security holes
Automate Infrastructure Creation	Avoid Cost	Less human error, Faster deployments
Automate Infrastructure Cleanup	Reduce Cost	Less infrastructure costs from unused resources
Faster and More Frequent Production Deployments	Increase Revenue	New value-generating features released more quickly
Deploy to Production Without Manual Checks	Increase Revenue	Less time to market
Automated Smoke Tests	Protect Revenue	Reduced downtime from a deploy-related crash or major bug
Automated Rollback Triggered by Job Failure	Protect Revenue	Quick undo to return production to working state