



Fundamentals and Benefits of CI/CD

The best way to give your application
Auto-Deploy superpowers.

TOC

Continuous Integration.

Continuous Delivery.

Continuous Deployment.

CI/CD Benefits.

Introducing: CI/CD



Overview

There is some confusion in the community around Continuous Delivery, Continuous Integration and Continuous Deployment. For the sake of consistency throughout these slides, we will assume that Continuous Delivery is an overarching paradigm or mindset that informs and enhances the practices of Continuous Integration and Continuous Delivery.



Continuous Integration

The practice of merging all developers' working copies to a shared mainline several times a day.



Continuous Delivery



An engineering practice in which teams produce and release value in short cycles.



Continuous Deployment

A software engineering approach in which the value is delivered frequently through automated deployments.



CI/CD Benefits

Reduce Cost:

- Catch Compile Errors After Merge --> Less developer time on issues from new developer code.
- Automate Infrastructure Cleanup --> Less infrastructure costs from unused resources

Avoid Cost:

- Catch Unit Test Failures --> Less bugs in production and less time in testing
- Detect Security Vulnerabilities --> Prevent embarrassing or costly security holes
- Automate Infrastructure Creation --> Less human error, Faster deployments

Increase Revenue:

- Faster and More Frequent Production Deployments --> New value-generating features released more quickly
- Deploy to Production Without Manual Checks --> Less time to market

Protect Revenue:

- Automated Smoke Tests --> Reduced downtime from a deploy-related crash or major bug
- Automated Rollback Triggered by Job Failure --> Quick undo to return production to working state

Thank you!

