Fundamentals and Benefits of CI/CD

to Achieve, Build, and Deploy Automation for Cloud-Based Software Products

Overview:

Some of the core principles to discuss:

Continuous Integration (CI):

- Compile
- Unit Test
- Static Analysis
- Dependency vulnerability testing
- Store artifact

Continuous Integration (CD):

- Compile
- Unit Test
- Static Analysis
- Dependency vulnerability testing
- Store artifact

Continuous Delivery:

An engineering practice in which teams produce and release value in short cycles. This is achieved by incorporating the principles of CI/CD detail above.

Benefits

Wide range of benefits from these principles which ultimately all point towards increased efficiency and reduced costs.

Details of the benefits from both points of views (Tech vs Business) lists below:

Technical	Benefit	Business
Catch Compile Errors After Merge	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

Summary

The key benefits of CI/CD will enable the development teams to release code more frequently and with increased stability. This is achieved through CI/CD by:

1. Superior Code Quality
2. Reduced Changes & Review Time
3. Accelerated Release Cycles
4. Fault Detection & Isolation
5. Enhanced Test Reliability
6. Reduced Backlog
7. Improved Mean Time To Resolution (MTTR)
8. Cost Deduction
9. Enhanced Transparency & Accountability
11. Frequent Updates & Maintenance
12. Boosted Customer Satisfaction
13. Enhanced Performance Metrics

Reference

- Udacity - Cloud DevOps Engineer Nanodegree Program – Build CI/CD Pipelines, Monitoring & Logging