

Fundamentals and Benefits of CI/CD to Achieve, Build, and Deploy Automation for Cloud-Based Software Products.

Continuous Integration

*The practice of merging all developers' working copies to a shared mainline several times a day. It's the process of "**Making**". Everything related to the code fits here, and it all culminates in the ultimate goal of CI: a high quality, deployable artifact! Some common CI-related phases might include:*

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

Continuous Deployment

*A software engineering approach in which the value is delivered frequently through automated deployments. Everything related to deploying the artifact fits here. It's the process of "**Moving**" the artifact from the shelf to the spotlight. Some common CD-related phases might include:*

- Creating infrastructure
- Provisioning servers
- Copying files
- Promoting to production
- Smoke Testing (aka Verify)
- Rollbacks

Benefits of CI/CD

Technical Language	Value	Translation	
Catch Compile Errors After Merge	Reduce Cost	Less developer time on issues from new developer code	
Catch Unit Test	Avoid Cost	Less bugs in	

Failures		production and less time in testing	
Detect Security Vulnerabilities	Avoid Cost	Prevent embarrassing or costly security	
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	