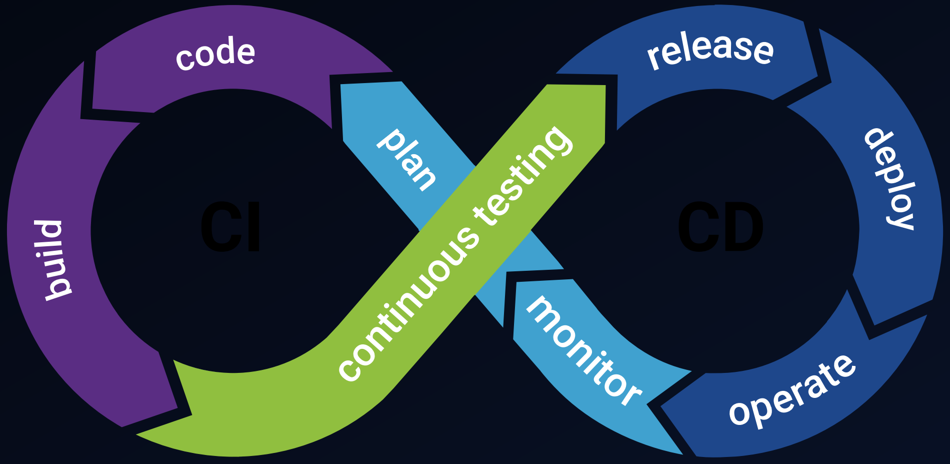
CI/CD

Effective way to build and deliver products.

Eng. : Eman Zaki

Definition

CI and CD stand for Continuous Integration and Continuous Delivery/Continuous Deployment. In very simple terms, CI is a modern software development practice in which incremental code changes are made frequently and reliably.

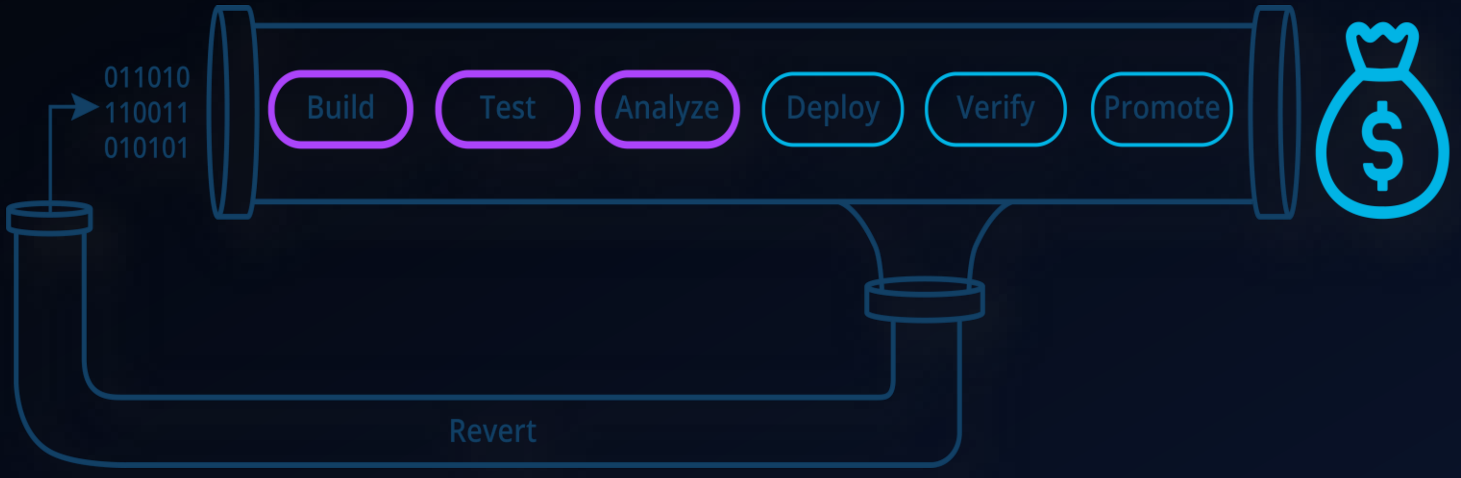


Automated build-and-test steps triggered by CI ensure that code changes being merged into the repository are reliable. The code is then delivered quickly and seamlessly as a part of the CD process. In the software world, the CI/CD pipeline refers to the automation that enables incremental code changes from developers’ desktops to be delivered quickly and reliably to production.

Continuous Integration

Everything related to the code fits here, Some common CI-related phases might include:

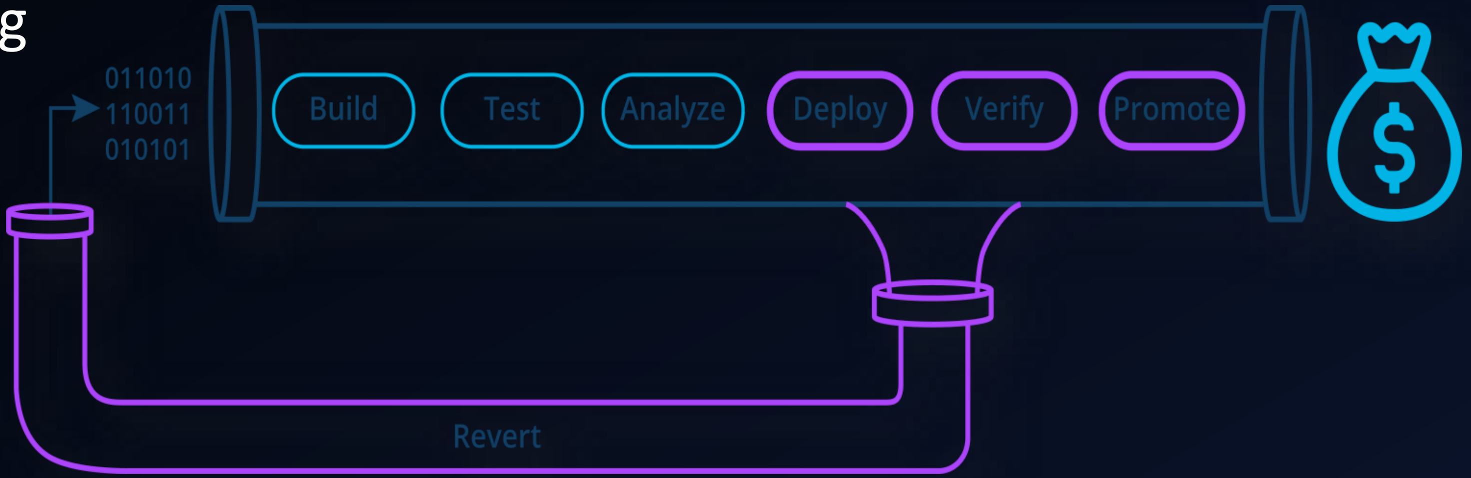
- Compile
- Unit Test
- Static Analysis
- Dependency vulnerability testing
- Store artifact



Continuous Deployment

Everything related to deploying the artifact fits here. 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

Using CI/CD enable us to :

- Reduce Cost :
  - Automate infrastructure cleanup so less infrastructure costs from unused resources
  - Catch compile errors after merge so less developer time on issues from new developer code
- Avoid Cost :
  - Less human error and faster deployments by automate infrastructure creation
  - Detect security vulnerabilities which prevent embarrassing or costly security holes
  - Catch Unit Test failures that leads to less bugs in production and less time in testing
- Increase Revenue :
  - Faster and more frequent production deployments to have new value-generating features released more quickly
  - Deploy to production without manual checks lead to less time to market
- Protect Revenue
  - Automated smoke tests help us to reduce downtime from a deploy-related crash or major bug
  - Automated rollback triggered by job failure so we quick undo to return production to working state