



A circular inset image shows a man with a beard and short hair, wearing a dark blue suit, white shirt, and dark tie. He is sitting at a table outdoors, looking down at a silver laptop. In the background, there's a blurred view of a city street with buildings and other people.

Know how to deploy in an artifact and start getting benefits in seconds?

SOLUTION IS CI/CD



A large stack of gold coins is shown on the right side of the slide. The coins are arranged in several vertical columns. A small orange plus sign is positioned above the topmost coin. To the right of the stack, there are three balloons: one pink, one yellow, and one light blue.

WOW!

Continuous Integration Key Points

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



CI Cost affection

- Catch Compile Errors After Merge

Less developer time on issues from new developer code=>Reduced Cost.

- Catch Unit Test Failures

Less bugs in production and less time in testing=> Avoided Cost

- Detect Security Vulnerabilities

Prevent embarrassing or costly security holes=> Avoided Cost

All about code and Dev workflow through phases



Compile



Unit Test



Static Analysis

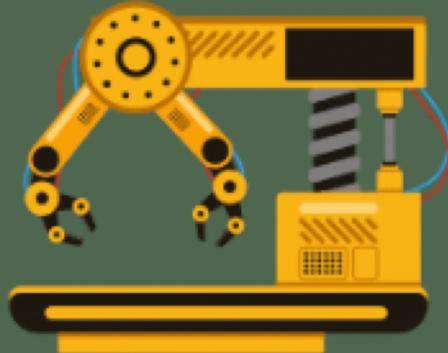


Store artifact



Dependency vulnerability testing

Continuous Deployment



CD Phases

Automated and frequent deployment

the process of "Moving" the artifact from the shelf to the spotlight automatically without human intervention

Ops Workflow

It is all about deployment to get a revenue.

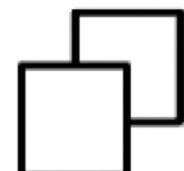
Creating infrastructure



Provisioning servers



Copying files



Promoting to production



Smoke Testing



Rollbacks



Benefits of CI/CD

BENEFIT	VALUE
Less time spent on issues from new developer as code passed by multiple verification steps	Reduced cost
Issues found previously before going to production	Avoid cost
No security breaches or vulnerabilities	Avoid cost
Faster deployments which avoids human errors	Avoid cost
Quick valuable feature	Increase revenue
Less time to market release	Increase revenue

Why should I start working with CI/CD?

Trustworthy

No dependency on one engineer

Speed Checking

No more time on merging issues after a new feature

High security

No more neglecting for testing functionalities

Testable code

More integration, evolving and contribution

Less Errors

Devops = Development + Ops deps.