



CI/CD?

Understanding
What is the
CI/CD, its
benefits and
selling it

TODAY'S PRESENTATION



1

What is the CI/CD ?

2

How the CI/CD works

3

CI Benefits

4

CD Benefits

5

Selling CI/CD

CI/CD

Before we start with the session, let's warm up a little with What's the CI/CD

CI/CD allows organizations to ship software quickly and efficiently.

CI/CD facilitates an effective process for getting products to market faster

What is CI/CD ?

CI/CD

CI/CD is a method to frequently deliver apps to customers by introducing automation into the stages of continuous delivery, and continuous deployment. CI/CD is a solution to the problems integrating new code can cause for development and operations teams (AKA "integration hell"). Specifically, CI/CD introduces ongoing automation and continuous monitoring throughout the lifecycle of apps, from integration and testing phases to delivery and deployment. Taken together, these connected practices are often referred to as a "CI/CD pipeline" and are supported by development and operations teams working together in an agile way with either a DevOps or site reliability engineering (SRE) approach.

ensures software quality and security and increases the profitability of code in production.

How CI/CD works?

**LET'S
BEGIN!**



CI/CD WORK

How CD works?

CD is a stage in CI/CD pipeline where the software is containerized and virtualized using tools such as Docker for containerization and Puppet or Ansible for configuration management. The fundamental difference between continuous delivery and continuous deployment is a manual quality assurance check stage that exists in the continuous delivery.

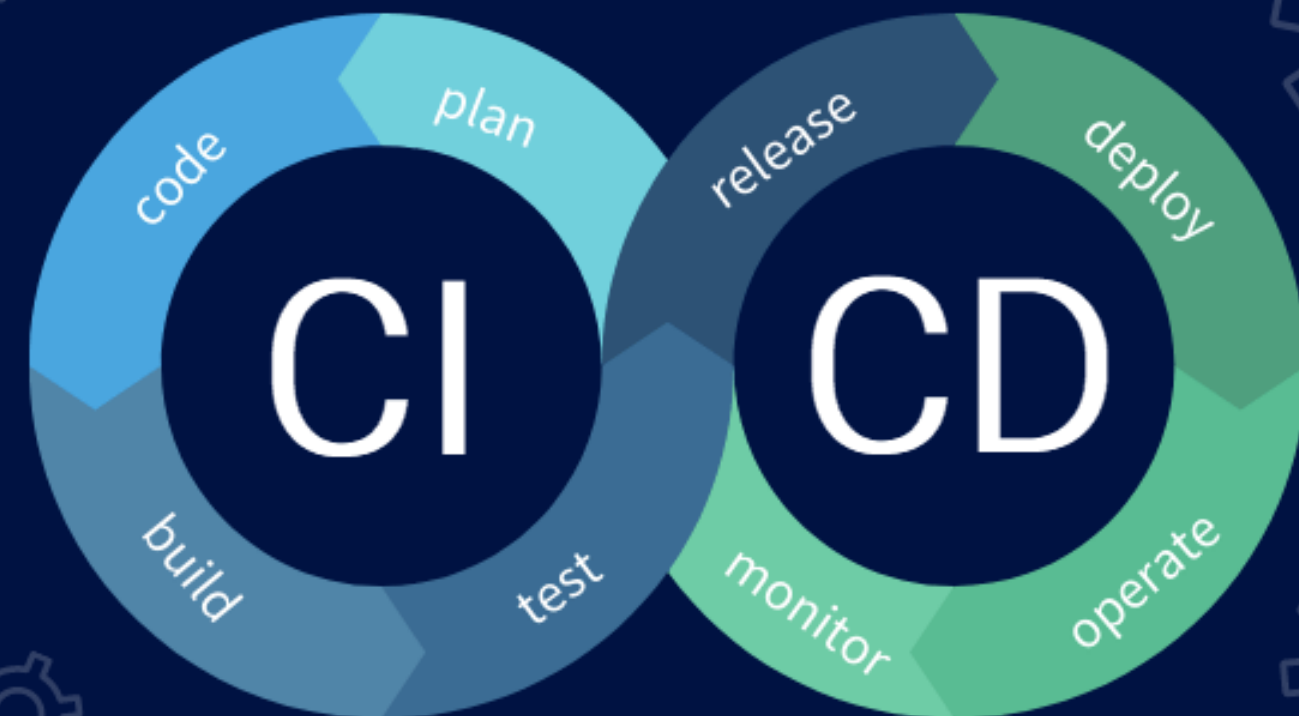
Containerization is the process of packing software with all its dependencies to ensure it runs on every system.

Configuration management is a part of continuous deployment where the server such as puppet or ansible connects with other tools such that it can run on any system.

How CI works?

CI is a process of developing, building, testing, and deploying software using automated tools. The process starts when a developer updates the source code, a CI tool pulls the code and pushes it to a build and test server.

Upon successful testing, the software is deployed to the manual check stage; once the operator approves the software it is then deployed to the production server where it is available to the customer.



CI - BENEFITS

1

Increase Revenue

2

Reduce costs

LEAD TIME:

- Early feedback and build/test automation help decrease the time it takes to go from code committed to code successfully running in production.
- Iterate faster: Smaller code changes allow software development teams to iterate faster and are easier to manage.

CATCH UNIT TEST FAILURE

- Will achieve reducing cost, which there will be less bugs in the production and less time in staging.
- Improve transparency: Continuous feedback through frequent testing helps developers see where bugs are.

DETECTING SECURITY VULNERABILITIES

- Automated pipelines enable preventatives embarrassing or costly security holes, which will reduce the costs or avoid them.
- Also catching compile errors after merging will reduce the cost which will be less developer time on issues from new developer code.

AUTOMATED INFRASTRUCTURE CREATION

- This will achieve reducing costs as well by less human errors and faster deployment.

CD - BENEFITS

1

Time Saving

2

Reduced Risk and Cost

AUTOMATE INFRASTRUCTURE CLEANUP

- By Cleanup, this will reduce the costs, by implementing less resources.

FAST PRODUCTION DEPLOYMENT & WITHOUT MANUAL CHECKS

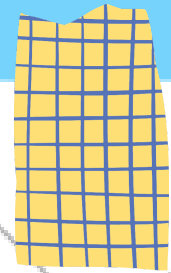
- This two will achieve increasing in revenue by generating new value features quickly in less time to market.

AUTOMATED SMOKE TESTS

- Smoke tests will reduce the down time from deploy-related crash or any bugs.

AUTOMATED ROLLBACK

- Automated rollbacks triggered by job failure will increase the revenue by undo all failure changes to make production back to the working state



SELLING CI/CD

Because CI/CD automates the process of integrating, releasing, and deploying software while removing traditional roadblocks. It supports the larger goal of agile methodology to accelerate the software development lifecycle, and it supports the DevOps approach of aligning development and operations teams.

1

In addition to supporting these larger goals, CI/CD enables you to:

Ship software quickly and efficiently

Increase productivity

Reduce risk on delivery

Incorporate user feedback faster

Standardize processes

Supports all rollbacks types

revenue and costs

Why CI/CD?

Value	Technical	Action
Reduce Cost	Automate Infrastructure Cleanup	Less Infrastructure costs from unused resources
Increase Revenue	Deploy to production without manual checks	Less time to Market
Avoid Cost	Catch Unit Test Failures	Less Bugs in production and less time on staging
Increase Revenue	Automated Smoke Tests	Reduced down time from deployed-related crash or bugs

66

**And Here we go,
In the end, I'd like to say
thank you for each one
who helped me in this
course**





**THANK
YOU!**

Name:
Mahmoud Qadry