CI/CD – ADOPTION – BENEFIT ANALYSIS

Developed for UdaPeople

WHAT IS CI/CD – SHORT DEFINITION

CI/CD is a method to frequently deliver apps to customers by introducing automation into the stages of app development. The main concepts attributed to CI/CD are continuous integration, continuous delivery, and continuous deployment.

KEY DEFINITIONS

Continuous Integration : helps developers merge their code changes back to a shared branch, or "trunk," more frequently—sometimes even daily. Once a developer's changes to an application are merged, those changes are validated by automatically building the application and running different levels of automated testing, typically unit and integration tests, to ensure the changes haven't broken the app.

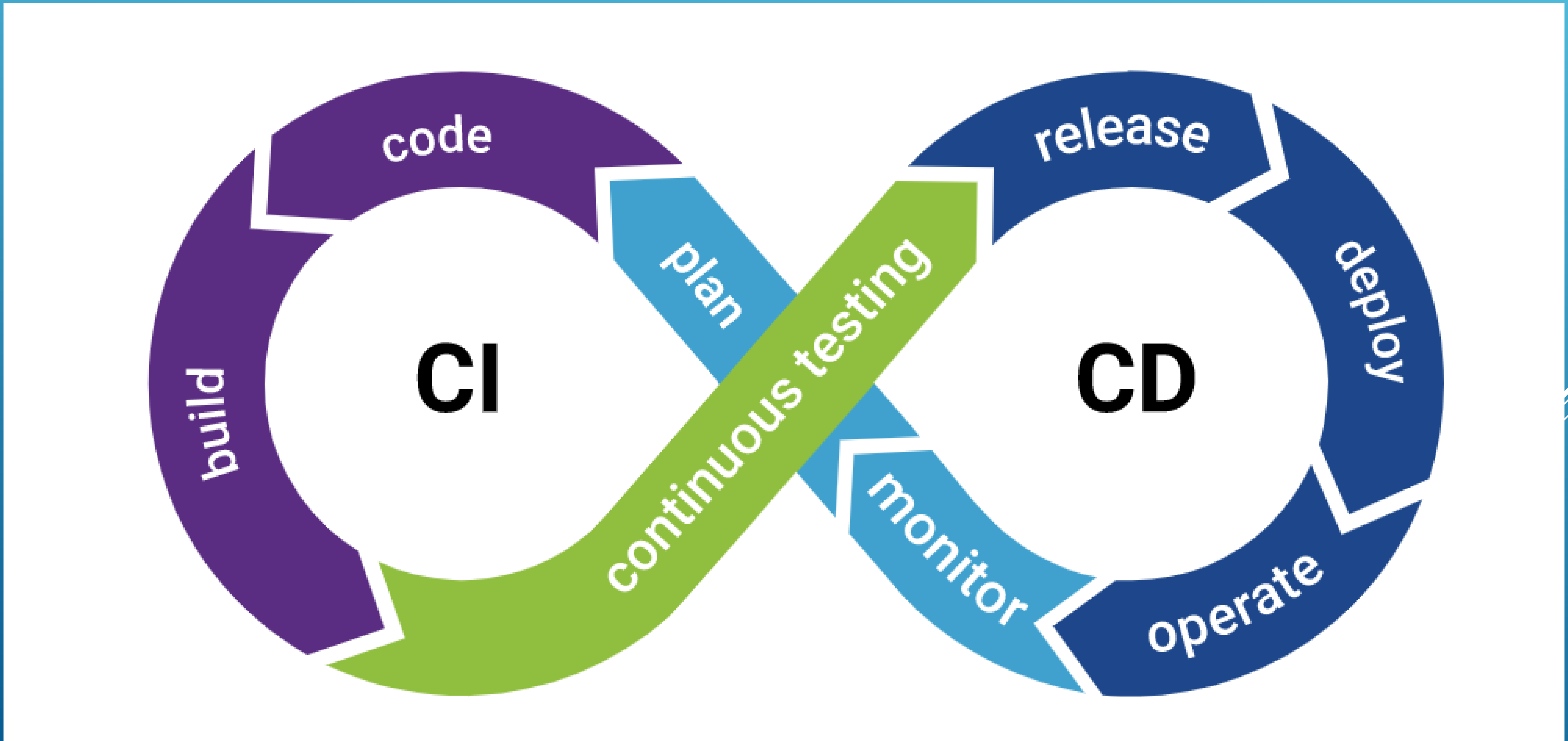
Continuous Delivery : Following the automation of builds and unit and integration testing in CI, continuous delivery automates the release of that validated code to a repository. So, in order to have an effective continuous delivery process, it's important that CI is already built into your development pipeline.

Continuous Deployment: The final stage of a mature CI/CD pipeline is continuous deployment. As an extension of continuous delivery, which automates the release of a production-ready build to a code repository, continuous deployment automates releasing an app to production.

WHY CI/CD ? CURRENT PAIN POINTS

1. Manual Build and Release Process : The current process is very manual where multiple team members are working on various features and they tend to step on each other thus delaying the key milestones. The UT and CIT Phase thus gets delays and sometimes skipped partially to meet the timelines.
2. Deployments are very complex and require significant planning. Sometimes, incorrect features are prioritized that basically forces a complete overall rework.
3. Poor Software quality identified at all key phases.
4. The business is not very happy as they see delays and incorrect configurations leading to pre-prod phase wich delays User Acceptance Testing.

SOLUTION


code
build
plan
continuous testing
CI
release
deploy
CD
monitor
operate

BENEFIT STATEMENT

1. Reduced Cost due to less rework
2. Reduced burden on developers so that they can pick new features more quickly and can plan things more efficiently.
3. Increased Customer Satisfaction as they can see features getting implemented quickly thus providing them the key value.
4. Reliable – The entire process is reliable and can be used for each of the feature statement and would not change.