POTENTIAL BENEFIT OF ADOPTING CICD APPROACH FOR THE UDAPEOPLE PROJECT

Being the Text of a Presentation Prepared
by

Tewogbade Ayobami

For Udacity Devops Nanodegree Project 3 submission

August 21, 2022

What is CICD

CI and CD are two acronyms frequently used in modern development practices and DevOps.
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.
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.

CICD tools

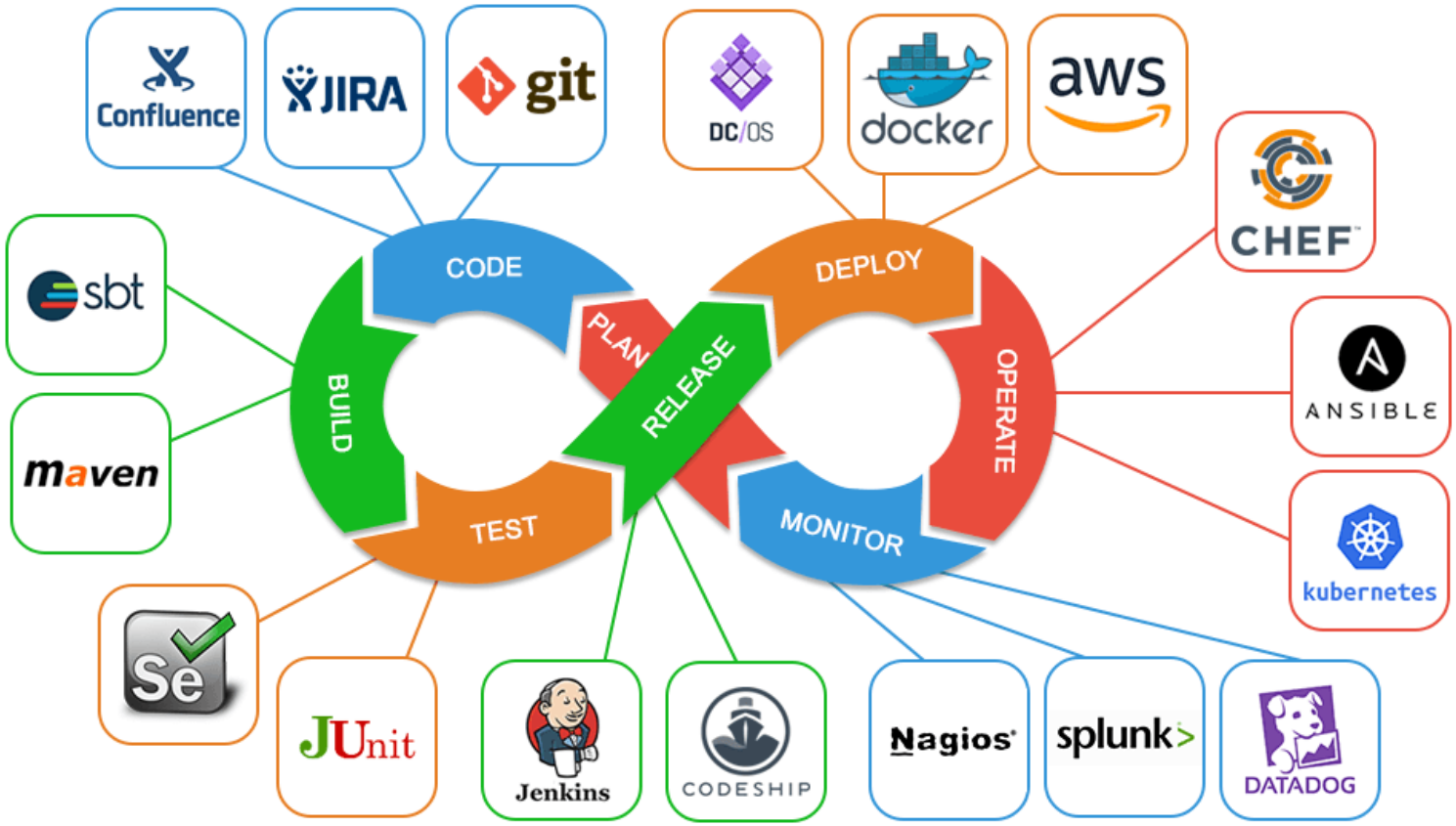

Figure:

Continuous integration

What you need (cost)

- Your team will need to write automated tests for each new feature, improvement, or bug fix.
- You need a continuous integration server that can monitor the main repository and run the tests automatically for every new commit pushed.
- Developers need to merge their changes as often as possible, at least once a day.

Continuous integration

What you gain

- Fewer bugs get shipped to production as regressions are captured early by the automated tests.
- Building the release is easy as all integration issues have been solved early.
- Less context switching as developers are alerted as soon as they break the build and can work on fixing it before they move to another task.
- Testing costs are reduced drastically – your CI server can run hundreds of tests in a matter of seconds.
- Your QA team spends less time testing and can focus on significant improvements to the quality culture.

Continuous delivery

What you need (cost)

- You need a strong foundation in continuous integration and your test suite needs to cover enough of your codebase.
- Deployments need to be automated. The trigger is still manual but once a deployment is started there shouldn't be a need for human intervention.
- Your team will most likely need to embrace feature flags so that incomplete features do not affect customers in production.

Continuous delivery

What you gain

- The complexity of deploying software has been taken away. Your team doesn't have to spend days preparing for a release anymore.
- You can release more often, thus accelerating the feedback loop with your customers.
- There is much less pressure on decisions for small changes, hence encouraging iterating faster.

Continuous deployment

What you need (cost)

- Your testing culture needs to be at its best. The quality of your test suite will determine the quality of your releases.
- Your documentation process will need to keep up with the pace of deployments.
- Feature flags become an inherent part of the process of releasing significant changes to make sure you can coordinate with other departments (support, marketing, PR...).

Continuous deployment

What you gain

- You can develop faster as there's no need to pause development for releases. Deployments pipelines are triggered automatically for every change.
- Releases are less risky and easier to fix in case of a problem as you deploy small batches of changes.
- Customers see a continuous stream of improvements, and quality increases every day, instead of every month, quarter or year.

Thank you for listening!

References

Atlassian. (n.d.). Continuous Integration vs. delivery vs. deployment. Atlassian. Retrieved August 21, 2022, from https://www.atlassian.com/continuous-delivery/principles/continuous-integration-vs-delivery-vs-deployment: :text=CI%20stands%20for%20continuous%20integration,continu

Red Hat. (n.d.). What is Ci/CD? Red Hat - We make open source technologies for the enterprise. Retrieved August 21, 2022, from https://www.redhat.com/en/topics/devops/what-is-ci-cd