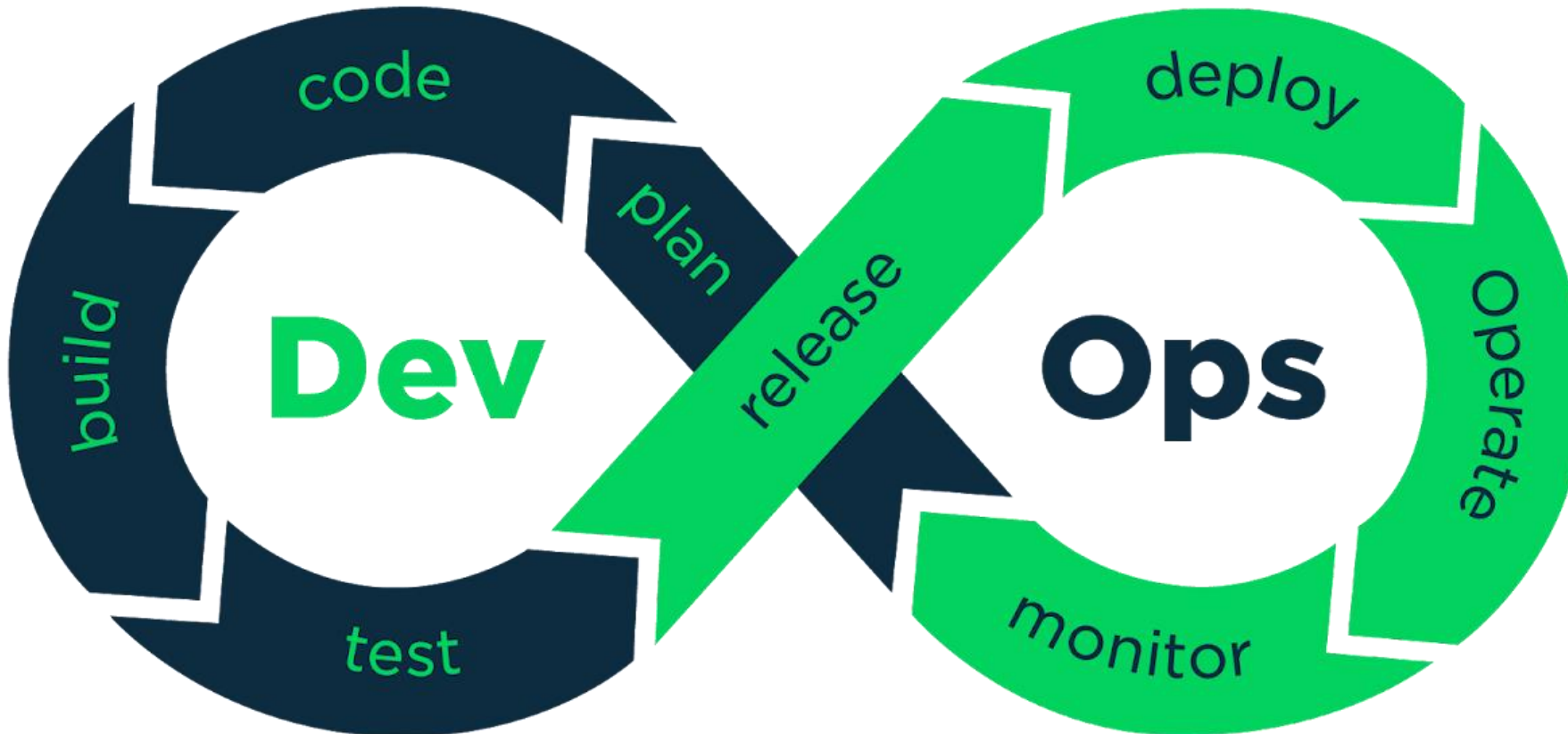


# CI/CD

Towards a time of less effort and  
greater savings



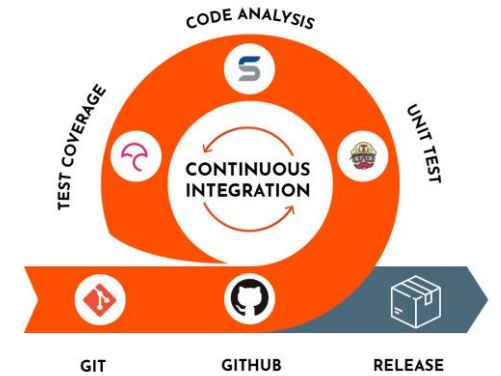
# Continuous Integration:

## What is CI?

- The daily practise of merging all developers' working copies to a shared mainline.
- It refers to a fundamental DevOps best practise in which developers frequently merge code changes into a central repository where automated builds and tests are run.
- It places a high value on testing automation to ensure that the application does not break when new commits are integrated into the main branch.

## Benefits

- Each new feature, improvement, or bug fix will necessitate the creation of automated tests by your team.
- Developers must merge their changes as frequently as possible, at least once per day.



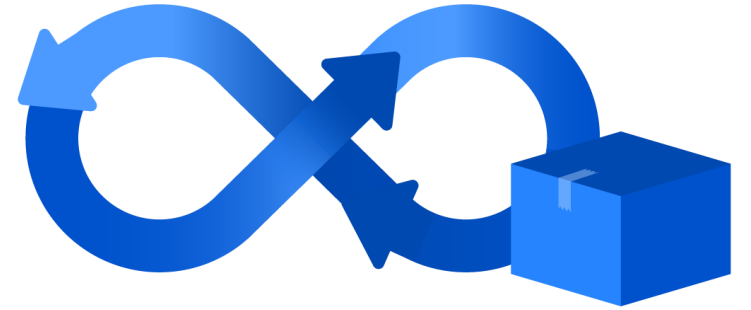
# Continues Deployment:

## What is CD?

Continuous deployment is an excellent way to shorten the feedback loop with your customers and relieve pressure on the team because there is no longer a "release day." Developers can concentrate on creating software while watching their work go live minutes after they finish.

## Benefits

You can develop more quickly because there is no need to pause development for releases. For each change, deployment pipelines are automatically triggered. Customers notice a steady stream of improvements, and quality improves every day, rather than every month, quarter, or year.



# Why CI/CD ?

- 1. Low cost** : The CI/CD pipeline approaches software delivery differently. It is comparable to the delivery pipeline of an assembling unit. Time and assets are critical in any business situation. With such requirements, firms are expected to respond to client demands quickly and effectively.
- 2. Higher efficiency** : One of the primary benefits of a CI/CD pipeline is increased productivity. If your review process includes deploying code to development, testing, and production environments as well as entering multiple commands across multiple domains, you should automate it. This necessitates the use of a CI/CD framework.

**3. Faster Delivery** : If a new security feature is required, your team can use CI/CD and automated testing to bring the fix to production systems more quickly and with greater assurance. What used to take weeks or months to complete can now be done in a matter of days or even hours.

**4. Better Planning** : Organizational designs must be flexible enough to adapt to changing economic conditions. However, adapting to rapid changes in dynamic business conditions is difficult for development and testing teams. A CI/CD pipeline enables organisations to do so by ensuring a well-organized surplus of items and a continuous line of communication with customers.

**5. Reduced risk of defects** : A CI/CD pipeline allows you to test and deploy code more frequently, giving QA engineers the ability to identify and fix errors as soon as they occur. You can effectively mitigate risks in real time this way.