CICD

CONTINUOUS INTEGRATION CONTINUOUS DEPLOYMENT

OVERVIEW

Continuous Integration

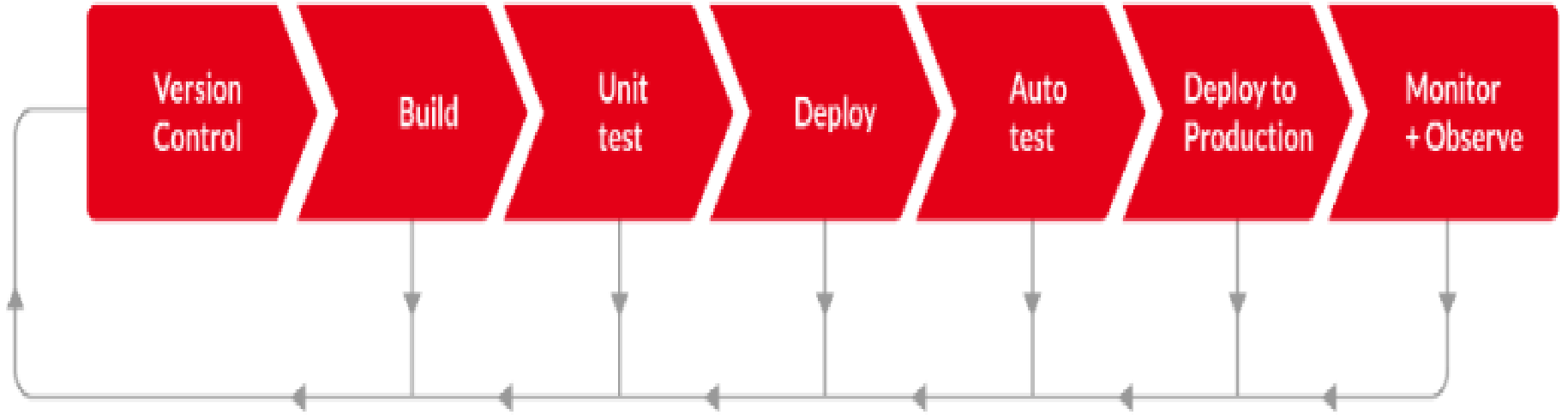
Consider an application that has its code stored in a Git repository in GitLab. Developers push code changes every day, multiple times a day. For every push to the repository, you can create a set of scripts to build and test your application automatically. These scripts help decrease the chances that you introduce errors in your application.

Continuous Delivery

It is a step beyond Continuous Integration. Not only is your application built and tested each time a code change is pushed to the codebase, the application is also deployed continuously. However, with continuous delivery, you trigger the deployments manually. Continuous Delivery checks the code automatically, but it requires human intervention to manually and strategically trigger the deployment of the changes.

Continuous Deployment

It is another step beyond Continuous Integration, similar to Continuous Delivery. The difference is that instead of deploying your application manually, you set it to be deployed automatically. Human intervention is not required.


Version Control
Build
Unit test
Deploy
Auto test
Deploy to Production
Monitor + Observe

WHAT ARE THE BENEFITS OF CI/CD?

- Avoid Cost
- Reduce Cost
- Increase Revenue
- Protect Revenue

AVOID COST

- Less bugs in production and less time in testing
- Prevent embarrassing or costly security holes
- Less human error, Faster deployments

REDUCE COST

- Less developer time on issues from new developer code

INCREASE REVENUE

- New value-generating features released more quickly
- Less time to market

PROTECT REVENUE

- Reduced downtime from a deploy-related crash or major bug
- Quick undo to return production to working state

THANK YOU!