CI/CD

Section 3

What you will learn in this section

Theory:

- What Continuous Integration (CI) and Continuous Delivery (CD) are
- Compare and contrast Continuous Integration, Continuous Delivery, and Continuous Deployment

Practice:

- Implement a CI workflow with GitHub, Jenkins, and Slack

What is CI/CD

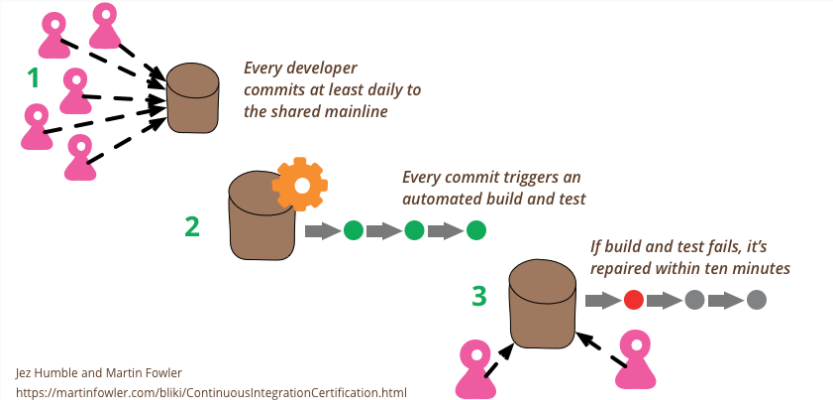
Video 3.1

What you will learn in this video

- What Continuous Integration (CI) and Continuous Delivery (CD) are
- Compare and contrast Continuous Integration, Continuous Delivery, and Continuous Deployment
- Benefits of CI and CD

Continuous Integration

- **Continuous Integration (CI)** is a software development practice whereby members of a team integrate, build, and test their code frequently in a development environment



CI Workflow

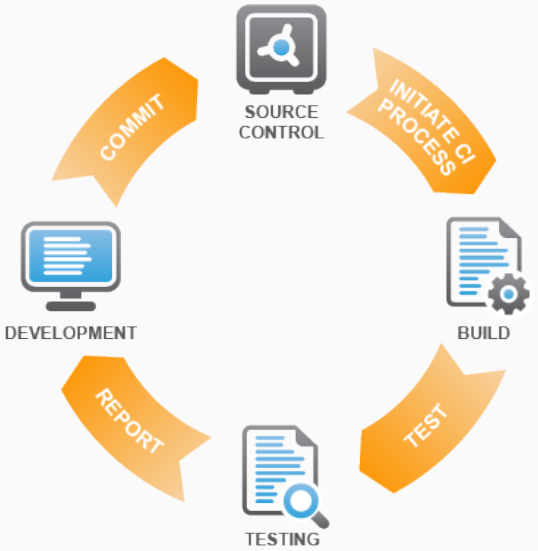


Image: http://www.techdesignforums.com/practice/technique/continuous-integration-jenkins-verification-open-source/

Benefits of CI

- **Code quality**: bugs are spotted immediately and can be fixed quickly, regression bugs are avoided
- **Integration**: reduced risks, time, and costs for integration
- **Automation**: no manual steps, faster and less error prone build process, no more 'but it worked on my machine' issues

Continuous Delivery and Continuous Deployment

- **Continuous Delivery (CD)** is a software development practice whereby a team builds software in such a way that it can be released to production at any time

- **Continuous Deployment** means that every change automatically gets pushed to production, resulting in many production deployments every day

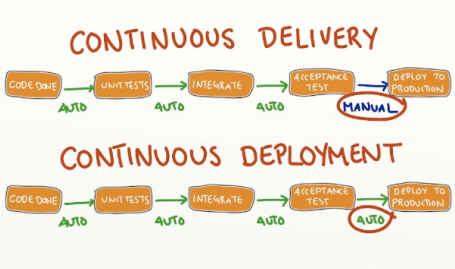
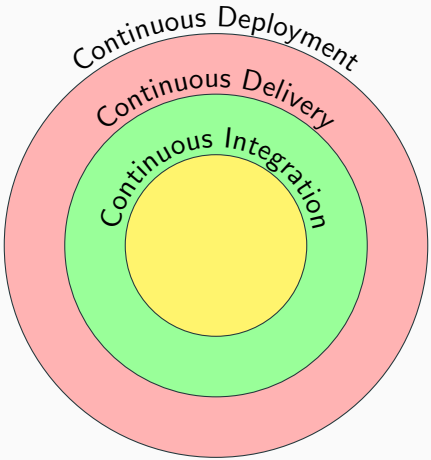


Image: http://blog.crisp.se/2013/02/05/yassalsundman/continuous-delivery-vs-continuous-deployment

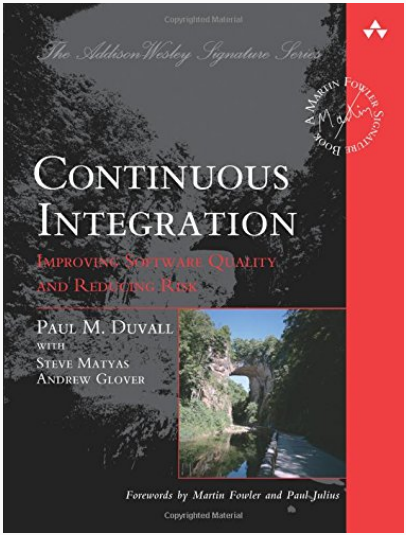
Benefits of CD

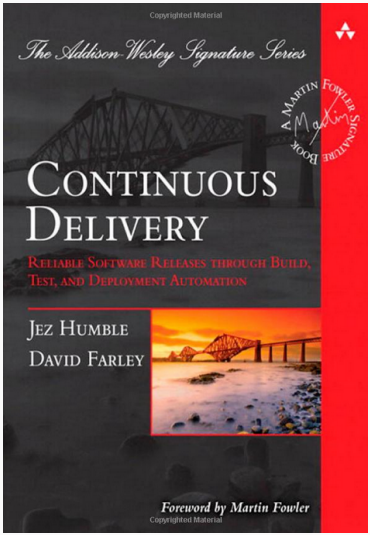
- Faster reaction to market changes, **quicker** to adapt to changing user and business needs
- **Faster** software life cycle
- The product is always in a **shippable** state
- No risky big bang releases, **smaller and more frequent** releases, reduced risk, time, and cost to deploy to production

How Do They Compare?



Book References





Summary

- **Continuous Integration (CI)** is a software development practice whereby members of a team integrate, build, and test their code frequently in a development environment

- **Continuous Delivery (CD)** is a software development practice whereby a team builds software in such a way that it can be released to production at any time

- **Continuous Deployment** means that every change automatically gets pushed to production, resulting in many production deployments every day

Next Video

Video 3.2 CI Workflow