Continuous Integration & Continuous Development

Accelerating development work and code submission

Scott Hanselman

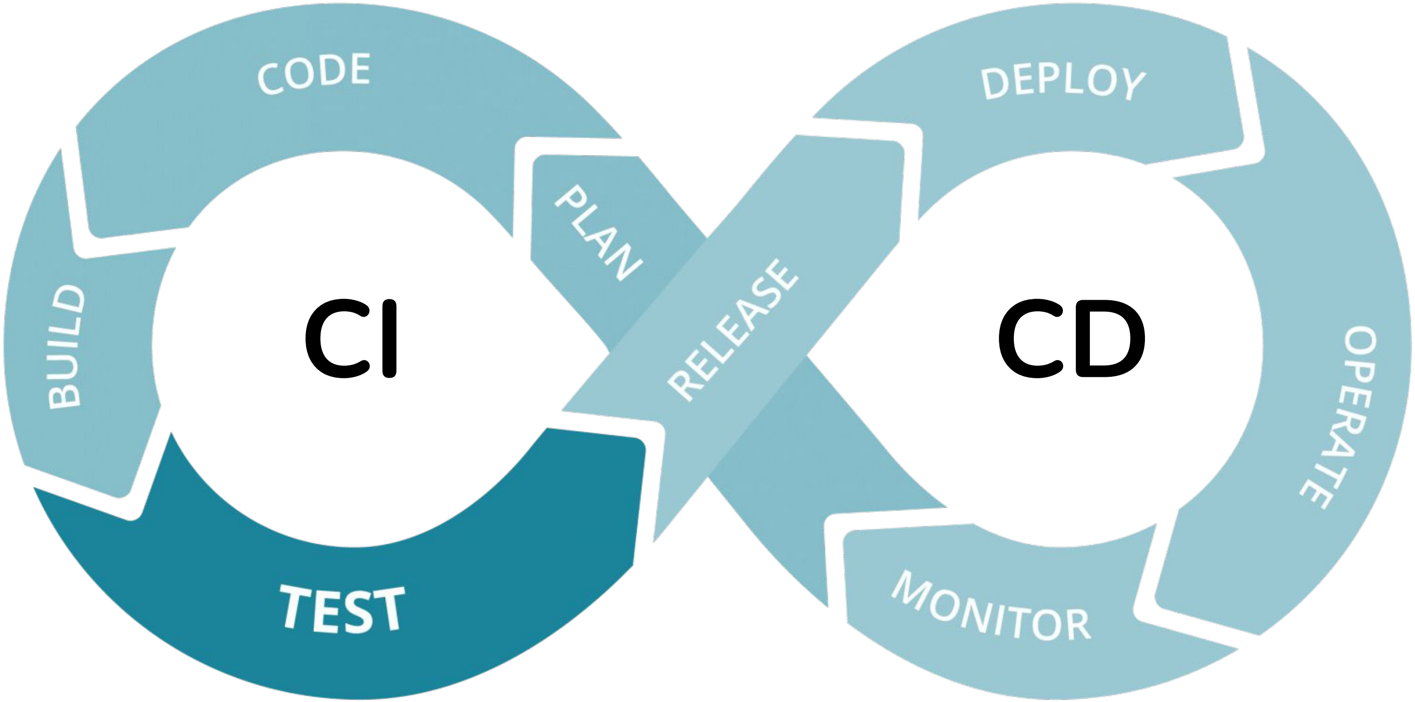
Professor at Oregon Institute of Technology

The most powerful tool we have as agile developers today is automation supporting us shipping reliable code in minutes…

not hours or days.

"CI / CD is a software development principle where you build software in such a way that the software can be released to production at any time."

CI CD lifecycle



Gians

Continuous Integration

- Less bugs get shipped to production as regressions are captured early by the automated tests.
- Testing costs are reduced drastically – your CI server can run hundreds of tests in the matter of seconds.
- The QA team spend less time testing and can focus on significant improvements to the quality culture.

Continuous Development

- 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 problem as you deploy small batches of changes.
- Customers see a continuous stream of improvements, and quality increases.

Engineering improvements

BETTER CODE

- Smaller code changes
- Fault isolations

FASTER & LESS FIXES

- Faster MTTR to find and fix broken features
- Enhanced test reliability

LESS WORKLOAD

- Reduced backlog
- Overall cost reduction

BETTER STRUCTURE

- Enhanced transparency, accountability and ownership
- Streamlined communication

Requirements and costs

Continuous Integration

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

Continuous Development

- Testing culture needs to be at its best. Quality of the test suite will determine the quality of releases.
- 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.

Building code confidence

Implementing CI / CD is a crucial step in increasing the confidence level of developers and business people in newly released software.

Stakeholders need confidence in:

- Features are being implemented correctly
- Production is an reliable environment for users
- Issues can be detected and resolved quickly

With CI / CD the
engineering team and
business team becomes…

…ONE.