

CI vs CD

Adopting modern development practices like [continuous integration \(CI\)](#), [continuous delivery \(CD\)](#), and [continuous deployment](#) can help teams meet these demands and ship software more frequently.

But what's the difference between these three approaches?



Continuous integration is the practice of merging developer working copies to shared repositories multiple times per day.

With CI, developers frequently commit their code changes to a shared version control repository.

Each commit triggers an automated build and test process to catch integration errors as early as possible.

CI helps teams avoid "integration hell" that can happen when developers work in isolation for too long before merging their changes.



Continuous delivery takes CI a step further with automated releases.

CD means that at any point, you can push a button to release the latest app version to users.

The CD pipeline deploys each code change to a testing/staging environment and runs automated tests to confirm the app is production ready.

This ensures developers always have a releasable artifact that has passed tests.

While CD enables releasing often, someone still needs to manually push the button to promote changes to production.



Continuous deployment fully automates the release process.

Every code commit that passes the automated tests triggers an immediate production deployment.

This enables teams to ship features as fast as developers write code.

However, the business may not want to release daily since this could overwhelm users with constant changes.

Many teams use feature flags so developers can deploy new features, but limit their exposure until the business is ready for the public launch.

Adopting CI, CD, and CD practices can accelerate a team's ability to safely deliver innovation.

The key is automating repetitive processes to limit manual errors, provide rapid feedback, and reduce risk.

This frees up developers to focus their energy on writing great code rather than building and deploying it.

The outcome is faster time-to-market and more frequent delivery of customer value.

Revision #5

Created 13 January 2024 21:26:21 by sedawk

Updated 8 February 2025 22:21:39 by sedawk