

# CI/CD Technology

HOW IT CAN CHANGE THE TECHNOLOGY  
AND SAVE COST AND INCREASE PROFIT

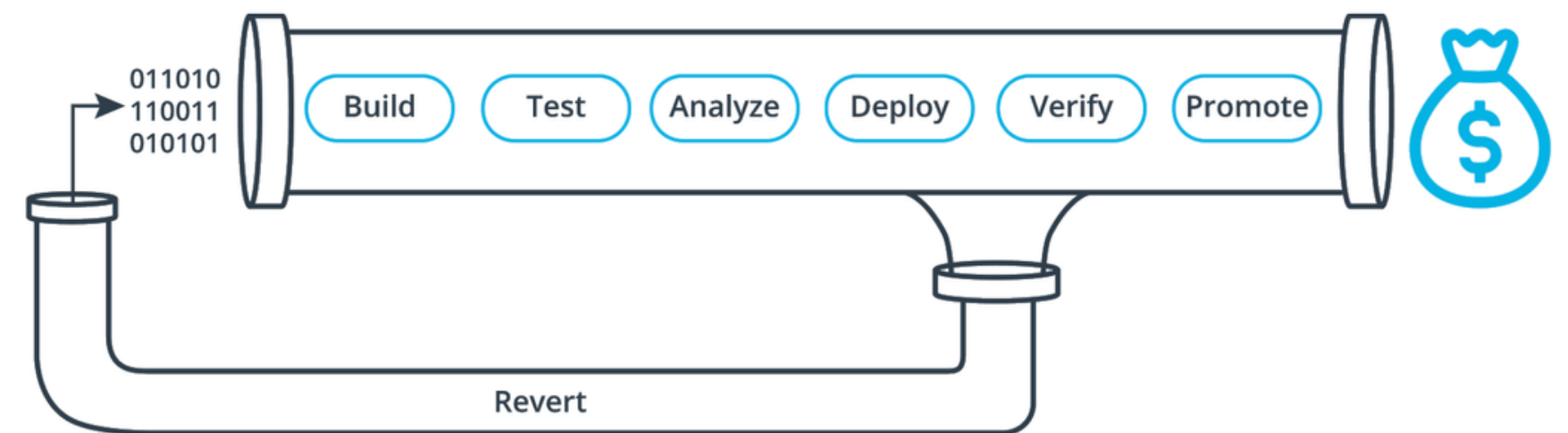




# What is CI/CD?

CI/CD is a method to frequently deliver apps to customers by introducing automation into the stages of app development. The main concepts attributed to CI/CD are continuous integration, continuous delivery, and continuous deployment.

## The CI/CD Pipeline



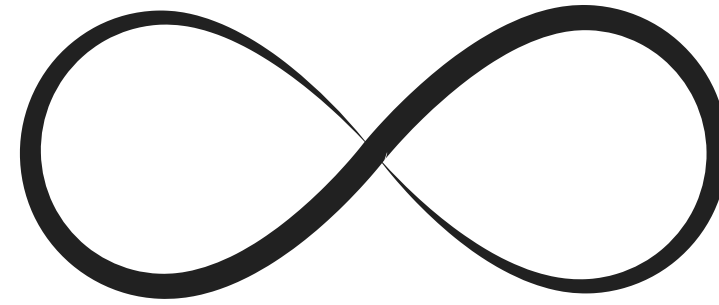
# How CI/CD Automation Saves Money and Time?

Today developers can make dozens/hundreds and in some cases thousands of changes to their main software daily, so it's even hard to imagine how much time the process of testing and delivering those changes would take, if done manually.

Taking plenty of toil out of SDLC is the key point of DevOps, which is all about delivering high-quality software and features as fast as possible and gaining swift feedback from customers. DevOps promotes CI/CD automation, as with CI/CD properly set-up and programmed, product delivery speed evolves notably. In fact, Puppet experts in their recent report identified that organizations utilizing CI/CD practices deploy 46x more frequently and spend up to 44% more time on creative work.

From IT experts' point of view the significance of DevOps methodology, in general, and CI/CD automation, in particular, is quite clear. However, for some leaders, it might be hard to follow how the whole investment in CI/CD tools (pretty often very sizable) can be beneficial in the end. If this is your case, let's discuss how CI/CD automation can lower costs for your organization. Fast/frequently released software and features save not only time but also cut down on:

- Deeply expensive downtime. Long-term downtime does not only influence your revenue negatively but also undermines your company's credibility.
- Troubleshooting costs. With CI/CD pipeline properly automated, the process of the product deployment and release is more reliable and developers are not wasting hours on bug-fixing.
- Switching/onboarding costs. Automated CI/CD let new members of the team get down to business right away without having to deal with an exhausting learning curve.
- Staff turnover costs. Happy employees — more successful company! When developers are freed from manual, repetitive tasks, they focus on creating new features and code and continuous quality improvement, isn't that right? Sure, it is. With them knowing they can rely on their system, employees become more relaxed and satisfied and, as a result, they stay with you, work harder and become more and more in love with the company.



## How it can increase revenue?

Continuous integration and continuous deployment are both powerful and profitable tactics in and of themselves. But if you've been reading this with an "either/or" to the subject, you're limiting your effectiveness right out of the gate.

If you want to get everything that CI and CD have to offer, they need to be adopted in tandem with one another. A strong regimen of CI will solve many of your back-end development problems before they have a chance to get serious. This also prevents many layers of complexity from forming, and making it difficult to trace problems back to their source when they occur. If all that's not enough, a rigorous commitment to CI will strengthen your team's intuitive ability to work with one another.

Continuous deployment, as we've noted, specializes in solving the front-end feedback around your product. How do your users feel about it? Are there key components you're missing or central user goals you're failing to meet? These issues, though not strictly technical, are every bit as important as the actual coding.

Just as a dance is being conducted between coding team members in CI, a constant dance is going on between your production team and your users. Users are constantly giving feedback to your team through your frequent "low-risk releases", and that instant feedback is swiftly used to construct a better product.

Do you see the continual, self-perpetuating loop that's created when you enforce CI and CD together? Your development process runs more smoothly and successfully, as does each recurring release cycle, with the benefit of instant feedback from the last release.

## Business Benefits of CI/CD



Efficient  
Software Delivery



Improved  
Team Productivity



Identify and  
Fix Bugs Quicker

Reduced  
Costs



Improved  
MTTR



Greater  
Collaboration



Deliver Updates Faster

