

TOP 10 TIPS TO IMPLEMENTING CONTINUOUS DELIVERY



Transitioning to continuous delivery is a challenge full of failure leading to success

Shifting an existing team to Continuous Delivery (CD) is hard. And make no mistake, you will fail along the way. The results, however, can be unmistakably awesome.

Companies that make the transition to Continuous Delivery see measurable benefits, especially when they can demonstrate that failures are, in fact, opportunities for learning. Those benefits can show themselves at different times, seeing as each organization is unique.

As you embark on the shift towards Continuous Delivery, you can expect to measure significant changes, including:

- Improved business agility
- Increased speed in deployment frequency
- Reduced lead time for changes
- Faster experiments
- Lower change failure rates

Equally powerful, but harder to measure, are the benefits to teams who will personally develop new people and tech skills. Each member will be stretched in ways they didn't expect and everyone (including you) will learn a hell of a lot along the way.

Know this: discomfort means it's working

Balancing the speed of change with experimentation will no doubt cause discomfort. Especially for stakeholders with SLA's to maintain around uptime. Stability and availability often brings some tough conversations, and tougher stakeholder management hurdles. Continual testing, measurement, and solid communication are the foundational elements for gaining adoption.

But the real secret to success? Keeping everyone aware of *progress* is ultimately the best way to gain momentum with a Continuous Delivery program.

Steps to make the shift to Continuous Delivery (and measure success along the way)

1. The very first thing to do is set the vision behind Continuous Delivery (not the end-state) and get started. Share across the organization why and what Continuous Delivery means to the business.
2. Treat CD as a new agile product. Identify your customers, assign an owner, establish a backlog and use an agile framework of lean method for delivery. Plan increments, experiments, get feedback at demos, release often, hold retrospectives. ***Let your customers tell you what's valuable.*** Learn.
3. Combine value stream mapping (VSM) with a visualisation of the potential CD pipeline (from source code to Prod) to make the process as lean as possible. Whiteboard it out to get your major principles, components and pipeline mapped out as a starting point.
4. People, processes and tools are all up for grabs when it comes to successful CD. Don't depend on the tools alone! This is a cultural shift and it must be observed across the entire company.

5. Never prioritize throughput over stability, they go hand in hand. Think [DevOps](#).
6. Automate everything and shift it left in the CD pipeline that you produced. To go fast you must also be safe and automation will help set the team free of "busywork", so they can **focus on solving complex problems and innovating**.
7. Embrace the cloud and open-source tools; use the wider external community to sound out ideas; and get to Meetup groups to discuss in person. The DevOps community is a powerful resource invested in each other's success.
8. Include non-functional requirements (NFR's) early; reliability, testability, availability, resiliency - they need to be in your Product Owners backlog and given sufficient priority alongside new features.
9. Measure the hell out of it so that you can improve; know where you start, where you want to get to, and publish your actuals along your journey.
10. Don't try this alone, use DevOps and CD consultancies or hire experts for tech spikes. Your staff will also need support, training and mentorship so invest in your people. Have a plan to get the help to put good practices in place and then allow your team to make it their own.

Successful transitions have one thing In common: customer focus

As I've watched, coached, and collaborated with companies to successfully deliver products and transition to Continuous Delivery, the most common thing I've seen among successful groups is they keep a *laser focus on the customer*. In addition, they commit to streamlining everything from inception to delivery and beyond.

Focusing on the customer, and committing to continual improvement means a faster and more successful delivery pipeline overall.

Want to hear from more top DevOps leaders and practitioners on how to flip the switch to Continuous Delivery?