

DevOps Myths

1. Adopting tools makes you DevOps

Some who have recently caught wind of the DevOps movement believe they can instantly achieve this nirvana of software delivery simply by following a checklist of tools to implement within their team. Their assumption is that if they purchase and implement a configuration management tool like Chef, a monitoring service like Librato, or an incident management platform like VictorOps, they've achieved DevOps!

[DevOps requires a cultural shift](#) beyond simply implementing a new lineup of tools. Each department, technical or not, needs to understand the cultural shift behind DevOps. It's one that emphasizes empathy and better collaboration. It's more about people.

2. DevOps and continuous delivery are the same thing

While it's true that continuous delivery of software does indicate that an organization has established key components of DevOps, it isn't a binary relationship. They aren't immediately tied to one another, and they certainly aren't the same thing.

Improving the culture of the teams building and maintaining the software and infrastructure, along with the supporting groups like sales and marketing, should be the focus. Everyone must truly begin to empathize with one another and care about what takes place outside of their own little world.

3. DevOps means going to conferences

I go to a ton of conferences, and because of this I've been exposed to a lot of thoughts and ideas, rapidly and consistently. This has helped me wrap my head around a variety of things more quickly than I otherwise would have, but the fact that I have been to dozens of events doesn't mean I can then implement DevOps.

One person isn't the keystone in DevOps. It's a cultural shift in an organization that requires support, understanding, and empathy from everyone.

Attending conferences to learn new concepts, emerging technologies, and innovative tools is important not only in [software development](#) but also every industry. Much can be gained simply by checking out one or two conferences or workshops each year. Likewise, going to local meetups in your own area, where others are getting together regularly to discuss DevOps, has proven to be a great way to begin absorbing key concepts. But beginning the transformation in your organization still requires your effort.

4. Having DevOps in your title means you're doing DevOps

I'm a DevOps evangelist. However, despite what recruiters think when they see my LinkedIn profile, I can't immediately provide DevOps to an organization. Likewise, hiring a DevOps engineer, manager, or consultant doesn't mean you have acquired DevOps. That's not how it works.

Hopefully, those who do have DevOps in their title are taking all the necessary steps to begin transforming the culture to build empathy throughout the organization. This empathy will drive a deeper understanding of how things get done, and how everyone is part of not only the delivery of software but also the maintenance of code and the infrastructure in which it resides.

5. Continuous delivery is only for web companies

Although [continuous delivery](#) and the processes that help teams achieve it were the creation of modern operations teams within software-as-a-service (SaaS) companies (i.e., web companies), it's a concept that applies to all modern software delivery. You can make the argument that it's imperative to the success and livelihood of SaaS companies to provide the latest versions of their software quickly and reliably, but it's equally important to non-web software providers.

Being able to quickly make new versions available—especially those that include enhancements and bug fixes—is critical to pleasing customers who have a low tolerance for crappy software. If you aren't implementing new ways of shortening software development and delivery cycles, you can bet that your competition is. Don't forget, continuous improvement is the goal. Continuous delivery is just one result of those efforts.

6. Continuous delivery means releasing software every five minutes

The term "continuous" is ambiguous. Not even the shiniest of DevOps unicorns such as Amazon, Netflix, and Etsy, are delivering new versions of their software continuously. What's actually taking place within these organizations is that they've achieved a level of systems and process confidence that allows them to release new software when required. That may mean releasing new code once every two or three weeks, or it may mean releasing several times a day.

Facebook's motto is "Ship Often." It's not specific to a time frame. At any given time, Facebook can [roll out new enhancements](#) or changes to its service for a variety of reasons. Refusing to roll back, if a problem is detected, a fix is engineered and released to the public as soon as possible. It is always improving.

7. DevOps is only for unicorns

The term "unicorn" has been assigned not only to companies that have fully adopted the DevOps philosophy but also to those that have matured operationally to a point where they're in a state of continuous improvement, allowing for continuous delivery of software—and more. Yet, they've been doing it since before the term DevOps even existed.

Amazon, Facebook, Twitter, and Etsy are anointed as unicorns. Still, these organizations have set the standard on what can be accomplished, but they didn't get there overnight.

[Unicorns follow DevOps philosophy and principles](#), and realize that it's more about the people and building empathy across teams than any kind of tool or process. Otherwise, there's nothing all that special about these unicorns. In fact, most prefer to be called a sparkly DevOps horse instead.

8. DevOps is just for engineers and operators

As an evangelist, I'm a member of the VictorOps product team. However, a large part of what I do effectively falls under marketing. Although DevOps was originally proposed and brought to life by engineers and operators, nontechnical teams drank the Kool-Aid as well.

Sales, marketing, tech support, and even our C-level executives have bought into the importance of creating a culture of automation, sharing, and measurement. It's really cool to see an entire organization find a sense of tribal knowledge, situational awareness, and most importantly, empathy. By bringing additional teams into the DevOps fold, even greater improvements are found deeper within the organization's culture.

9. You can be certified in DevOps

DevOps isn't about tools and processes but rather building a culture of collaboration and empathy. Any attempts to create a certification program will spark a passionate debate on its merits and necessity.

The same has been said about the need for a DevOps manifesto. Like its predecessor, agile software development, more value is placed on individuals and interactions rather than processes and tools. Having a clear path for your [DevOps journey](#) is the desire of many, but those who have been doing this for quite some time feel very strongly that reducing the principles of DevOps down to a standardized list of key components doesn't follow the spirit of DevOps.

10. Adopting DevOps doesn't require C-level buy-in

Recently, Puppet Labs released its [2015 State of DevOps Report](#). One of the subjects covered in the report indicated that adoption from upper management and the grassroots is what you really need. While it may require a small "skunk works" team to get the ball moving toward a cultural shift, eventually you'll need your CEO and the rest of the executives on board, too. The sooner you can get everyone focusing on the transformation, the sooner you can begin that change.

In larger and more dispersed enterprises, wading through communication channels, paperwork, and red tape just to have a meaningful conversation with upper management can be a challenge. But it's not impossible. Just look at Target, Capital One, Disney, and countless other large organizations that have turned to DevOps to begin their transformation.

From the lengthy, informative, and entertaining conversation that took place on social media surrounding DevOps, I thought the discussion of DevOps myths was worth sharing. It's often easier to explain what DevOps isn't than what it is.

For those looking to learn more about DevOps and how you can begin bringing its concepts and principles to your organization, checking out the upcoming DevOps Enterprise Summit. I'll be there and would love to hear more about your own DevOps journey.