

# DEVOPS IN THE CLOUD



In today's fast-paced, digital-first business environment, [DevOps](#) is rapidly becoming as mainstream as the Cloud. Yet many companies are still grappling with questions around how to implement, foster

and benefit from a DevOps culture – particularly in cloud environments. This year, I've had the opportunity to attend DevOps events around the world and speak with many IT pros about the challenges they are facing.

Here are some of the most common questions I get asked.

## What is DevOps, and how does it relate to the Cloud?

I believe it's important to state that DevOps is not software, hardware or anything tangible that you can purchase. It is a cultural approach to how software is developed, tested, promoted and operated in a production environment. DevOps also includes collecting feedback from each stage of the delivery lifecycle (especially production), and using that input to infuse the next cycle of development and delivery. As such, DevOps is not limited to cloud, on-premises, or any other set of environmental attributes. Rather, it is focused on how people communicate and collaborate in the process of creating new business services underpinned by software.

Having said that, there certainly are environmental attributes – like cloud technology – that help:

- Make DevOps easier to implement
- Fulfil the promise of multi-disciplinary teams working collaboratively to accelerate delivery while also improving software quality
- Give all participants a greater sense of accomplishment and contribution

## How does DevOps in the Cloud differ from other types of DevOps?

DevOps in the Cloud does not fundamentally differ from DevOps in any other environment. But, cloud technology does help enable organizations to engage in and implement a DevOps culture.

## How does DevOps in the Cloud help organizations? What are the main benefits, and why?

It may be useful to think of the [Software Development Life Cycle \(SDLC\)](#) and how cloud helps each stage. For simplicity, let's just think about the build, test and run phases.

**Build** – Modern applications require complex technology stacks that take considerable effort to create and configure. Cloud technology has given developers unprecedented access to development environments in minutes or hours, instead of the weeks or months required in the past. A click or two is all it takes to instantiate a LAMP stack (**L**inux, **A**pache, **M**ySQL and **P**HP), and even hugely complex application ecosystems like SAP can now be provided to development teams almost instantaneously. Not only does this result in massive time savings, but on-demand availability delivers equally massive cost reductions by eliminating the need to keep systems running when they are not being used (or the habit of project teams to “hoard” resources).

Less tangible, but equally valuable, is the ability to easily explore and experiment with different approaches and technology components resulting from the ‘ease of access’ that cloud technology provides.

**Test** – The ideal DevOps approach is doing all the “creative” or manual work up front during build phase, and the rest of the SDLC is a fully automated “pipeline”. Automated testing is mandatory to make that vision a reality, and the ability to automatically build and tear down test environments is

mandatory to achieve automated testing. Here again, the Cloud provides unprecedented capabilities to instantiate test environments of ever increasing complexity as we proceed through the delivery pipeline. To make the transition from test to production a “non-event”, testing should be performed in an environment that is identical to production. In the past, this was a total impossibility with testing frequently being performed on a couple of servers with miniscule amounts of data and few of the other technologies and applications that make up the production environment. With Cloud, test environments that begin to approach the scope and complexity of production can be built quickly and cost-effectively.

**Run** – Deploying to production frequently requires new or additional infrastructure. Cloud provides unparalleled speed in building operational environments that are resource-efficient and highly scalable.

## **What are the successful steps organizations need to take to make DevOps in the Cloud a success?**

**Plan** – Like any new undertaking, planning is key. Today there are many cloud options and with diversity comes complexity. Organizations need to find a balance between maximizing development team productivity in the short term with long-term requirements of operating business services.

**Simplify** – Make it easy for all roles (developers, testers, infrastructure, security, operations, etc.) to get started and consume cloud services without having to become experts. An organization's approach to Cloud will most likely combine multiple vendors and/or technologies. Some level of abstraction across that complexity is necessary to let people start using cloud services without having to learn multiple, complex technology stacks.

**Monitor** – Ensure you have methods in place to monitor usage and manage costs.

**Measure** – Collect metrics and observe adoption. Have a realistic view of how, and if, cloud consumption is delivering the benefits described above. If some groups/projects are getting more value than others, extract best practices and proliferate them throughout the organization.

## **What are the best practices for DevOps in the Cloud, and why?**

Provide a self-service approach. This is critical to ensure that the speed Cloud can deliver is not bogged down with the traditional processes you were trying to escape in the first place.

Automate, Automate, Automate. Develop a culture of automation. Encourage everyone to find manual gates and roadblocks, and work on eliminating them.

And of course, measure everything.

## **What are the most important things an organization should know about DevOps in the Cloud?**

“Achieving” DevOps is a cultural transformation that people and the organization must go through. Cloud may be an enabler, but organizational support and commitment, a collaborative environment, a multi-disciplinary approach to project teams, and a “DevOps mindset” are far more important than the tools and technologies being used.

# How will DevOps in the Cloud change in the next 12-18 months?

DevOps and Cloud will both become ubiquitous and organizations will require greater discipline to formulate and adhere to a strategy that fosters some measure of stability and consistency. It's tempting for project teams to select the latest, coolest technology or to choose a cloud vendor that seems to offer the lowest cost or the simplest interface. However, in enterprises, there are myriad factors to consider when operationalizing business services, and diversity breeds complexity.

Cloud services and "server-less" models will become more prevalent and organizations will be more likely to adopt standard operating environments that include select technology stacks. Project teams will have the freedom to choose alternatives but those teams will then be obligated to provide ongoing support. This may lead to adopting open and flexible automation platforms that can provide a consistent layer of abstraction for all practitioners across the SDLC.

I hope these insights are helpful as you plot out your [digital transformation](#) journey. Are you currently implementing or planning to implement DevOps strategies at your company? How do Cloud technologies fit into the picture? I'd love to hear your feedback and questions!