DEVOPS FOR ENTERPRISES



DevOps is a set of practices that works to build collaboration among teams in order to develop, test, and roll out software more quickly and effectively. The concept of DevOps was originally developed in the context of agile web startups, and many question whether these practices can be applied to larger businesses and enterprises.

The reality, however, is that while specific solutions might vary based on organizational size and demands, the basic principles of DevOps can be applied to and add value to organizations of all sizes.

What is DevOps?

To fully appreciate what DevOps is, it's helpful to first focus on the division that has long existed in IT organizations between development teams and operations teams:

- In a traditional model, development teams are responsible for creating and changing software. They are primarily focused on innovation, new products, and new features.
- In contrast, operations teams are focused on system stability and accessibility. Their focus is more service-oriented, and they work to guarantee that systems are stable and consistently working effectively.

At their core, the teams seem to represent two competing interests, as one team works to consistently deliver software changes while the other works to maintain availability and function. In a

traditional model, the two departments are separate with organizations consistently having to choose between them.

DevOps, on the other hand, get away from this divided, silo mentality. DevOps instead focuses on collaboration and shared values across teams. The basic goal of DevOps is maximizing the development-to-operations value stream and ensuring an efficient flow of changes from development to operations.

The specific solutions organizations use to meet this goal vary, but often include:

- A culture of collaboration
- The use of automation
- A culture of continuous development

In addition, the DevOps model generally involves shared values across teams, including more frequent releases of new software and software changes, an increased emphasis on automation, and a sense of shared responsibility.

While DevOps looks different from organization to organization, at its core it's culture, tools, and practices that work to bring teams, individuals, processes, and products together in order to develop and roll out software faster and more effectively.

Does DevOps work for enterprises?

DevOps is often associated with small and agile organizations, and many of the traditional tools associated with DevOps are geared towards these types of organizations. And, the truth is that many of those tools and practices are not applicable to large enterprises with big teams, operational complexity, and lots of internal and external regulation. Making the shifts to DevOps even more difficult is the reality that enterprises generally have to deal with change control processes, release teams and approval gates, large environments, and a lot of teams and groups. All of these factors can make it challenging for enterprises to move to a DevOps model.

Despite these very real concerns, DevOps practices can work effectively for businesses of all sizes. While the specific solutions will look different based on the size of the organization, the basic principle of collaboration between development and operations teams to lead to better outcomes can be effectively scaled.

Not only is DevOps possible for enterprises, but in the current environment, it's a necessary shift for many organizations to support ongoing digital transformation, organizational growth, and increased capabilities.

How do you make DevOps effective at enterprise scale?

While making the shift to DevOps is important for staying competitive and meeting customer demands, implementing these practices on an enterprise scale is hard and requires some major shifts in the way teams plan, build, test, release, and manage software. For many organizations, the shift can feel overwhelming, but here are some things to help make this transition smooth and successful.

Don’t Ignore or Replace What Works

Most enterprises have already spent a substantial amount of time, money, and energy developing software and systems that work. When making the shift to DevOps and promoting an increased focus on continuous development and change, it can be tempting to always focus on what’s new and what’s next. However, doing so can be incredibly inefficient and ignores what’s already working well while meeting the organization’s needs.

Instead of overlooking or replacing what’s working, *build upon* what’s working. Organizations that are able to do this effectively are able to reduce risks, costs, and the all-important time-to-value metric.

Prioritize Building Confidence Across the Organization

A shift to DevOps can leave many across the organization concerned about unnecessary risks and potential negative impacts on customers. Ultimately, however, this shift is aimed at enabling enterprises to continue to be competitive and innovative without being held back by concerns about undue risks.

To effectively and comfortably make this shift, teams need to prioritize consistency, quality, and security from the very beginning. Addressing and preventing problems before they arise can help to make this organizational shift as seamless as possible. Further, ensuring that quality and consistency are always at the center of this change can help to build confidence throughout the organization, ensuring a successful transition.

Identify and Deliver the Right Outputs

It’s always important for IT organizations to focus on the right outputs. Many times outputs are centered around finishing a project or a product when instead they should be focused on targeted and measurable business outcomes.

When it comes to DevOps, generally the outputs should be focused on the benefits that customers receive. From the start of any project, it’s important for teams to understand customer needs thoroughly and to identify the specific need that is being addressed with a project or product. To ensure that teams stay on track to meet these goals, it’s necessary to consistently revisit these objectives and evaluate the progress towards them, which often includes course corrections. Implementing a continuous feedback system is an ideal way to ensure that all teams are on track to deliver the right outputs. However, regardless of the system that your team uses, it’s essential to focus consistently on identifying and delivering the right outputs.

Limit Operational Friction

DevOps aim to have a quick flow of changes across development and operations teams to stay competitive and meet customer demands. Doing this effectively means working efficiently across teams and throughout the organization. Eliminating any operational friction that will interfere with this objective makes it easier for teams to meet demands, increase automation, and develop organization-wide systems. As a result, proactively reducing operational friction among everything from teams to departments to vendors will help enable an effective transition to a DevOps model.

Share Progress

As discussed, the shift to DevOps is not easy, especially for large organizations. One way to ensure that the transition is successful is to share progress regularly. This sharing of information doesn't have to be formal and can be as simple as regular status updates. However, it's important for there to be visible progress and results seen throughout the organization. To meet this need, consistent progress sharing is essential.

Many large organizations have successfully made the transition to DevOps practices, serving as proof points that this concept is applicable and effective for enterprises. While the shift might not be an easy one, it's an important change to make, ensuring that organizations stay competitive while timely meeting customer needs. Being strategic about making this shift and proactively planning for its success can make this transition smoother and easier.

Additional resources

For more on DevOps, explore these resources:

- BMC DevOps Blog
- DevOps Guide, with 25+ articles on tips, best practices, and more
- The State of DevOps in 2020
- What Is Cloud Native DevOps?