Learning how to get started driving DevOps in your company, including how to overcome resistance to change.

View more content in this series

Driving the DevOps transformation can be challenging, but it is a rewarding endeavor. Achieving success requires that you focus on the business drivers for DevOps and find the right focus by assessing existing best practices and choosing the best pilot projects to demonstrate success. Ultimately, you want to take view of the entire software and systems lifecycle. Here in Part 8 of this series, you'll learn how to get started driving DevOps in your company.

In my work as a DevOps evangelist, I am beginning to feel that my colleagues believe DevOps is the single most important factor in solving any IT-related challenge. They are not wrong, and the reasons why DevOps is so effective are becoming increasingly apparent. But DevOps also threatens the status quo, and many of my colleagues from other companies often ask me for help in convincing their organizations to adopt DevOps best practices.

IT managers across the industry are enthusiastically embracing DevOps principles and practices. Even so, confusion about what DevOps is and how to most effectively implement DevOps best practices still exist. Many efforts fail because existing stakeholders aren't comfortable with change. They also perceive DevOps as losing control over the function and group in which they have been working. I have seen operations groups fight back against DevOps, feeling that it threatened their very existence — that is until they realize just how much DevOps benefits them personally. The first thing that you need to do is to understand the goals of DevOps and how to measure its success.

Goals and measuring success

DevOps focuses on improving the way organizations update their applications, which enables them to deploy code more often and improve the reliability of their systems. The most common measure of a successful DevOps transformation is a completely automated deployment pipeline that can deliver application changes, including emergency bug fixes, as often as necessary to
support business goals and objectives. DevOps also significantly enhances system security by providing a deterministic approach to verify that the correct code has been deployed and to ensure that there are no unauthorized changes through either human error or malicious intent. To really understand the value of DevOps, you need to consider the business drivers that make this methodology compelling.

We live in a challenging and dynamic business environment. Companies are obliged to constantly deliver complex functionality that meets and exceeds the expectations of their demanding and tech-savvy customer base. DevOps provides a competitive advantage by empowering agility through the rapid and reliable delivery of business features. Companies with mature DevOps capabilities can respond to business pressures better and enjoy significantly more reliable and secure systems infrastructures. Still, many companies struggle to understand what DevOps is all about, and misinformation is a major impediment to successful DevOps adoption. If you expect to overcome resistance to change and enable a successful DevOps adoption, you need to ensure that you communicate what DevOps is all about and how it helps to enable business success.

What is DevOps?

DevOps is a loosely defined set of principles and practices that help teams, including development and operations, communicate and collaborate more efficiently. The effective DevOps approach is to create a high-performance cross-functional team that combines the expertise of development and operations technology professionals. Uniting smart people from different groups provides synergy and a highly effective working group that can tackle almost any challenge with a positive attitude. DevOps, by definition, means that you have both development and operations represented. Development focuses on developing new functionality using the latest technologies. Operations has the job of ensuring reliable systems. When you put these two important perspectives together, you get amazing results. But DevOps is not just about Dev and Ops; DevOps should include stakeholders from other silos as well, such as:

- Product management
- QA
- Testing
- Information security

Including all of the right stakeholders is an absolute necessity, as is communicating the DevOps message in a clear, accurate way. Too often, technology professionals use DevOps buzzwords in a vague and confusing way.

Misinformation

Misinformation about DevOps exists. Even some folks who are trying to advance their own agendas, such as using DevOps to bypass operation and IT controls, often required by federal laws or industry best practices, are misinformed. Many big banks do great at implementing DevOps, but this does not mean individual developers can push their code to production bypassing federal regulatory requirements for a separation of duties. DevOps deployed well enhances your ability to demonstrate compliance with regulatory requirements and common IT audit controls, such as those required by Section 404 of the Sarbanes-Oxley Act of 2002.
Developing a clear message is important, along with selecting the right projects to demonstrate the value and benefits of DevOps best practices. The best way to determine how to implement DevOps is to start with an assessment of your current best practices.

**Conducting the DevOps assessment and overcoming resistance**

When I work with any technology group to implement process improvement initiatives, I start by asking what is working well and what could be improved. The folks who are closest to the real work going on are often in the best position to provide insight into the existing practices that should not be changed (at least not in the beginning) and what pain points should have priority. I meet with a variety of stakeholders, from development managers to testers, and I ask them to explain what is going well and what can be improved. The DevOps assessment provides valuable insight into exactly what practices need improving in any organization. The key here is that process improvement suggestions from within the group are accepted more than ideas from an outsider.

To overcome resistance to change, find the right initiatives within the organization to start with and that already have some supporting stakeholders. When I do an assessment, I compare existing practices to industry standards (such as ISO/IEEE) and frameworks (ITIL v3, ISACA COBIT) to help to identify process improvement initiatives.

Immerse yourself in DevOps by taking the IBM DevOps self-assessment and reading "Implement ITIL with DevOps."

Enterprise adoption requires that you have strategies to scale your DevOps best practices throughout the enterprise. I usually do this by developing common criteria to help each group understand what it needs to work on. Ideally, you can create and utilize your own process maturity framework that contains criteria that is relevant and important to your organization. I start by ensuring that teams communicate and collaborate effectively through high-performance cross-functional teams. Then I review their configuration management best practices. I also make sure I review their automated application testing. I start with something small they can easily achieve, and I focus on showing success early in the process.

I always pick a small- to medium-size project that provides the best chance of showing success. After a team sees that things can improve for the better, they become more confident and ready to tackle any challenge. If you pick a project that is difficult or takes too long to implement, you run the risk of the team losing steam before you've demonstrated success. For example, I might think that the team's choice of a version control system is weak, but fixing the scripts used for deployment will show results a lot faster. Choosing the right pilot project is important for your long-term success. You also need to take a broad view when you plan your DevOps initiatives.

Learn about the DevOps lifecycle by reading "Drive Application Lifecycle Management with DevOps."

DevOps is not just about development and operations. When planning your DevOps initiatives, consider the entire application lifecycle and involve all of the stakeholders. I have often found that product managers give me a new and fresh perspective that helps me focus on the most important objectives. Sometimes the best ideas come from your help desk or other support functions.
Implementing DevOps means you are facilitating better communication between different silos in the organization and helping them communicate and collaborate more effectively.

For me, reliability is often the first priority.

**Reliability**

Too often, DevOps initiatives start by trying to tackle unrealistic goals, such as single push-button deploys or what might even be an unnecessary goal like continuous deployment. My first priority is ensuring that deployments are completely reliable even if the deployment is not completely automated to a point where it is a single push-button job with no operator intervention. This means that a first step is acceptable, even if someone has to run some scripts and click **Enter** a few times — reliability is the most important objective. I have met people who were resigned to problems during the deployment process. These folks often believe that we must do all deployments on Friday nights so that if something goes wrong we have the whole weekend to fix the problem (assuming of course that your systems can be down over the weekend). The first thing I do is to break the deployment into smaller pieces and schedule them for one or two days in the middle of the week. This involves a cultural change showing all of the stakeholders that completely reliable deployments are not only possible but a must-have. I always remind people that life-support systems and nuclear power plants all have complex software that has to be updated, and obviously the goal must be completely reliable deployments. After your deployments are completely reliable, you can graduate to continuous delivery and continuous deployment.

**Security**

The next consideration is using DevOps to help ensure secure deployments. This is where DevOps partners with your data security folks and can produce dramatic results. My effort is to make sure I can verify that all of the correct files were deployed and detect any unauthorized changes through human error or malicious intent.

The next consideration is ensuring that your QA and testing function can keep pace with the rapid pace of DevOps automated deployments. QA and testing DevOps cannot survive without robust and comprehensive automated testing and this should include not only unit and functional regression testing but more complete API and service virtualization testing as well. Your DevOps transformation is at risk of failure if your automated testing can’t keep up with the pace of rapid deployments that comes with any DevOps transformation.
DevOps in the cloud

It is almost impossible to imagine trying to manage cloud-based infrastructures without DevOps best practices. The cloud benefits greatly from DevOps practices that you get with infrastructure as code and provides dramatic and demonstratable results.

DevOps provides a powerful set of principles and practices that help address the challenges inherent in cloud-based computing. Read "DevOps in the cloud" to learn more.

Conclusion

DevOps provides a robust set of principles and practices that help improve communication and collaboration. Driving the DevOps adoption and overcoming resistance to change requires that you understand which initiatives will demonstrate value to your organization and help you build up support for scaling your DevOps best practices throughout the enterprise.
Related topics

- What is ITIL?
- Configuration Management Best Practices: Practical Methods that Work in the Real World

© Copyright IBM Corporation 2016
Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)