How DevOps Drives the Agile ALM

With some of the recent enterprise software rollout disasters, it is time to get back to basics with utilizing DevOps with your agile ALM.

by Bob Aiello and Leslie Sachs | bob.aiello@ieee.org and LeslieASachs@gmail.com

DevOps is getting a lot of attention these days as the benefits of improving communication and collaboration between development and operations is becoming readily apparent. The need for DevOps is especially obvious in the wake of software and systems glitches that have impacted major financial services, including large banks, trading firms, and the trading exchanges themselves. The healthcare.gov website is the latest high-profile software system that failed to meet its goals, due in part to problems related to software and systems reliability. These problems lead us to wonder if the technology industry is really capable of creating reliable enterprise software.

> "One of the most effective approaches to DevOps involves moving the automation of the application build, package, and deployment upstream to the beginning stages of the software development lifecycle."

Software developers, by and large, are very smart and highly skilled technology professionals. Equally skilled are the operations experts who establish the IT controls necessary to ensure that large-scale systems are available continuously, scaling them to meet the capacity demands required during peak usage. Both development and operations teams bring much expertise to the table, but they have fundamentally different perspectives. Developers are expected to write code that implements new features, while operations is charged with seeing that systems maintain a high degree of reliability even under heavy system load. In addition, there are actually many other key stakeholders, without whom we could never create robust enterprise-wide software systems.

Technology organizations consist of a wide array of professionals, from business analysts to QA and testing professionals, each of whom is essential to the successful development of complex software systems. Coordinating their work is no easy task, and most organizations employ a number of full-time project managers who track and report on the work being accomplished by each member of the team. But managing the development of large-scale software systems involves a lot more than just creating Gantt charts and resource reports.

Developing large-scale software systems requires the coordination of hundreds (or even thousands) of tasks, each with its own set of dependencies that are rarely completely understood up front. Helping the entire team understand what needs to be done is what application lifecycle management (ALM) is all about. With many organizations embracing the enhanced productivity and quality that comes from employing agile principles, the agile ALM is becoming an essential software methodology. It turns out that agile ALM benefits greatly from the very same principles behind the DevOps revolution.

Enhancing collaboration between development and operations works because each group brings a set of complementary skills to the table that, when integrated, enhances both productivity and quality. Developers know the technology they have been creating better than anyone else. They should; most developers have months to get up to speed and focus on changing technologies and software development frameworks. Developers often get to choose which technologies to use in creating systems and then focus on building their expertise on a daily basis. Operations professionals need this information in order to be successful. The operations team understands what happens when a critical system is unavailable for any period time. One of the most effective approaches to DevOps involves moving the automation of the application build, package, and deployment upstream to the beginning stages of the software development lifecycle—an industry best practice long before DevOps became as popular as it is today. [1]

True DevOps groups involve development and operations teams working collaboratively to automate the complete application build, package, and deployment process, creating what is becoming known as "the deployment pipeline." This practice enables the team to best support iterative development by emphasizing the synergy between development and operations. The rise of DevOps demonstrates the powerful synergy that can be achieved with close collaboration between development and operations.

Recognizing that the other stakeholders also possess expertise, it becomes clear why improved collaboration and communication throughout the complete lifecycle can help the entire team achieve a high degree of personal productivity and effectiveness. The fact is that encouraging other stakeholders to apply these principles often yields significant benefits as well. Each team member can benefit significantly by better communication and collaboration.

Information security (InfoSec) is often in the position of trying to ensure the integrity of complex systems they do not completely understand. Similarly, testing and QA professionals are expected to ensure that complex systems are defect-free. Each stakeholder on the team brings expertise, and successful companies are realizing that DevOps really applies to the entire ALM. This is particularly apparent in an agile development methodology embracing iterative development.

The agile ALM helps brings structure to the demanding and constantly changing application lifecycle that is part of any agile development effort. While the level of ceremony and software process maturity may vary a great deal from one project to another, there is always a need for just enough structure so that each stakeholder understands what he needs to do on a day-to-day basis. The Scrum methodology provides an excellent basis for communicating and sharing knowledge and is used successfully by many highly effective self-organizing teams. In an agile ALM, information security professionals have the ability to start looking at working milestones of the system much earlier in the process. This enables InfoSec to understand the interfaces and core requirements for ensuring systems security. Similarly, QA and testing professionals who get involved early in the process build quality in from the beginning [2] and are better equipped to develop and automate robust testing frameworks. The DevOps environment involves moving work upstream and creating milestone releases that have fully automated deployment pipelines and robust testing frameworks, including information security.

Within the agile ALM, the customer or his representative is also a key stakeholder. Applying DevOps to the agile ALM ensures that systems meet their specifications and also satisfy their business purpose. Many technology teams struggle to fully understand the business requirements up front, and iterative development provides an excellent means to allow business experts early access to release milestones to ensure that the system meets its intended requirements and, more importantly, that the requirements are indeed correct. Competitive pressures, including new and ever-changing regulatory requirements, mean that understanding a given system's requirements can be very complicated and involve aiming for a moving target. As a result, DevOps applies to a lot more than just development and operations. The agile ALM needs to ensure that each stakeholder embraces the collaborative synergy of sharing knowledge and better communication as the core lesson that DevOps brings to the table! {end}

Sticky Notes

Click here to read more at StickyMinds.com.

- References