▶ *- Solution Spotlight*

ACCELERATING APPLICATION DEPLOYMENT WITH DEVOPS

TechTarget

> **SearchSoftwareQuality**

B

RINGING TOGETHER DEVELOPMENT, testing and operations can help organizations address many traditional software quality and application deployment challenges.

This essential E-Guide highlights the benefits of a DevOps approach and explores how business can successfully integrate their teams to increase collaboration, streamline testing processes and reduce risks.

SPONSORED BY **IBM.**

> SearchSoftwareQuality

MOVING TO DEVOPS SPEEDS DEPLOYMENT AND BOOSTS ROI

Melanie Webb

Is there a need to change traditional testing approaches? What is DevOps? Who's using it? These are a couple of the questions addressed by Manoj Narayanan of Cognizant Technology Solutions at STAREAST 2012 in his session, "Testing in the DevOps World of Continuous Delivery." He explained the benefits of DevOps and how it impacts the organization.

Traditional testing approaches are geared toward sequential delivery, which leads to an emphasis on discrete skill sets and therefore minimizes cross-functional behavior. The sharing of ideas and mutually beneficial collaboration is limited as there are barriers between development, testing and deployment. Narayanan identified the drawbacks to the sequential delivery approach as a longer wait time and higher risk; essentially a developer may be saying, "I don't know if what I'm building is what the end user wants."

In order to meet the objectives of business – to reduce risk, respond fast and beat competitors – Narayanan supports moving from a discrete delivery

SPONSORED BY **IBM.**

approach to a continuous delivery approach. The language changes from "build it right" to "build the right it."

AGILE DEVELOPMENT AND DEVOPS

Agile development offers one path to continuous delivery as it brings developers and testers together and focuses on iterative development. Yet, according to Narayanan, "the 'last mile problem' still exists." The wall separating deployment from the development and testing side encourages a long time between testing and deployment, and furthermore, testers do not necessarily have a systems administration skill set.

DevOps offers a potential solution to this integration challenge, bringing together development, testing and operations all on the same team, a team capable of playing all the different roles.

"Like Agile, this approach aims at getting features deployment ready at a high frequency. But where Agile primarily focuses on the functional and non-functional readiness of the application, DevOps takes it one step further and ensures the Operational and Business readiness as well," Narayanan writes in his blog.

Narayanan noted several websites that are currently employing the DevOps

model, sites like Facebook, Etsy, Orbitz, Groupon and Flickr. He explained that the frontrunners have tended to be Web 2.0 firms with strong reliance on eCommerce where fast changes and improved response time are paramount. He mentioned that Netflix is involved in this space as well, actually going with a NoOps approach.

When it comes to testing, the change to the DevOps environment must be gradual, and the talent involved will need to acquire new skills and adapt to new responsibilities. Testers need to gain knowledge in development languages— which fortunately is a bit easier with user-friendly tools like Python and Cucumber – and they also need to learn deployment processes and tools. Developers and systems administrators, on the other hand, must learn about test processes, design techniques and related tools.

MAKING THE CHANGE TO DEVOPS

Shifting to DevOps also makes continuous integration mandatory, according to Narayanan. He explained that in test-driven development we expect to fail first, but those errors inform the next cycle of development and testing. He emphasized that teams must have strong discipline around the single source code repository. The build process can be automated, resulting in a fast build.

In order for all of these elements to work effectively, he highlighted the need for transparency; everyone on the team needs to know what is happening.

The process is also affected by the heavy reliance on innovative automation, which is embedded early in the lifecycle. "'Smart testing' is dissolving the boundaries of traditional system and integration testing," said Narayanan. Teams can now "leverage an optimal mix of automation across the lifecycle," which includes automated unit testing, automated service layer testing and automated regression testing. Continuous integration is further facilitated by release management automation.

GOVERNANCE, TECHNOLOGY AND STAKEHOLDER BUY-IN

When it comes to governance, Narayanan holds that teams work best in pods which are comprised of 'jacks of all arts;' these pods are centrally coordinated and are characterized by ongoing communication and collaboration. The incremental changes taking places are driven by business goals. One result of this pod arrangement is that a greater need for specialists arises; they take on the challenging sub-tasks that the regular pod members may not be equipped to handle. Pod members can also identify repeatable tasks and automate them. This approach works especially well in smaller and mid-sized organizations.

The cultural impact of DevOps also appears in the technology the organiza-tion uses. DevOps teams need user-friendly tools that all team members—de-velopers, testers and systems administrators— can learn and use. In addition, the focus on automation and reusability intensifies.

In order for stakeholders to buy in to these changes, communication is key, and stakeholder buy-in, likewise, is key to the success of this model. The suc-cess may be viewed in terms of speed and ROI—as the deployment happens faster, ROI improves.

USING DEVOPS APPROACHES FOR PRE-RELEASE AND PRODUCTION SOFTWARE QUALITY

Jan Stafford

How many helpdesk (HD) tickets does your IT team get right after software is released into production? Some organizations' post-release HD volume is 50% of the annual total.

Too many organizations today strive to create software quality in development and then are unable to preserve it during release and in production. This problem is common, and one root cause is development and operations teams not working together before a software release. The teams are unclear about what production QA processes are and who should be handling them, according to experts.

Disorganization is widespread when it comes to getting applications into production and maintaining application quality once it's there, according to Randy Clark, CMO of IT Process Automation software company UC4. He has witnessed the above-mentioned helpdesk ticket explosions.

Clark, ALM Consultant Howard Deiner and Quality Director Joe Strazzere

explore DevOps strategies and tactics for working together to achieve and maintain software quality in this article.

PRE-DEPLOYMENT DEVOPS TEAMWORK

"If you have any kind of a back end that's going to be hit by lots of people, the risk of deployment becomes significant," said Deiner. "That's why a lot of DevOps teamwork should happen in the cusp right before deployment."

What goes on when Dev and Ops teams meet? For one thing, Dev and Ops teams collaborate on monitoring strategies before release so Ops knows what is going on with production apps. "That means monitoring key capabilities, some nonfunctional requirements and other operational aspects," said Howard Deiner.

The development team also provides operations with migration scripts, so Ops can test before production, according to QA director Joe Strazzere. Ops should be briefed on development testing, too. Clear responsibilities should be set. For example, development teams should not be messing with schemas as the app goes into production, as it's the job of database administrators to monitor them.

Before application release, Strazzere's operations engineers require the

development team to provide preparation assistance. "Production will accept a new or modified application only after being trained and only after a Run Book has been created for the application," he said. The Run Book explains system details and how production is expected to respond in certain cases.

Pre-deployment and production testing is done by both development and operations teams, but they are testing different things at different times. Production testing centers on processes, like planning, control and automation, said UC4's Clark. Development teams don't get involved in processes, but test application code and other product-centric tests.

Strazzere's production engineers test code during the pre-release and re-lease period. This short-duration, release testing is usually handled only by QA engineers. During this period, Deiner noted, developers move from nitty-gritty tests, like unit tests, to smoke tests, a broad, shallower test format to ensure quality is stable.

Once an application is in production, most organizations move the responsibility for it from the development team to operations. Deiner would like the lines to be blurred a bit and advising extending continuous integration into production application quality processes. "With CI, we can automate our re-lease processes from code check-in to deployment." Extending CI servers from

testing, to release and then into production processes can improve software quality throughout the SDLC.

QA BY ANY OTHER NAME

As application QA responsibility moves from development to operations, what is the process of production QA and testing called and who's in charge? SSQ sources call this production QA, production test and ALM test management, and the person in charge is the ALM QA director, project manager and software architect, among others.

Other monikers for the process and manager abound. In a post in the SQA&T blog, a test pro wrote that he and his manager were stumped about what to call production QA in their status reports. Responding, software pros said their organizations used such terms as operational quality, production monitoring, and production support team. One suggested that the process be called Customer Experience Assurance. Others on SQA&T have seen it called production monitoring, production support, sustained engineering and site confidence.

PRODUCTION SOFTWARE QA APPROACHES

Strazzere's production QA process is called production monitoring and is done by production engineers who report to the director of production. Production engineers monitor production systems to ensure that time-critical process proceed as expected. They jump in to deal with the occasional interruptions so that our service-level agreements (SLAs) can be met.

There is just a little overlap of production monitoring and development QA in Strazzere's approach. "We have lower-level testers that check some of the weekly or monthly files before they are moved along to the next part of the system," he said. "But that 'QC' role is diminishing as we automate more and more of the systems."

Automating production monitoring and quality control provides such benefits as being able to get rid of manual intervention, thereby speeding intervention timeliness and reducing labor and downtime costs. Also, Strazzere said, new systems and enhancements to existing systems can get into production with a smaller investment in infrastructure.

Automation can also solve other problems. For example, manual in-production actions don't scale. "If you have too many of them, you must have a huge team ready to respond at any time," Strazzere said. "They are best used in

smaller doses."

Automation can also provide an effective feedback loop. Otherwise, the fixes that the production engineers apply won't make their way into a future revision of the code, and manual intervention could be required indefinitely.

While he's bringing automation into production QA and elsewhere, Strazzere is not assuming that he should automate everything. "Sometimes it makes sense to have real people, instead of automation, as a small part of the system," he said. "This is a business decision, not always a technology decision."

AGILE METHODS IN PRODUCTION QA

Strazzere's organization doesn't use any formal Agile methods, but does use Agile's less formal methods. "For example, we have manual production processes that revolve around an Annual task. When we are doing 'Annuals,' we have daily stand-up meetings, and very short -- usually two-day -- time-boxed turnaround for some production tasks of creating output and checking them," he said. "But we do little that most software groups would term Agile with a capital A."

Often the impact of Agile is seen in just the way Strazzere describes, Deiner said. Over the years, parts of the methodology have become common practice. Agile's primary contribution is enabling the DevOps process in the first place,

he added. "There were two camps, and both didn't like each other," he said. Agile has created a more collaborative environment. Today, he said, developers have more understanding of the operation's needs and problems, and operations people are more involved in development. The result is better application quality once an application goes into production and better monitoring and quality assurance once software is in production.

SPONSORED BY IBM

> **Search**SoftwareQuality

FREE RESOURCES FOR TECHNOLOGY PROFESSIONALS

TechTarget publishes targeted technology media that address your need for information and resources for researching products, developing strategy and making cost-effective purchase decisions. Our network of technology-specific Web sites gives you access to industry experts, independent content and analysis and the Web's largest library of vendor-provided white papers, webcasts, podcasts, videos, virtual trade shows, research reports and more —drawing on the rich R&D resources of technology providers to address market trends, challenges and solutions. Our live events and virtual seminars give you access to vendor neutral, expert commentary and advice on the issues and challenges you face daily. Our social community IT Knowledge Exchange allows you to share real world information in real time with peers and experts.

WHAT MAKES TECHTARGET UNIQUE?

TechTarget is squarely focused on the enterprise IT space. Our team of editors and network of industry experts provide the richest, most relevant content to IT professionals and management. We leverage the immediacy of the Web, the networking and face-to-face opportunities of events and virtual events, and the ability to interact with peers—all to create compelling and actionable information for enterprise IT professionals across all industries and markets.

SPONSORED BY **IBM**