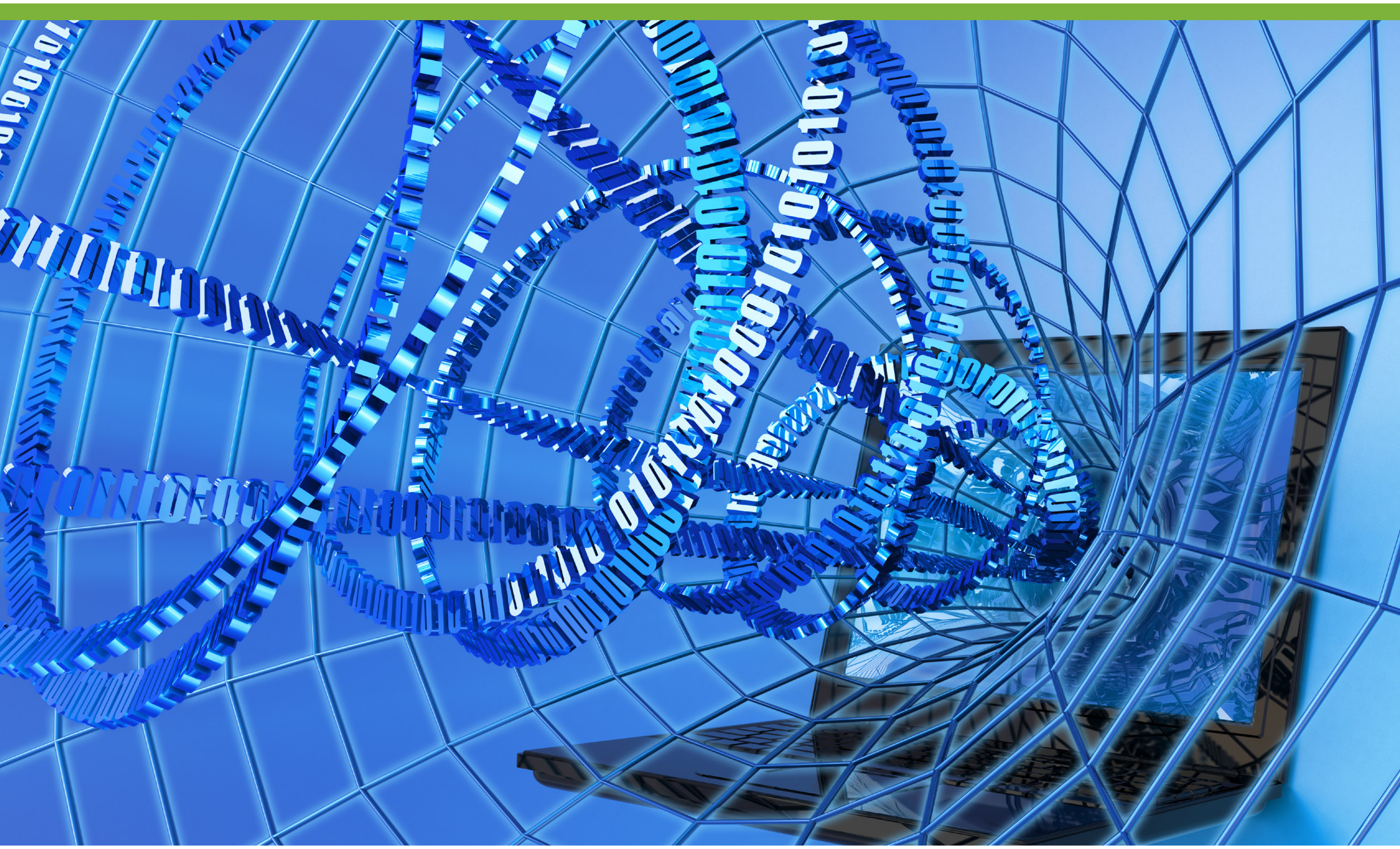
WHITEPAPER

ADTMAG
APPLICATION DEVELOPMENT TRENDS



THE TOP 5 DEVOPS CHALLENGES

DELPHIX

www.delphix.com

IN MANY ORGANIZATIONS, the key challenge in deploying DevOps is a data management project. IT professionals are familiar with the challenge of needing data for their environment. In DevOps, this familiar need for data grows exponentially. Even if data is not a bottleneck in your organization today, with the inevitable increase in automation, data will soon become a big bottleneck.

The data challenge associated with DevOps includes the need to deliver faster releases while scaling with demand. To walk this tightrope, DevOps groups test massive sets of data activities in parallel. The challenge, however, becomes whether the data can be supplied to support testing without exponential hardware spend or IT operations time.

Lean management and continuous delivery practices hold incredible promise: the prospect of a 98% reduction in code failures, and recovery accelerations from weeks to days. But in order to reach these benefits, DevOps transformation owners need to navigate five major challenges.

CHALLENGE #1: WAITING TO HURRY UP

Speed is what drives the demand for DevOps. However, adding speed to the data tier is a challenge many DevOps environments face. Without being able to manage and deliver copies of production data and databases as rapidly as possible your DevOps could hit a significant development speed bump.

The key to avoiding this obstacle is to understand that most speed-related problems in production seem to cluster around the transfer of data among teams. Whether the data is in a separate QA team, or the DevOps team is dealing with the hand-off to pre-production, this transfer of data often causes DevOps to come to a halt.

It's a perennial problem in DevOps: The development team is held up waiting for data.

To overcome this bottleneck, DevOps organizations need access to full and scalable virtual copies of production data on demand. Without these types of solutions it can become difficult to stand-up the data in any environment and it may cause DevOps teams to resort to starting development without a robust data testing environment, or to build-out a limited synthetic data environment to test against. This, in turn, leads to poor testing, poor code, and poor business outcomes.

EVEN IF DATA IS NOT A BOTTLENECK IN YOUR ORGANIZATION TODAY, **WITH THE INEVITABLE INCREASE IN AUTOMATION, DATA WILL SOON BECOME A BIG BOTTLENECK.**

CHALLENGE #2: INTEGRATING SOLUTIONS IN INTEGRATION TESTING

Most enterprise applications interact with multiple databases, either directly or indirectly. Thus, tests have to be run against an entire simulated data ecoysystem. This step poses its own challenges, centered on acquiring consistent, simultaneous copies of multiple distinct production databases.

This need for data copies presents both a technical and organizational challenge because different databases belong to different teams. Now in order to acquire the copies of data, the organization has to decide if they can even permit an outage to acquire the data copies. The organization must determine when it will be a good time for that outage across all departments, and whether from a technical standpoint the various teams have the capabilities to take a copy of the data during the outage.

Despite these challenges, having a final integration state is worth the effort. Without this stage development teams are forced to test different applications against limited copies of a few databases, and ultimately this increases the likelihood that bugs will flow through to production. A final integration testing stage is therefore a best practice in the legacy universe.

There are, however, additional challenges with integration testing that we classify as Contention and Complication.

THE CONTENTION CHALLENGE

Let's say you have a legacy development group working on three apps and despite the fact that each app has its own database all three apps interact with each other. Usually, the teams will have different development environments, and developers will focus on making sure their code is correct for the corresponding database environment. But their integration testing environment will have three databases in order to validate the code's functionality across applications.

This means that the integration environment, and indeed the whole integration testing setup, is a resource held in common between different teams. That means that a team as it's moving forward needs not only to spend time in this integration testing environment but they might have to wait for another team to get out of the environment before they can get in.

THE COMPLICATION CHALLENGE

This challenge stems from the fact that integration testing in fact works too well. Integration testing often reveals bugs that haven't yet been discovered. But since integration testing is the final step in the software development process at its completion you get knocked back to the beginning. This produces a need for developers that catch bugs at the end of the release cycle to go all the way back to the beginning and scramble to complete the process.

When organizations move to a continuous integration or continuous deployment practice they lose multiple steps of the processes. Organizations no longer move through multiple steps, testing code against a subset, then formally in QA, then finally in integration environments. Instead, as soon as code's developed, it goes live immediately.

This means the qualitative need of integration testing, which includes the ability to run tests against the full-spectrum of code that will be running in production, gets moved forward in the process. This involves the whole team. Traditional development involved developers working on a release two weeks from the end of the quarter, placing all their changes together, and running that as the release against the integration testing environment. They could then remediate bugs detected there. In DevOps, the same number of developers works in parallel, check in code, and expect it to go live.

In order to achieve this speed and elimination of steps, organizations need to be able to accomplish the same ends that a single integration testing environment provides, automatically throughout the DevOps process. This requires much greater data agility.

This level of data agility can be achieved with solutions that provide the ability to make copies of any data source that it's connected to, at any particular point in time. By leveraging such a solution organizations can move away from dealing with elaborate scheduling operations. Then there is no need for a single integration environment that is resource intensive and complicated to maintain. The organization has now moved to an automated environment with less complexity.

Increasingly, DevOps organizations have to provide developers with the data they need to work on their app, the bucket containing virtual copies of the databases, and the power to hit refresh to get those databases synchronized to the most recent time available. Without these capabilities, continuous integration is an impossible target.

CHALLENGE #3: MITIGATING SAFE RISKS

There is a danger that the DevOps approach may result in security vulnerabilities by increasing the number of people dealing with sensitive data. This is especially problematic if outside consultants and offshore contractors are working on the project.

At heart of the issue is a culture clash between application delivery teams motivated to deliver new and innovative capabilities quickly and security teams that are determined to ensure that systems and data are secure.

The tension between these goals is real, and DevOps owners need to work with IT partners to navigate it. Some DevOps practices are riskier than non-DevOps practices. DevOps willingness, for example, to extend access to end users goes against most IT security practices.

With advanced data masking solutions, it is possible to protect data involved in DevOps projects. Masking takes sensitive, personally identifiable information such as names, addresses, payment information, and credit card numbers and replaces those values with realistic but fictitious data. By masking data before it is sent to downstream environments, sensitive information is removed and the surface area of risk decreases.

CHALLENGE #4: WHAT IS THE REAL COST?

Is your DevOps transformation going to cost too much? What are the real cost centers associated with moving to DevOps practice?

As mentioned earlier, developers understand that the need for data is a challenge. And with Data often comes costs including the need for vastly more copies of production data, and vastly more frequent operations against those copies. In fact, DevOps accelerates the growth in data needs far beyond what traditional application development methods would have required.

Traditionally, within a legacy environment, organizations would have to go out and buy more storage to house the expanding data, as well as acquire more database administrators to service the subsequent increase in tickets. This can make it cost prohibitive to expand and scale. As organizations move to a DevOps environment where copies of data are needed throughout the process, and the organization moves from monthly to daily releases, IT operations managers may look at requests for expanded and expensive tickets and just say, "No!"

The goal is to deliver the type of speed required in DevOps, and corresponding data copies needed, without producing prohibitive costs. This requires a solution that:

- Can automate the steps that otherwise would require a massive number of administrators and
- Mitigates the hardware cost of a legacy environment and moves the organization to an environment where every developer has a copy of production data they need without the exorbitant storage spend.

An automated DevOps solution provides for the data requirements of a continuous integration or continuous deployment practice. This produces workforce efficiency that allows staff, including database administrators, to off-load the more repetitive and lowest value activities and focus on more important tasks.

TRADITIONALLY, WITHIN A LEGACY ENVIRONMENT, ORGANIZATIONS WOULD HAVE TO GO OUT AND **BUY MORE STORAGE TO HOUSE THE EXPANDING DATA**, AS WELL AS **ACQUIRE MORE DATABASE ADMINISTRATORS** TO SERVICE THE SUBSEQUENT INCREASE IN TICKETS.

CHALLENGE #5: INDEPENDENTLY WORK TOGETHER

Sharing copious amounts of data between large teams of developers, within a tight timeline, presents a collaboration challenge. Moreover, the operations which are ordinary for code, such as checking out the code version needed, sharing it with a colleague, switching between versions, and so forth are much harder to accomplish with the larger requirements for data.

The key to mitigating this challenge is to select a solution that enables this collaboration, including providing.

- Identical application configuration
- Identical automated testing
- Similar environments

It is important to have the ability to make parallel copies of identical data that is easy to share between different members of the team to facilitate collaboration.

PUTTING IT TOGETHER

The data challenge associated with DevOps at first glance appears onerous and insurmountable. However if organizations employ DevOps solutions from Delphix the data challenges become a thing of the past:

- Waiting for data is no longer an issue
- Data validation becomes more efficient
- Collaboration increases
- Production becomes more cost-effective

In DevOps not only do you have the need for data but that need grows exponentially. With Delphix data no longer becomes a bottleneck in your organization and organizations can take advantage of the DevOps efficiencies including:

- Delivering software faster
- Identifying bugs earlier
- Deploying software more frequently

Organizations that achieve this level of success emerge to find an organization where technology is ingrained within the DNA of the enterprise. If you're ready to take on data challenges and unlock the full potential of DevOps, **CLICK HERE TO LEARN MORE.**

ABOUT DELPHIX

Delphix is the market leader in data virtualization, helping organizations release applications up to 10x faster by delivering secure, virtualized data across the application lifecycle. More than 30 percent of the Global 500 use Delphix software to deliver data across development, testing and reporting environments, improving developer productivity and data security on premises or in the cloud. Delphix is headquartered in Menlo Park, California, with offices around the world, and can be found online at **www.delphix.com.**