DevOps for the Database 2015
A DBmaestro Survey
Whitepaper

DBmaestro
DevOps for Database

Executive Summary

The survey of over 450 IT professionals found that companies are increasingly adopting automation and continuous practices for software development. Survey participants also asserted to adopting these practices - to an also growing but lesser extent - for the database. However, based on subsequent responses, it became clear that only a very small percentage is actually performing safe continuous delivery practices for the database.

Continuous practices are strong and continuing to grow. 67% of respondents replied that they use continuous integration or build automation for their application. This represents an 11 point increase from a survey conducted last year by DBmaestro which found that 55% of respondents were implementing continuous delivery for their application code.

Continuous delivery for the Database is also growing but not as fast. When asked if they use continuous practices for their database changes, 34% answered in the affirmative, this was despite 81% answering in a previous survey that they believed it is possible to practice CD for the database.

Perception doesn't always meet reality. Despite 34% responding that they implement continuous processes for the database, when asked how their organization deploys database changes, only 13% had an automated process in place – a prerequisite for database continuous delivery.

Organizations are making frequent database changes. This lack of automation processes is despite two thirds responding that their companies make frequent database changes. In response to the questions of how often their organization makes changes to the database, 32% said daily and 37% responded weekly.

Developers see high risk in non-automated processes: When asked to rate the risk level of particular practices, 57% considered it high risk for the database to be out of sync with the version control repository (26% considered it medium risk), 57% said it was a high risk to override changes when integrating valid changes (28% said medium) and 44% said it was high risk to integrate asked to rate the risk level of particular practices, 57% considered it high risk for the database to be out of sync with the version control repository (26% considered it medium risk), 57% said it was a high risk to override changes when integrating valid changes (28% said medium) and 44% said it was high risk to integrate changes directly into the QA (28% said medium).

About the Survey

The "DevOps for Database" survey was conducted to get a deep insight into the progress and adoption of automation for database development and deployment. The survey focused on methodologies used in various stages of the development lifecycle.

The online survey was conducted in February and March 2015 and included over 450 participants from around the world.

The respondents included DBAs (20%), software developers (17%), Software Architects (13%), DBA managers (10%), senior managers (10%) and others, from diverse sectors including Software vendors (32%), financial (18%), government (11%), healthcare (7%) and more.

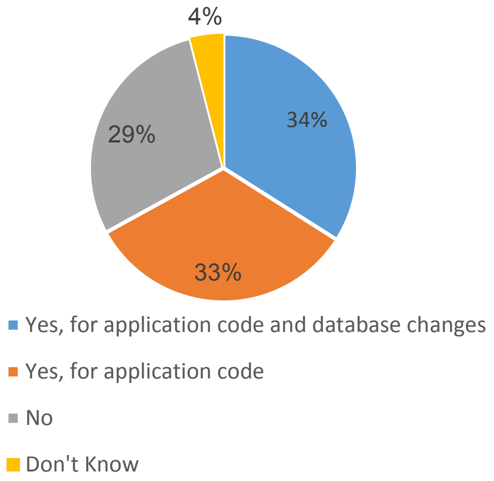
Some of the statistics presented here are from a previous survey conducted by DBmaestro in September of 2014.

Database Automation: Gap between perception and reality

Continuous delivery has continued to go mainstream, with two thirds implementing automation in their development. When asked about implementing continuous process in the database, that number halves to 34%. (Figure 1)

Continuous delivery is a concept in software and database development of a well-defined process, feedback loop and automation. Through automation, you can dynamically perform the same task over and over again, without human shortcuts or errors while ensuring you get the desired results every time. Automation allows you to break down the hard parts of a task into smaller, easier parts. It also saves you time, because your team members will be free to work on other tasks.

Figure 1: Does your development organization use a continuous integration/build automation process?



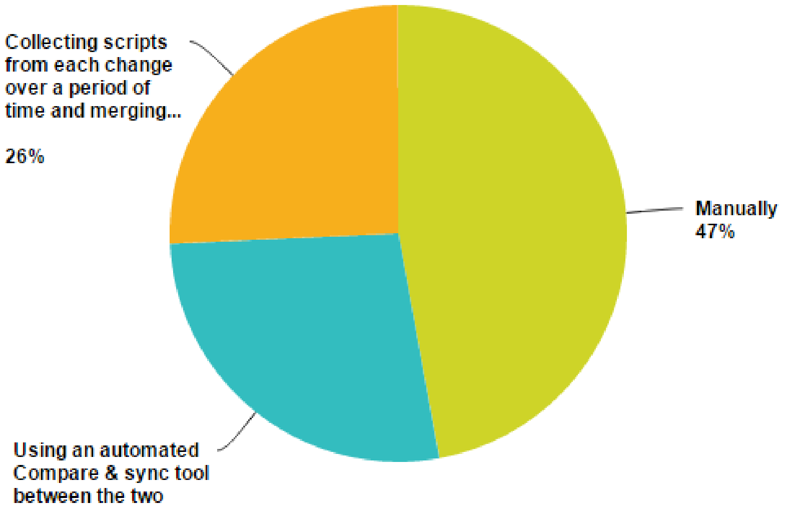
One third of respondents practicing continuous delivery for the database may be lower than the two thirds who practice CD for the application, but it would lead us to believe that a large and growing percentage of organizations are adopting the same best practices they already use for the application - to the database.

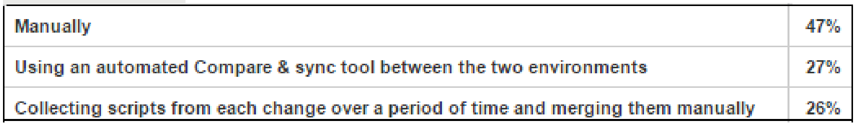
But when asked about the specifics of the automatic processes organizations have in place, a different picture emerged.

When asked how their deployment scripts were created, in order to propagate changes between environments, almost half said manually and another quarter said that it is done by collecting scripts from each change to the database over a period of time and then merging them manually. This left just over a quarter who are actually practicing continuous delivery by using an automated compare and sync tool between the environments. (Figure 2)

A compare and sync tool generates the database change script automatically by comparing the source environment (development) to the target environment (test, UAT, Production, etc.). This saves the developers and DBAs time because they don't have to manually maintain the script if it is a create script or an alter script for the release. Scripts can be generated when needed, and refer to the current structure of the target environment.

Figure 2. How is the deployment script created?

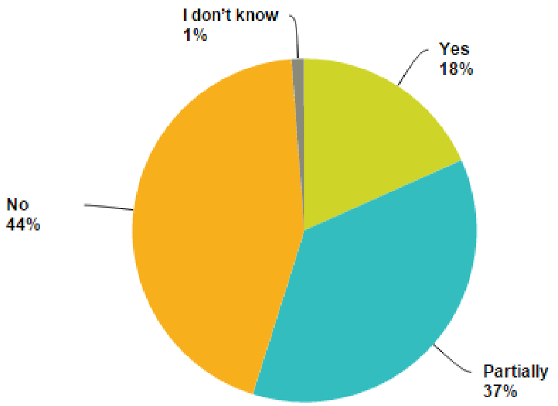




Manually	47%
Using an automated Compare & sync tool between the two environments	27%
Collecting scripts from each change over a period of time and merging them manually	26%

Based on those responses, only 27% of developers are practicing full automation on their database, but the next question would reduce the number even further. When asked if their database build and deploy process is automated, only 18% replied in the affirmative. (figure 3)

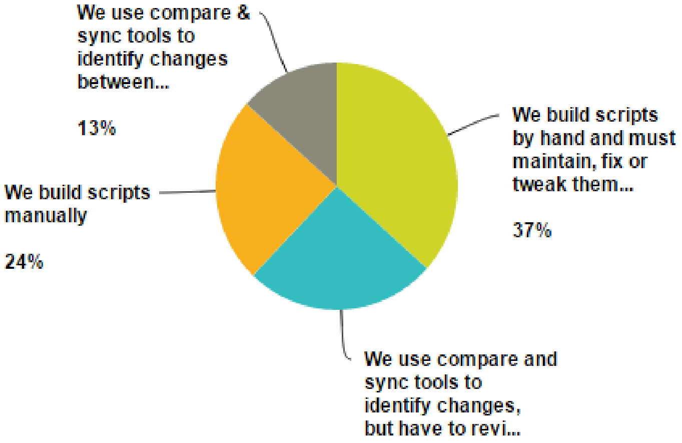
Figure 3. Is your database build and deploy process automated?



Finally, the survey asked participants to choose which of four options best described the way the respondents deploy database changes. 87% chose options that involve manual processes including 37% who replied that they build scripts by hand and then fix or tweak them regularly, 26% who use compare and sync tools to identify changes, but have to review and fix the results because they can't always trust them to automatically deploy correctly and 24% who build scripts manually.

Only 13% responded that they use compare & sync tools to identify changes between environments, and automatically deploy them with no issue. (Figure 4)

Figure 4. Which of these options best describes the way you deploy database changes?



We build scripts by hand and must maintain, fix or tweak them regularly as part of a deployment process	37%
We use compare and sync tools to identify changes, but have to review and fix the results as we can't always trust them to automatically deploy correctly	26%
We build scripts manually	24%
We use compare & sync tools to identify changes between environments, and automatically deploy them with no issue.	13%

Thus, despite 1 in 3 claiming that they have automated continuous process in place for the database, less than 1 in 8 are actually leveraging automation.

The reason for this discrepancy is that many developers believe they are indeed practicing continuous delivery because they consider the manual process which is plugged into the automated process to be a successful implementation of CD. But there are major shortages in a "partially automated" process:

The segments that are done manually are not repeatable, one has to start again each time from the beginning, and not just improve on top of a stable base. The reason for this is that the base might change next time, even slightly. Thus it is not scalable and it does not work for high frequency, rather it depends on specific people doing what they do, repeatedly.

Continuous on the other hand, is by definition, full automation.

Fear of database automation still reigns

The "state of CD for the database" survey conducted by DBmaestro in 2014 confirmed that the vast majority believe CD for the database is possible. (figure 5) This DevOps for database survey revealed that significant percentages are also attempting CD for the database, but what's holding them back is a fear of automation.

In this survey, 26% who use compare and sync tools to identify changes, feel they have to review and fix the results because they can't always trust them to automatically deploy correctly. (figure 4) In the 2014 survey by DBmaestro, fear of automation was listed as the number one reason for not implementing CD for the database.

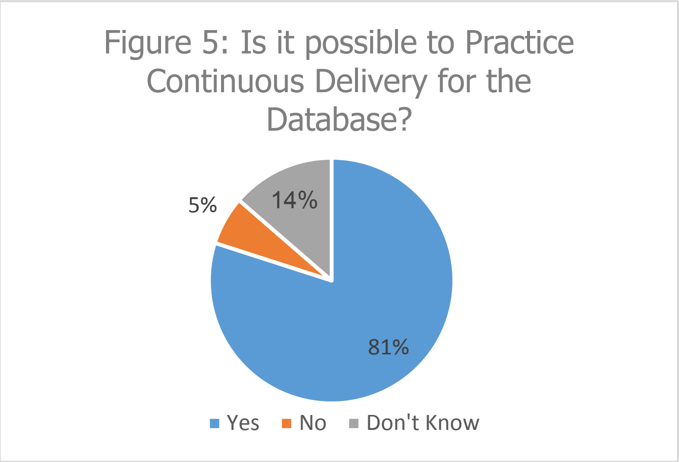


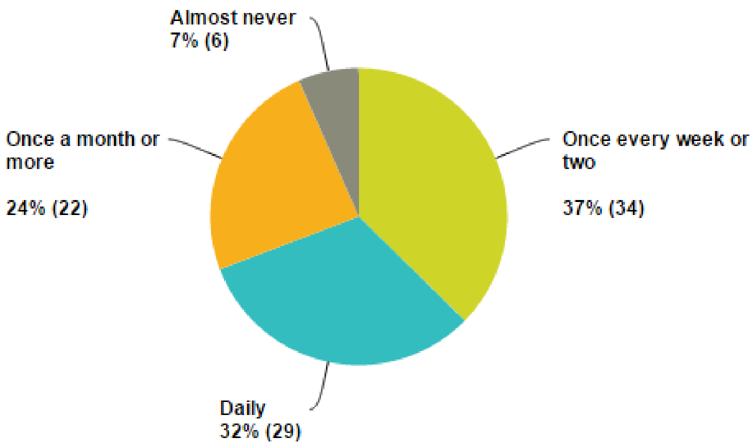
Figure 5: Is it possible to Practice Continuous Delivery for the Database?

Organizations are making frequent database changes

The lack of database automation processes is despite two thirds responding that their companies make frequent database changes. In response to the questions of how often their organization makes changes to the database, 32% said daily and 37% responded weekly. (figure 6)

Best practices are important to implement and follow if you are traveling the fast lanes of the agile world. Frequent changes should be the motivation for automation to take center stage in order to increase overall efficiency and reduce the risks inherent in frequent changes and releases.

Figure 6. How often do you make changes to your database?

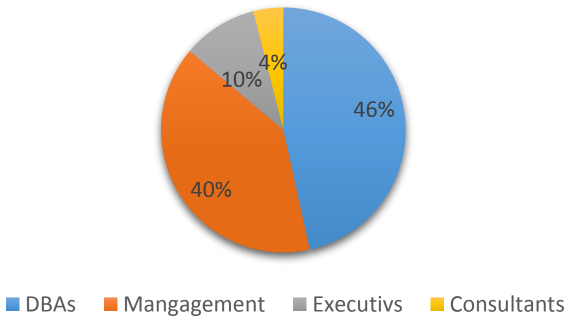


Continuous Delivery Must Come From All Levels

Continuous Delivery is a cross-organizational effort to be embraced by all parts of the organization. When it comes to the database, 46% said DBAs should be responsible for adoption CD for the database while a similar 40% said it is the responsibility of Management.

This flies in the face of a commonly held belief that either the developer or top executives alone are responsible for adopting new development methods, when in reality; in order for it to be successful it must be accepted throughout the organization. (figure 7)

Figure 7: Who within the organization should be responsible for the adoption of Continuous Delivery for the Database?



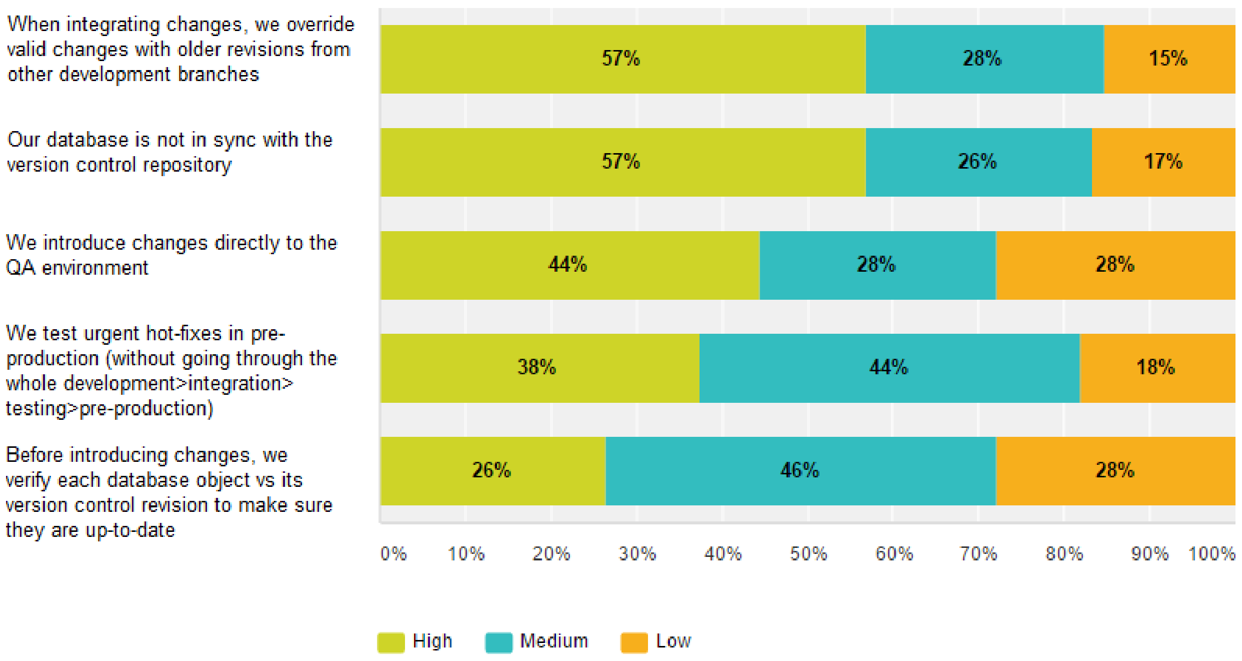
Developers see high risk in manual processes

When asked to rate the risk level of five non-automated, database development and deployment processes, respondents found significant risks.

57% considered it high risk for the database to be out of sync with the version control repository (26% considered it medium risk), 57% said it was a high risk to override changes when integrating valid changes (28% said medium) and 44% said it was high risk to integrate changes directly into the QA (28% said medium). (Figure 8.)

The inability to trust the validity of their version control repository, overriding other people's changes and breaking the deployment order are exactly the reasons why developers and DBAs have a hard time trusting their change management process. Out of date changes being deployed over more recent ones when automating the compare & sync tools results, is a common reason why these tools are not used in the continuous delivery pipeline.

Figure 8. Rate the level of risk associated with the below statements



Conclusions

As more organizations continue to increase the frequency of their database updates, automation best practices become more important in order to increase overall efficiency and reduce the risks inherent in frequent changes and releases.

Developers recognize this. That is why a large number are making a sincere effort to implement DevOps and continuous delivery for the database.

But the belief that they are implementing continuous process does not always meet reality. The vast majority of organizations have clearly defined, highly automated process for the code, but are then plugging in to manual process for the database. This is both risky and inefficient.

Database professionals recognize that database continuous delivery is possible, and vitally important, but organizations will have to overcome this lack of trust in automating the database if they wish to increase productivity, reduce time to market and maintain that competitive advantage.

To learn more about the challenges of database deployment automation and how to overcome those challenges, read "The Challenges and Pitfalls of Database Deployment Automation" white paper.