

DBmaestro and Jenkins - Simplifying Database Continuous Integration (CI)

DevOps teams interested in adopting Continuous Integration (CI) will tell you that Jenkins is the go-to tool. As one of the most versatile tools in the DevOps space, it commands over 71% of the total market share. With over a million active users as of now, it's highly popular for its flexibility and adaptability, helping users with regular deployments, daily backups, and weekly cleanups; all the way to executing projects with multiple plugins.



The same applies to Git, which has become an essential component of DevOps setups today. Almost 90% of developers prefer Git as their version control or source code management system while executing large projects with multiple stakeholders. It provides an isolated environment to test every change in the codebase and helps to preserve the master branch.

Besides being an open source, lies in its distributed peer-to-peer model which is particularly suitable for projects with complex structures.

DBmaestro and Jenkins: A Potent Combo

When it comes to databases, code build automation is taken care of by tools like Jenkins, but processes are still done manually. This makes them risky and difficult to modify when it comes to bigger projects. Matters get even more complicated when there is limited manpower and the time-to-market times need to be reduced. This is when corners are cut (i.e. - testing).

With the CI industry projected to grow at more than 18% every year, we will see a rise in tools that make CI/CD processes faster, secure, and scalable. Even though Jenkins works wonderfully as a CI tool, it does not assist in documentation or version control. This leaves the gap wide open for human error, painful rollbacks, configuration drifts, and overall frustration.

But there is a silver lining - Jenkins has over 1700 plugins that can help solve problems unique to a project. One can enhance productivity with plugins that take care of source code management, UI/UX, administration, building management, platforms, etc. One such plugin is the [DBmaestro Jenkins Plugin](#) - an automation solution that is helping eliminate a lot of pain points.

The DBmaestro devOps platform Jenkins plugin presents a solution to elegantly integrate and swiftly execute projects, while breaking down walls between your development and operations teams. This integration allows you to secure the process of adding database deployments into your CI/CD pipeline, while seamless syncing with your existing ecosystem.

The end results - you can instantly improve the deployment of incremental database changes, which is a key DevOps requirement. Furthermore, you can enforce best practices for the database, automate releases, mitigate downtime risk, and run audits for traceability and compliance. All company security and development policies become easier to enforce.

DBmaestro, when used in tandem with Jenkins, gives you:

- Automated database Devops
- Less release bottlenecks and errors
- Saved time, money, and resources with automated procedures
- Better policy enforcement with user and role management
- Improved security and compliance standards

Thanks to the aforementioned benefits and advantages, you can accelerate your time to market with zero disruptions. The Database Administrator (DBA) can finally govern and automate database releases, while preventing the obscenely expensive downtime which can cause extensive operational and business damage (product launch delays, buggy releases, etc.).

Learn More About DBmaestro and Jenkins working together to Create Seamless Database DevOps - [Click Here](#)

Achieving Zero Disruption

Moving towards zero disruption when introducing new processes is an important goal in any DevOps setup today, especially when it comes to database release management. That's why your database automation solutions need to be agile and should be able to adapt to your existing processes and tools. Integration is the name of the game today.

With DBmaestro you can model a visual database pipeline, a first-of-its-kind in the industry, to steer clear of disruptions and downtimes.

It allows you to build the release process and a path to production while supporting seamless integration with all sources of database changes. The release process built with DBmaestro while incorporating scripts created by developers or third party database source control tools also works with standard SQL coding without the need for an intermediate language.

Another hurdle, especially when working with multiple teams is security and governance. Defining roles, specifying access, and enforcing policies are all important steps when it comes to maintaining razor-sharp focus in extensive projects. With DBmaestro, implementing organizational policies, managing permissions, and meeting compliance regulations are all automated.

Furthermore, it combines version management with PreCheck that can identify and flag configuration drifts and conflicts. By doing this you are essentially preventing rogue code from running and revalidating the final state required to successfully end the process. You can also create custom project-specific policies while creating a detailed audit trail of all changes.

Besides the technical benefits mentioned above, you also stand to gain as a manager or a business owner. Faster development with zero disruption means that you can make informed executive decisions faster, measure the number of successful and failed deployments, the lead time, and the mean time to recovery, and establish the required KPIs for your database teams.

Related: [Zero Downtime Database Deployment](#)

Shortening Feedback Loops

While we are on the topic of pain points in database DevOps, here's one that can cost you a fortune. Feedback loops and their elongated length.

In an increasingly dynamic market, continuous and shortened feedback is a great way to improve quality. One of the major differentiation factors in modern DevOps is its ability to incorporate feedback to improve the functionality and reliability of any software. Whether the feedback comes from your clients or other stakeholder, ignoring it can cost you dearly.

Did You Know?

As per a recent [OverOps survey](#), almost 50% of DevOps professionals claim manual processes are their biggest hurdle to their CI/CD goals.

For the longest time feedback related data was only available to those who were at the top of the analysis pyramid because it was highly valuable to a business and pretty much unaffordable. But with automated databases, actionable insights and performance metrics are available in real-time to improve the productivity of developers, DBAs, and IT professionals.

The process followed by DBmaestro in this regard is particularly helpful.

- A developer deploying changes to your company's software or database will create the code, run its versions, do regression and unit testing, build automation and commit to Git.
- Before those changes are implemented DBmaestro pulls them up and runs your pre-defined permission, policy, and run time checking. The dev team receives feedback about potential issues, if such arise.
- Completed tasks are moved to Jenkins via JIRA. Jenkins then commands DBmaestro to run its checks. For instance, if a developer has not named a table correctly it will pop an error message. In this case, the error may read *"you can't make a new table that doesn't start with project_name_292929 - fix"*. The code is sent back to the first stage for remediation. This is how a short feedback loop looks like.
- Once the detected issues have to be fixed, the code is sent for a dry run where issues such as spelling errors and code rifts may crop up. An email is sent back to the dev team with the details of the issue.
- Only after the dry run is performed smoothly, the DBA can pull the trigger on the release and make sure that there are no issues or bottlenecks that can create disruptions or downtime.

To sum this up, the entire CI/CD pipeline can finally work as one unified entity, where feedback loops are shortened for faster development and improved quality. Also, fewer errors translate to enhanced productivity and engagement with the DevOps process. Because let us face it, it is tough and frustrating to go back to fixing code you deployed a month ago.

Related: [Automated DevOps: The Key to Business Continuity](#)

Why Are Short Feedback Loops Important?

Furthermore, the importance of short feedback loops goes beyond the operational and development points of the CI/CD pipeline. Think consumers. Each user is unique and will use your software in unpredictable ways making it impossible to not have undesirable performance implications. Even a small deployment can increase in your page load time or lead to 404 errors.

Receiving comprehensive feedback on your software's performance helps you understand customer expectations and alter your development processes to match that. It is the difference between speculation and informed decisions.

One is like shooting an arrow in the dark while the other is clearly seeing the target. You also get more time for performance testing.

The other option? Reworking code and patching bad releases. Besides the operational and financial losses, there is brand and reputation damage.

Did You Know?

Downtimes cost anywhere from \$5600 per minute to \$300,000 per hour. This is before we dive into brand damage.

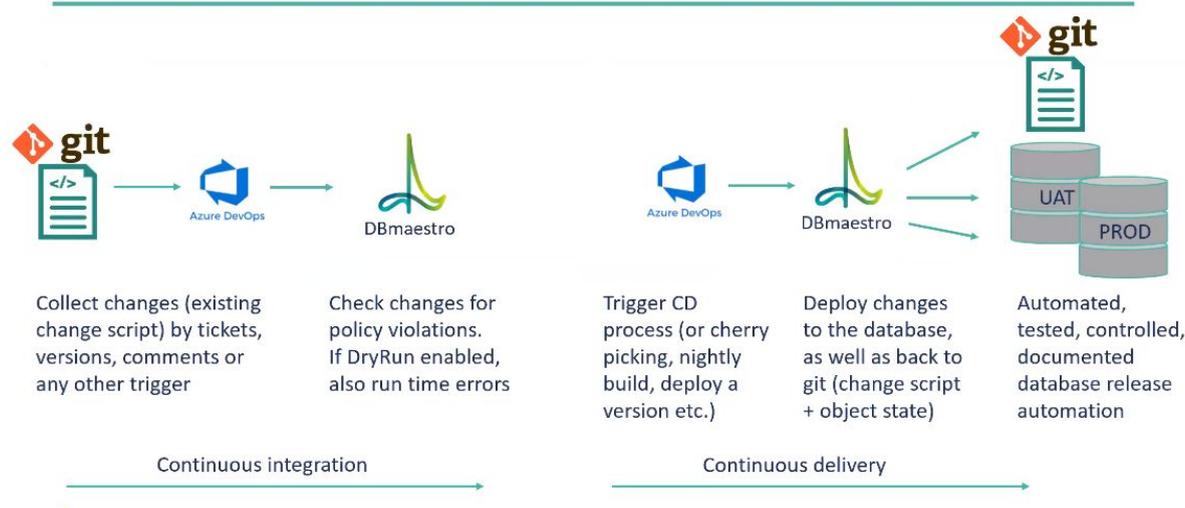
In an ideal scenario, you want to see all possible issues before your clients experience them. When you build an automated release pipeline you can incorporate a strong feedback loop to diagnose issues before the users see them. DBmaestro allows you to check any code in an isolated environment and sends feedback to the DevOps team about its performance.

In cases where issues crop up outside of the testing environment, it provides accountability for each action through its comprehensive audits which allows you to trace who did what, when, and why. Every database action is automatically recorded and documented. This ensures effective rollouts of quality releases and software updates without disrupting user experience.

Freightways - End-To-End CI/CD Pipeline

Freightways, a huge transportation and information management company operating out of New Zealand, recently got to use DBmaestro with Jenkins.

Automated End-to-End CI/CD Process with DBmaestro



The dev teams now collect all script change tickets from Git, trigger the process with Azure DevOps, and let DBmaestro work its magic to look for potential issues that can cause problems at a later stage. DBmaestro essentially gets this package and initiates a CI pre-check process that evaluates security, policies, and dev errors like dropping of tables.

This CI process allows the smooth triggering of Continuous Deployment (CD), where cherry-picking is done when needed or the team simply reverts to nightly builds. Changes are seamlessly deployed to the database and also to Git for automated testing, prior to database release automation that is a part of your app release. Nothing is segmented. It's all hooked together.

Continuous Delivery (CD) Comes Naturally

Once you have your database Continuous Integration (CI) figured out, the Continuous Delivery (CD) aspect is simplified. The right time for CD is, literally anytime. It can be done instantly if an urgent release is required. In other setups, it can be triggered on a nightly or weekly basis. What's important is that the version at the end of every sprint is tested, verified and good to go.

Partial DevOps will accelerate your proverbial sports bike for sure. You will feel the rush as the wind goes through your hair. But the wind will also hinder your vision and prevent you from seeing the potholes that can cause your bike to skid or hit a tree, whatever comes first. DBmaestro and Jenkins are like helmets and gloves. They'll enhance your ride and help you win the race.

This setup addresses all the challenges DevOps teams face today - security, source control, delivery roadblocks, risk, and compliance. With early testing and validation, along with shortened feedback loops, disruptions and downtimes become things of the past. This is the only way to shorten your time-to-market without sacrificing quality and innovation. Get proactive now.

Complete Your CI/CD Pipeline Now

Automate your database releases and shorten your feedback loops to optimize your performance and business metrics

[SCHEDULE MY DEMO NOW](#)