# You Version Your Code, Why Not Database

Submitted By : Kajal Gupta (MSc 2)

For software engineers, it is almost impossible to work without versioning their code. The importance of versioning and tracking history of code changes are long understood in the field of software production. For a long time, database version control was infeasible but it’s no longer the case. Now, database development teams can also catch the benefits of version control.

So here I am sharing some of the reasons why you should version your database code.

## 1. Code Changes can be easily shared within team

Using a version control system to store database code makes it easier to manage the work of the teammates that are responsible for the database. It is especially important for teams located in different locations to be able to share and handle changes quickly. Item locking, for example, helps you avoid disputes and function more efficiently without stepping on each other's toes.

## 2. Improve your understanding of the production pipeline

A version control system shows what development work is being done, how far it is progressing, who is doing it, and the reason behind it. Version control keeps comprehensive change logs and is often used in conjunction with issue tracking systems.

## 3. Retrieve Previous Versions of the Database

Although you should always have a secure backup plan in place, putting a database into version control often makes it easier to back up the SQL code. Version control provides a risk-free sandbox since the history it provides is gradual. This allows developers to experiment with new solutions and roll back safely in the event of errors.

## 4. Demonstrate Compliance and Monitoring Effortlessly

Change monitoring in version control is the first step in getting the database ready for compliance, as well as a vital step in keeping an audit trail consistent and managing risk. Complete revision of a database or database object would be available to check the person who made the changes, the time and reason behind the changes.

## 5. Database Automation could be Achieved

Change management becomes easier with a single version of truth for database code. Complex procedures become easier to automate and replicate, and deployments become even more stable. As code is checked into Source Control as the basis for DLM Automation's automated builds and tests, bugs are discovered sooner and higher-quality code is ultimately shipped and deployed.

## 6. Synchronize Application and Database Code Changes

Database updates can be integrated with application code changes if the database is maintained with version control alongside the application. You'll still know which version of the database corresponds to which version of the application is being used. This direct integration aids in

better communication between teams, increased productivity, and troubleshooting.

## Summary

While it's true that database version control wasn't always achievable, the availability of tools like Liquibase,  SQL Source Control etc. means there is now no reason why the percentage of companies and organizations versioning their database code shouldn't be higher. If you are not currently versioning your database, one of the six reasons mentioned above may persuade you to do so.