



Database as Code CI/CD Pipeline

SUMMARY

As development teams transitioned to building applications on AWS, they faced significant challenges related to database management. With relational databases hosted on Amazon RDS, a centralized database administration team was initially responsible for deploying, updating, and managing database servers, schemas, and data. However, this centralized approach quickly became a bottleneck, slowing productivity of development teams. The reliance on a single team to handle all database-related tasks created delays, and the lack of repeatable processes made work prone to errors. The key issue was how to reduce deployment times for new databases while granting development teams the autonomy to manage their AWS resources in a secure and reliable way.

SOLUTION

The team implemented Flyway, a database migration tool integrated with Amazon RDS and the CI/CD pipeline. This enabled version-controlled database schemas and automatic application, reducing dependency on centralized database administration. Flyway made database migrations an integral part of application deployment, ensuring schema-application version synchronization. This autonomy allowed teams to manage and deploy databases consistently across environments.

The CI/CD pipeline, utilizing a Jenkins server in AWS, consisted of three steps: Deploy Infrastructure (provisioning Aurora PostgreSQL RDS cluster via CloudFormation), Retrieve Values (storing CloudFormation outputs in Jenkins variables), and Execute Database Migrations (applying Flyway scripts). This setup enabled repeatable, consistent, and fully automated database deployments, giving development teams control over database environments and ensuring secure, reliable operations.

RESULTS

500+

Migration scripts created without any external dependencies

50%

Reduction in teams needed to deploy or update databases

< Hour

Reduction in time to deploy a new database went from weeks to less than an hour